Branch and cut algorithms for detecting critical nodes in undirected graphs

Marco Di Summa

Dipartimento di Matematica Pura e Applicata, Università degli Studi di Padova
Via Trieste, 63, 35121 Padova, Italy
e-mail: disumma@math.unipd.it

Andrea Grosso

Dipartimento di Informatica, Università degli Studi di Torino
Corso Svizzera, 185, 10149 Torino, Italy
e-mail: grosso@di.unito.it

Marco Locatelli

Dipartimento di Ingegneria Informatica, Università di Parma
Via G.P. Usberti, 181/A, 43124 Parma, Italy
e-mail: locatell@ce.unipr.it

Abstract

In this paper we deal with the critical node problem, where a given number of nodes has to be removed from an undirected graph in order to maximize the disconnections between the node pairs of the graph. We propose an integer linear programming model with a non-polynomial number of constraints but whose linear relaxation can be solved in polynomial time. We derive different valid inequalities and some theoretical results about them. We also propose an alternative model based on a quadratic reformulation of the problem. Finally, we perform many computational experiments and analyze the corresponding results.

Keywords: Critical Node Problem, Branch and Cut, Valid Inequalities, Reformulation-Linearization Technique

1 Introduction

This paper deals with the problem of deleting from an undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$, whose cardinality is bounded from above by a given K, in order to obtain a residual graph $G[V \setminus S]$ (i.e., the subgraph induced by $V \setminus S$) where the number $f(S)$ of node pairs $\{i, j\}$ connected by at least one path is as small as possible. Following [2] we name this the *critical node problem* (CNP).

From a practical point of view, the problem is interesting in assessing the robustness (or symmetrically, the vulnerability) of network structures; indeed the objective function $f(S)$ measures the minimum number of connections that can be still reliably guaranteed on G after a "smart" attacker has damaged $K \leq |V|$ nodes. With this aim in mind, similar problems have been studied in the literature.

The problem can be seen as a multicommodity version of the *interdiction problems* pioneered by Wollmer [19], and later developed by Wood [20] and Smith and Lim [18].

Myung and Kim [12] tackle the problem of deleting a limited number of *edges* from an undirected graph in order to minimize the *weighted* number of connections guaranteed in the residual graph. Their approach is based on a Mixed Integer Linear Programming (MILP) model and branch and cut. Such an MILP formulation has a non-polynomial number of constraints. They report computational experience on instances with up to 100 nodes and 200 edges.

Matsziw and Murray [11] propose a polynomial-size MILP formulation, and apply it to instances derived from the Ohio's state highway network.

Dinh et al. [6] focus on telecommunication networks; they propose algorithms for detecting in a *directed* graph what they call *node-disruptors* and *arc-disruptors*, i.e., sets of nodes and arcs to be deleted in order to minimize the number of directed connections surviving in the residual graph.

Arulselvan et al. [2] precisely formalize the CNP, establish its NP-hardness for general graphs, and propose an MILP model and a simple and effective heuristic, which is tested on sparse graphs. They also point out applications in immunization problems for populations or computer networks: if the graph G represents physical links between computers or contacts between people, vaccinating the optimal set of critical nodes S^* would minimize the ability of malicious software or infective diseases to spread on the graph, since the disease cannot spread through immunized nodes. Also, Boginski and Commander [3] present applications of CNPs in biology, in order to achieve maximum fragmentation of protein-protein interaction graphs through node deletions. For the case where the connections between node pairs have weights subject to uncertainty, a robust optimization model is considered in [8].

The paper is structured as follows. In Section 2 we extend and enhance the model provided by Arulselvan et al. [2], switching to a formulation with non-polynomial size (related to that in [12]), working in the branch and cut framework. In Section 3 we introduce a number of valid inequalities to tighten the model and we present a theoretical study of such inequalities giving insights about the polyhedral structure of the problem. In Section 4 we develop stronger — although more time consuming — relaxations based on a quadratic programming reformulation of the problem. Finally, in Section 5 we perform a computational study on sparse as well as dense (with density up to 30%) graphs.

2 Basic models

Throughout the paper we use the following notation. $G = (V, E)$ is the graph of the problem, with node set $V = \{1, 2, \ldots, N\}$, hence $N = |V|$. We denote the edges $\{i, j\} \in E$ with the shorthand notation $ij \in E$. The set of neighbors of a node $v \in V$ is $N(v) = \{u \in V : uv \in E\}$. When we deal with a special subgraph H of G — paths and/or cycles — the sets of nodes and edges of H are denoted by $V(H)$ and $E(H)$ respectively.

The first model we consider is drawn from Arulselvan et al. [2]. We consider the CNP in its complementary form: delete at most K nodes in order to disconnect as many node pairs $\{i, j\}$ as possible in the residual graph. Define binary variables $x = (x_i : i \in V)$, with $x_i = 1$ if and only if node $i \in V$ is deleted from the graph, and binary variables $y = (y_{ij} : i, j \in V, i < j)$ with $y_{ij} = 1$ if and only if nodes i and j are disconnected in the residual subgraph. A model with $O(N^3)$ constraints is the following:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{\substack{i,j \in V \\ i<j}} y_{ij} && && (1)\\
\text{subject to} \quad & \sum_{i \in V} x_i \le K && && (2)\\
& x_i + x_j \ge y_{ij} && ij \in E && (3)\\
& y_{ij} + y_{jk} - y_{ik} \ge 0 && i,j,k \in V,\ i<j<k && (4)\\
& y_{ij} - y_{jk} + y_{ik} \ge 0 && i,j,k \in V,\ i<j<k && (5)\\
& -y_{ij} + y_{jk} + y_{ik} \ge 0 && i,j,k \in V,\ i<j<k && (6)\\
& x_i \in \{0,1\},\ y_{ij} \in \{0,1\} && i,j \in V,\ i<j. && (7)
\end{aligned}
$$

Constraint (2) ensures that at most K nodes are deleted; constraints (3) guarantee that if two nodes i, j linked by an edge are to be disconnected, then at least one of i, j is to be deleted. The $O(N^3)$ constraints (4)–(6) model a series of implications in order to ensure that if a node pair $\{i, j\}$ is disconnected in the residual graph then we cannot have both pairs $\{i, k\}$ and $\{j, k\}$ connected. Note that the variables y_{ij} are not strictly required to be binary, but can be relaxed to bounded variables $y_{ij} \in [0, 1]$. Apart from using the complementary version of the problem, this model is the one used in [2]. It can be seen as an adaptation of the model proposed by Matisziw and Murray for the edge deletion problem tackled in [11].

Model (1)–(7) can be derived from a model exhibiting an exponential number of constraints, similar to the one used by Myung and Kim [12] again for the edge-deletion case. Keep the same variables as above, and let $\mathcal{P}(i, j)$ be the set of paths linking nodes i and j in G. Then the problem can be written as

$$
\begin{aligned}
\text{maximize} \quad & \sum_{\substack{i,j \in V \\ i<j}} y_{ij} && && (8)\\
\text{subject to} \quad & \sum_{i \in V} x_i \le K && && (9)\\
& \sum_{r \in V(P)} x_r \ge y_{ij} && P \in \mathcal{P}(i,j),\ i,j \in V,\ i<j && (10)\\
& x_i \in \{0,1\},\ y_{ij} \in \{0,1\} && i,j \in V,\ i<j. && (11)
\end{aligned}
$$

Again, variables y_{ij} are not strictly required to be binary and can be relaxed to bounded variables $y_{ij} \in [0, 1]$. Also note that for $ij \in E$ the unique non-redundant constraint among those in (10) is $x_i + x_j \ge y_{ij}$.

Constraints (10) (also called in what follows *path inequalities*) state that in order to disconnect two nodes i and j, at least one node for each path linking i and j must be deleted from the graph. The number of these constraints is clearly not polynomially bounded, but they can be separated by solving node-weighted shortest path problems as follows. Given an optimal solution (x^*, y^*) for the LP relaxation of (8)–(11), define an auxiliary directed complete graph

$H = (V, A)$ with $A = V \times V$ and arc weights $w_{ij} = x^*_j$; let P^*_{ij} be the minimum-weight path in H linking the node pair $i < j$, and $w(P^*_{ij})$ be its total weight. Then the most violated inequality (10) among those for the paths in $\mathcal{P}(i, j)$ is

$$\sum_{r \in V(P^*_{ij})} x_r \geq y_{ij} \qquad \text{if} \qquad x^*_i + w(P^*_{ij}) < y^*_{ij}.$$

If $x^*_i + w(P^*_{ij}) \geq y^*_{ij}$, no inequality for paths in $\mathcal{P}(i, j)$ is violated. The detection of the most violated inequalities (10) can be carried out in $O(N^3)$ time, for example by means of the Floyd-Warshall algorithm (see, e.g., [9]).

For the sake of completeness, we also considered a bilevel approach following the classical interdiction models (see [20]). We state the CNP as

$$\begin{aligned} &\text{maximize} && \sum_{\substack{s,t \in V \\ s<t}} f_{st}(x) && (12)\\ &\text{subject to} && \sum_{i \in V} x_i \leq K && (13)\\ &x_i \in \{0, 1\}, && i \in V, && (14) \end{aligned}$$

where the function $f_{st}(x) = 0$ if a path *exists* between nodes s, t in $G[V \setminus S(x)]$, with $S(x) = \{i \in V : x_i = 1\}$. We set $f_{st}(x) = 1$ if such a path does not exist. The value of $f_{st}(x)$ can be computed by solving the inner (un)reachability problem

$$\begin{aligned} f_{st}(x) = \max\ & u^{st}_s - u^{st}_t && && (15)\\ \text{subject to}\ & u^{st}_i - u^{st}_j \leq z_{ij} && ij \in E && (16)\\ & -u^{st}_i + u^{st}_j \leq z_{ij} && ij \in E && (17)\\ & u^{st}_s - u^{st}_t \leq 1 && && (18)\\ & x_i \leq z_{ij} && ij \in E && (19)\\ & x_j \leq z_{ij} && ij \in E && (20)\\ & x_i + x_j \geq z_{ij} && ij \in E && (21)\\ & u^{st}_i \in [0, 1] && i \in V && (22)\\ & z_{ij} \in [0, 1] && ij \in E. && (23) \end{aligned}$$

Problem (15)–(23) corresponds to the dual of a shortest $s - t$ path problem (or, equivalently, a unit-commodity minimum cost flow) on the directed auxiliary graph $H_{st}(\hat{V}, \hat{A})$ where

$$\begin{aligned} &\hat{V} = V \setminus S(x), \quad \text{and}\\ &\hat{A} = \{(i, j), (j, i) : ij \in E \text{ and } i, j \notin S(x)\} \cup \{(s, t)\}. \end{aligned}$$

All arcs in $\hat{A}$ have zero cost, except a unit-cost arc (s, t). Constraints (16)–(18) implement the dual problem, while constraints (19)–(21) account for the presence/absence of arcs in $\hat{A}$: $z_{ij} = 1$ if $(i, j), (j, i) \notin \hat{A}$, while $z_{ij} = 0$ if $(i, j), (j, i) \in \hat{A}$.

Combining the two problems, the CNP can be written as

$$\begin{aligned}
\text{maximize} \quad & \sum_{\substack{s,t\in V\\ s<t}} (u_s^{st} - u_t^{st}) && (24)\\
\text{subject to} \quad & \text{constraints (16)–(18)} \quad \text{for all } s,t \in V,\ s<t && (25)\\
& \text{constraints (19)–(21)} && (26)\\
& \sum_{i\in V} x_i \le K && (27)\\
& x_i \in \{0,1\} \qquad i \in V, && (28)\\
& z_{ij} \in [0,1] \qquad ij \in E. && (29)
\end{aligned}$$

Model (24)–(29) has $O(|V|^3)$ variables and $O(|V|^2 \cdot |E|)$ constraints. We note that the number of constraints (16)–(18) coming from the inner models is large, since all node pairs $s < t$ have to be taken into account.

3 Valid inequalities

In this section we discuss some families of valid inequalities for formulation (8)–(11) of the CNP. The inequalities that we derive are based on the presence of special configurations in the graph, such as cliques or cycles. Some of the configurations and inequalities considered here are the ones that gave the best results in our computational experiments (see Section 5).

3.1 Clique inequalities

Let $Q \subseteq V$ be a clique of G. Then formulation (8)–(11) includes the following constraints:

$$y_{ij} \le x_i + x_j, \quad i,j \in Q, i<j, \tag{30}$$

$$x_i, y_{ij} \in \{0,1\}, \quad i,j \in Q, i<j. \tag{31}$$

Note that (30) contains all the non-redundant inequalities of the model that involve only variables x_i, y_{ij} relative to nodes of Q, except for the cardinality constraint (9). We describe and discuss a family of inequalities, which we refer to as *clique inequalities*, that are valid for all the feasible solutions to (30)–(31).

Proposition 1 *Let $Q \subseteq V$ be a clique of G, with $|Q| = q$. Then, for every integer $1 < t < q$, the following* clique inequality *is valid:*

$$\sum_{\substack{i,j\in Q\\ i<j}} y_{ij} \le (q-t)\sum_{i\in Q} x_i + \frac{t(t-1)}{2}. \tag{32}$$

Proof. Since inequality (32) involves only variables x_i, y_{ij} with $i,j \in Q$, in this proof we ignore all other variables.

We fix any feasible solution (x,y) of (30)–(31) and show that (x,y) satisfies inequality (32). Define $R = \{i \in Q : x_i = 1\}$ and $r = |R|$. Constraint (30) implies that $y_{ij} = 1$ only if at least one of i,j belongs to R. Thus

$$\sum_{\substack{i,j\in Q\\ i<j}} y_{ij} \le r(q-r) + \frac{r(r-1)}{2}. \tag{33}$$

Then, in order to prove that (x, y) satisfies (32), it is sufficient to show that

$$r(q-r) + \frac{r(r-1)}{2} \le r(q-t) + \frac{t(t-1)}{2}.$$

To see that the above inequality holds, note that from simple calculations one finds

$$r(q-t) + \frac{t(t-1)}{2} - r(q-r) - \frac{r(r-1)}{2} = \frac{(t-r)(t-r-1)}{2} \ge 0, \tag{34}$$

where the last inequality is a consequence of r and t being integer values. □

Note that inequality (32) is valid also when $t = 1$ or $t = q$. However, in these cases it is implied by the inequalities defining the linear relaxation of (30)–(31): specifically, for $t = 1$, (32) can be obtained by summing inequalities $y_{ij} \le x_i + x_j$ for all $i, j \in Q, i < j$; and for $t = q$, (32) can be obtained by summing inequalities $y_{ij} \le 1$ for all $i, j \in Q, i < j$.

On the contrary, inequality (32) is never implied by the original inequalities when $1 < t < q$, as shown by the vector (x, y) defined by $x_i = 1/2$ for all $i \in Q$ and $y_{ij} = 1$ for all $i, j \in Q, i < j$, which is in the linear relaxation of (30)–(31) but violates (32).

3.1.1 Strength of the inequalities

One might wonder whether, for $1 < t < q$, every clique inequality (32) induces a facet of the convex hull of (30)–(31). Computational tests carried out with softwares specifically designed for the analysis of polyhedra (such as PORTA [5]) suggest that indeed this might be the case. Though we do not have a proof that this is true for all $1 < t < q$, we can prove a weaker result.

Proposition 2 *Let $Q \subseteq V$ be a clique of G, with $|Q| = q$. Then for $t \in \{2, q-1\}$ inequality* (32) *induces a facet of the convex hull of* (30)–(31)*.*

Proof. Let $d = q + q(q-1)/2$ be the number of variables of the model. In order to prove that inequality (32) induces a facet of (30)–(31), we show that there are d affinely independent points in (30)–(31) that satisfy (32) at equality.

For any fixed $r = 0, \dots, q$, we define a family S_r of feasible solutions to (30)–(31): $(x, y) \in S_r$ if and only if (i) exactly r components of x are equal to 1 and (ii) $y_{ij} = 1$ if and only if at least one of x_i, x_j is equal to 1. Note that $|S_r| = \binom{q}{r}$ and the sets $S_0, \dots, S_q$ are pairwise disjoint.

Fix an integer $1 < t < q$. It is easy to check that every point $(x, y) \in S_r$ satisfies (33) at equality. Thus, using the equation in (34), we see that a point $(x, y) \in S_r$ satisfies (32) at equality whenever $r = t$ or $r = t - 1$.

Now assume that $t = 2$. By the above arguments, the set $S_1 \cup S_2$ contains exactly d points that satisfy (32) at equality. To show that inequality (32) is facet-inducing, it remains to prove that the points in $S_1 \cup S_2$ are affinely independent.

Let M be the $d \times d$ matrix whose rows are the vectors in $S_1 \cup S_2$. We assume that the first q rows of M contain the vectors in S_1 and the other $q(q-1)/2$ rows contain the vectors in S_2. Also, the first q columns of M correspond to the x-components and the other $q(q-1)/2$ columns correspond to the y-components. Accordingly, we decompose M as follows: $M = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ — see the first part of Figure 1 to fix the ideas. Note that by choosing a suitable ordering for the first q rows of M we can assume that $A = I_q$ (the identity matrix of size $q \times q$). Since the column of B corresponding to variable y_{ij} has a 1 in rows i and j, and 0 elsewhere, we can transform B into the all-zero matrix with the following elementary operations on the columns of M: for every ij, subtract the columns corresponding to variables x_i and x_j from the column corresponding to variable y_{ij}. This yields a matrix $M' = \begin{pmatrix} I_q & 0 \\ C & D' \end{pmatrix}$ with $\det(M) = \det(M') = \det(D')$. Now it can be checked that D' has precisely one -1 per row and per column (and 0 elsewhere): more specifically, the row of D' corresponding to the feasible solution with $x_i = x_j = 1$ has a -1 in the column corresponding to variable y_{ij} and 0 elsewhere. Then $|\det(M)| = |\det(D')| = 1$. This shows that the points in $S_1 \cup S_2$ are linearly independent, thus they are also affinely independent.

Now assume that $t = q-1$. In this case we take the points in $S_{q-1} \cup S_{q-2}$: there are exactly d points in $S_{q-1} \cup S_{q-2}$ and they all satisfy (32) at equality. Using these vectors, we construct a matrix M similarly to what we did before. Since the property of being (or not being) affinely independent is not affected by translating all the points by a given vector, we can subtract the all-one vector from every row of M, thus obtaining a matrix $M' = \begin{pmatrix} A' & B' \\ C' & D' \end{pmatrix}$. It is easy to check that both A' and D' have precisely one -1 per row and per column (and 0 elsewhere), and $B' = 0$ (see the second part of Figure 1). It follows that $|\det(M')| = 1$. Thus the rows of M' are affinely (actually, linearly) independent, which implies that the rows of M are affinely independent, as well. □

If $2 < t < q-1$, using the same arguments and notation as in the above proof, we know that the points in $S_{t-1} \cup S_t$ satisfy (32) at equality. Furthermore, the number of points in $S_{t-1} \cup S_t$ is larger than d. Thus, in order to prove that inequality (32) is facet-inducing also for these values of t, it would be sufficient to show that $S_{t-1} \cup S_t$ contains d affinely independent points. However, we currently do not have a proof for this.

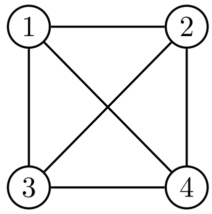
3.1.2 Sub-cliques

Since every clique with q nodes contains $\binom{q}{\bar{q}}$ sub-cliques with $\bar{q}$ nodes ($\bar{q} \le q$), one might ask whether all the corresponding clique inequalities are needed in the description of the convex hull of the larger model (i.e., the one describing the clique with q nodes). It turns out that every clique inequality that is facet-inducing for the $\bar{q}$-clique model is also facet-inducing for the q-clique model. Indeed, we can prove a more general result.

Proposition 3 *Let $G = (V, E)$ be any graph and let $Q \subseteq V$ be a clique of G. Every inequality $ax + by \le \gamma$, with $\gamma \neq 0$, that is facet-inducing for the model for Q is also facet-inducing for the model for G.*

Proof. We assume without loss of generality that $V = \{1, \dots, N\}$ and $Q = \{1, \dots, N-p\}$, where $p > 0$. In this proof, the model for G (respectively, Q) will be called the larger (respectively, smaller) model. Note that the larger model has $p + p(N-p) + p(p-1)/2$ more variables than the smaller one, namely x_i for $i \in V \setminus Q$, y_{ij} for $i \in Q$ and $j \in V \setminus Q$, and y_{ij} for $i, j \in V \setminus Q, i < j$.

If an inequality $ax + by \le \gamma$ is facet-inducing for the smaller model, then there exist d affinely independent solutions of the smaller model that satisfy $ax + by = \gamma$, where d is the number of variables of the model. Let us denote these points by $z^1 = (x^1, y^1), \dots, z^d =$



$$
\begin{array}{cc}
 & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 12 & 13 & 14 & 23 & 24 & 34 \end{array} \\
\begin{array}{c} S_1 \\ S_1 \\ S_1 \\ S_1 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \end{array} &
\left(\begin{array}{cccc|cccccc}
1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\
\hline
1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1
\end{array}\right)
\end{array}
\longrightarrow
\begin{array}{cc}
 & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 12 & 13 & 14 & 23 & 24 & 34 \end{array} \\
\begin{array}{c} S_1 \\ S_1 \\ S_1 \\ S_1 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \end{array} &
\left(\begin{array}{cccc|cccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1
\end{array}\right)
\end{array}
$$

$$
\begin{array}{cc}
 & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 12 & 13 & 14 & 23 & 24 & 34 \end{array} \\
\begin{array}{c} S_3 \\ S_3 \\ S_3 \\ S_3 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \end{array} &
\left(\begin{array}{cccc|cccccc}
0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1
\end{array}\right)
\end{array}
\longrightarrow
\begin{array}{cc}
 & \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 12 & 13 & 14 & 23 & 24 & 34 \end{array} \\
\begin{array}{c} S_3 \\ S_3 \\ S_3 \\ S_3 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \\ S_2 \end{array} &
\left(\begin{array}{cccc|cccccc}
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
-1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\
-1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
-1 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0
\end{array}\right)
\end{array}
$$

Figure 1: Illustration of the proof of Proposition 2 for a clique with four vertices.

(x^d, y^d). We show how to extend this set of d points in $\{0,1\}^d$ to a set of $\bar{d}$ affinely independent points in $\{0,1\}^{\bar{d}}$ that are feasible for the larger model and satisfy $ax + by = \gamma$, where $\bar{d} = d + p + p(N-p) + p(p-1)/2$.

(a) The first d points are defined as follows: for $k = 1, \ldots, d$, we extend point z^k by setting $x_i = 0$ and $y_{ij} = 0$ for all the additional variables that have been introduced.

(b) Then we define the following p points: for each fixed $i \in V \setminus Q$, we take point z^1 and extend it by setting $x_i = 1$ and all other additional variables equal to 0.

(c) The next $p(N-p)$ points are defined as follows. Fix $i \in Q$ and $j \in V \setminus Q$. Since $\gamma \neq 0$, the origin does not satisfy $ax + by = \gamma$. Thus there exists an index k such that $x_i^k = 1$. We then take point z^k and extend it by setting $y_{ij} = 1$ and all other additional variables equal to 0.

(d) Finally, we define the last $p(p-1)/2$ points: for $i, j \in V \setminus Q, i < j$, we take point z^1 and extend it by setting $x_i = y_{ij} = 1$ and all other additional variables equal to 0.

We denote the above points by $\bar{z}^1, \ldots, \bar{z}^{\bar{d}}$, where the first d points $\bar{z}^1, \ldots, \bar{z}^d$ are those constructed in (a).

It can be checked that $\bar{z}^1, \ldots, \bar{z}^{\bar{d}}$ are all feasible for the larger model and they all satisfy $ax + by = \gamma$. We claim that these points are affinely independent. To see this, assume by contradiction that one of the points, say $\bar{z}^\ell$, can be written as an affine combination of the other points. Since, for $i \in V$ and $j \in V \setminus Q$, there is a single point satisfying $y_{ij} = 1$ (either in group (c) or (d)), $\bar{z}^\ell$ cannot be a point in group (c) or (d); furthermore, all points of groups (c) and (d) have coefficient zero in the affine combination. Now, considering only groups (a) and (b), for each $i \in V \setminus Q$ there is a single point satisfying $x_i = 1$ (and this point is in group (b)); then $\bar{z}^\ell$ cannot be a point in group (b) and all points in group (b) have coefficient zero in the combination. It follows that $\bar{z}^\ell$ is in group (a) and $\bar{z}^\ell$ is an affine combination of the other points of group (a). This implies that the original points $z^1, \ldots, z^d$ are affinely dependent, a contradiction. $\square$

Since $\gamma \neq 0$ in the clique inequalities, we immediately have the following result.

Corollary 4 *Let Q be a clique of G and consider a sub-clique $\bar{Q} \subseteq Q$. Every clique inequality that is facet-inducing for the model for $\bar{Q}$ is also facet-inducing for the model for Q.*

Note that if Proposition 2 held for every $1 < t < q$ (as we conjecture), then the above result would show that when giving a linear-inequality description of the convex hull of (30)–(31) one has to include all the clique inequalities for all the sub-cliques of Q.

3.1.3 Cliques with three or four nodes

We now discuss the special cases of a clique with only three or four nodes. For a clique with exactly three nodes, there is a single nontrivial clique inequality (32). It turns out that adding this inequality to the linear relaxation of (30)–(31) gives the convex hull of the set.

Proposition 5 *If $|Q| = 3$, then a non-redundant linear-inequality description of the convex hull of points in* (30)–(31) *is obtained by adding to the linear relaxation the clique inequality* (32) *(with $q = 3, t = 2$).*

Proof. See Section A.1. □

For a clique with exactly four nodes, we can of course write two clique inequalities (32) (with $q = 4$ and $t \in \{2, 3\}$). By Proposition 2, these two inequalities are facet-inducing. However, since a clique with four nodes contains four sub-cliques with three nodes each, it is also possible to write the four clique inequalities corresponding to these sub-cliques. By Proposition 2 and Corollary 4, these inequalities are also facet-inducing. Therefore no clique inequality can be omitted in the description of the convex hull of (30)–(31) for a clique of size 4. On the other hand, no other inequality is required, as stated in the following proposition, whose proof, based on techniques that are similar to those used in the proof of Proposition 9, is omitted.

Proposition 6 *If $|Q| = 4$, then a non-redundant linear-inequality description of the convex hull of points in* (30)–(31) *is obtained by adding to the linear relaxation all clique inequalities* (32) *corresponding to Q, as well as the clique inequalities corresponding to all sub-cliques of Q containing three nodes.*

Unfortunately, a similar result does not hold for cliques containing more than four nodes: computational tests show that already with five nodes many other different types of inequalities are needed to describe the convex hull of (30)–(31).

3.2 Cycles of length 4

Let C be a cycle of length 4 in G with vertex set $V(C)$ and edge set $E(C)$. To simplify notation, we assume that $V(C) = \{1, 2, 3, 4\}$ and $E(C) = \{12, 23, 34, 14\}$. Then formulation (8)–(11) includes the following constraints:

$$y_{ij} \le x_i + x_j, \qquad ij \in E(C), \tag{35}$$
$$y_{ij} \le x_i + x_j + x_k, \quad ij \notin E(C),\ k \in V \setminus \{i, j\}, \tag{36}$$
$$x_i, y_{ij} \in \{0, 1\}, \qquad i, j \in V(C),\ i < j. \tag{37}$$

We describe two types of valid inequalities for the above set of points.

Proposition 7 *The following is a valid inequality for* (35)–(37)*:*

$$y_{13} + y_{24} \le x_1 + x_2 + x_3 + x_4. \tag{38}$$

Proof. We show that inequality (38) can be obtained by applying the Chvátal-Gomory procedure. Consider the following valid inequalities:

$$\begin{aligned}
y_{13} &\le x_1 + x_2 + x_3, \\
y_{13} &\le x_1 + x_3 + x_4, \\
y_{13} &\le 1, \\
y_{24} &\le x_2 + x_3 + x_4, \\
y_{24} &\le x_1 + x_2 + x_4, \\
y_{24} &\le 1.
\end{aligned}$$

Summing them together and dividing by 3, we find $y_{13} + y_{24} \le x_1 + x_2 + x_3 + x_4 + 2/3$. By applying Chvátal-Gomory rounding, we obtain inequality (38). □

Inequality (38) is not implied by the original inequalities, as proven by the following point: $x_i = 1/3$ for all $i \in V(C)$, $y_{ij} = 2/3$ for $ij \in E(C)$, $y_{ij} = 1$ for $ij \notin E(C)$. Also, it is not hard to find 10 affinely independent feasible points that satisfy (38) at equality, thus showing that this inequality induces a facet of the convex hull of (35)–(37).

Proposition 8 *The following is a family of four valid inequalities for* (35)–(37)*:*

$$y_{ij} + y_{jk} + y_{ik} \leq x_1 + x_2 + x_3 + x_4 + 1, \quad i, j, k \in V(C),\ i < j < k. \tag{39}$$

Proof. Fix any feasible solution (x, y) of (35)–(37) and define $R = \{i \in V(C) : x_i = 1\}$ and $r = |R|$. Inequality (39) is trivially satisfied if $r \geq 2$, as the value of the left-hand side cannot exceed 3. If $r = 0$, i.e., $x_1 = x_2 = x_3 = x_4 = 0$, then by (35)–(36) we necessarily have $y_{ij} = 0$ for all $i, j \in V(C), i < j$, thus inequality (38) is satisfied. Finally, assume that exactly one component of x is equal to 1. Then there are at least two indices $\ell_1, \ell_2 \in \{i, j, k\}$ such that $x_{\ell_1} = x_{\ell_2} = 0$. It follows by (35)–(36) that $y_{\ell_1 \ell_2} = 0$ and inequality (39) is satisfied. □

Inequality (39) is not implied by the original inequalities, as proven by the following point: assuming without loss of generality $i = 1, j = 2, k = 3$, we choose $x_1 = x_2 = x_3 = 1/3$, $y_{12} = y_{23} = 2/3$, $y_{13} = 1$, all other entries equal to 0. Again, it is not hard to see that this inequality induces a facet of the convex hull of (35)–(37).

It turns out that the two types of inequalities presented above are sufficient to describe the convex hull of (35)–(37).

Proposition 9 *A non-redundant linear-inequality description of the convex hull of points in* (30)–(31) *is obtained by adding inequalities* (38)–(39) *to the linear relaxation of the set.*

Proof. See Section A.2. □

3.3 Neighborhood inequalities

Further valid inequalities can be defined for our problem. For instance, the clique inequalities described in Section 3.1 can be extended to a more general configuration. Here we do not make any assumption on the structure of the graph.

Proposition 10 *Let $v \in V$ be any node and let $N(v)$ be the set of neighbors of v. Then, for every subset $Q \subseteq N(v)$ with $q = |Q| \geq 3$ and every integer $1 < t < q$, the following inequality is valid:*

$$\sum_{\substack{i,j \in Q \\ i<j}} y_{ij} \leq \left(\frac{q(q-1)}{2} - \frac{t(t-1)}{2} \right) x_v + (q - t) \sum_{i \in Q} x_i + \frac{t(t-1)}{2}. \tag{40}$$

Proof. Let (x, y) be a feasible solution. If $x_v = 1$, then the inequality is trivially satisfied, as the value of the left-hand side cannot exceed $q(q-1)/2$. So we assume $x_v = 0$. Note that for any pair of nodes $i, j \in Q, i < j$, we have the constraint $y_{ij} \leq x_i + x_v + x_j$. If $x_v = 0$, this inequality reduces to $y_{ij} \leq x_i + x_j$. Thus all constraints (30)–(31) are satisfied by our vector. Then, by Proposition 1, inequality (32) is also satisfied. To conclude, note that inequalities (32) and (40) coincide when $x_v = 0$. □

Unfortunately, this class of inequalities, as well as a few others that we tested and are not presented here, did not appear to be very effective, at least according to our computational experiments.

4 Further relaxations

In this section we discuss some further relaxations for the CNP. We immediately point out that, as we will see in Section 5, overall such new relaxations appear to be inferior with respect to those previously discussed, but there are some interesting exceptions (in particular as the density of the graph and the value of K increase) for which they are superior, thus making their presentation worthwhile.

For a given graph $G = (V, E)$, observing that for $ij \in E$ the following equality always holds

$$y_{ij} = x_i + x_j - x_i x_j,$$

we have that a nonlinear (quadratic) model strictly related to model (8)-(11) is

$$\begin{array}{lrl}
\text{maximize} & \displaystyle\sum_{\substack{i,j\in V\\ i<j}} y_{ij} & \\
\text{subject to} & \displaystyle\sum_{i\in V} x_i = K & \\
& \displaystyle y_{ij} \le \sum_{r\in V(P)} x_r & ij \notin E,\ P \in \mathcal{P}(i,j) \\
& y_{ij} = x_i + x_j - x_i x_j & ij \in E \\
& x_i \in \{0,1\},\ 0 \le y_{ij} \le 1 & i,j \in V,\ i<j.
\end{array}$$

Notice that we have an equality constraint in place of the inequality cardinality constraint (9). In fact, this is not a relevant difference, as equality always holds for optimal solutions of (8)-(11). Now, let us introduce the matrix variable

$$X = xx^T.$$

We notice that:

- $x_i \in \{0,1\} \quad \Leftrightarrow \quad x_i^2 = x_i \quad \Leftrightarrow \quad X_{ii} = x_i$
- for all $j \in V$,

$$\sum_{i\in V} x_i = K \quad \Rightarrow \quad x_j \left(\sum_{i\in V} x_i \right) = K x_j \quad \Rightarrow \quad \sum_{i\in V} X_{ij} = K X_{jj}. \tag{41}$$

The last equation is an RLT constraint, where RLT stands for Reformulation-Linearization Technique, see, e.g., [13, 15, 16]. Then, we can add the above constraints to our model, thus obtaining

$$
\begin{aligned}
\text{maximize} \quad & \sum_{\substack{i,j\in V\\ i<j}} y_{ij} && && (42)\\
\text{subject to} \quad & \sum_{i\in V} x_i = K && && (43)\\
& y_{ij} \le \sum_{r\in V(P)} X_{rr} && ij \notin E,\ P \in \mathcal{P}(i,j) && (44)\\
& y_{ij} = X_{ii} + X_{jj} - X_{ij} && ij \in E && (45)\\
& \sum_{i\in V} X_{ij} = K X_{jj} && j \in V && (46)\\
& X = xx^T && && (47)\\
& X_{ii} = x_i && i \in V && (48)\\
& x_i \in \{0,1\},\ 0 \le y_{ij} \le 1 && i,j \in V,\ i<j. && (49)
\end{aligned}
$$

Now we obtain a polynomially solvable relaxation by relaxing the binary constraints $x_i \in \{0,1\}$ into $0 \le x_i \le 1$ and substituting the rank one constraint $X = xx^T$ with some more manageable constraint(s). First, we will always assume symmetry of the matrix X. Next, we discuss two possibilities for the relaxation of the rank one constraint:

- *RLT relaxation*: remove the constraint $X = xx^T$ and substitute it with the RLT inequalities

  $$X_{ij} \le x_i, \quad X_{ij} \le x_j, \quad X_{ij} \ge x_i + x_j - 1, \quad X_{ij} \ge 0, \qquad i,j \in V,\ i<j \qquad (50)$$

  (note that, as previously remarked, also equations (41) can be viewed as RLT constraints).

- *Semidefinite (SDP) relaxation*: remove the constraint $X = xx^T$ and substitute it with the semidefinite constraint

  $$X \succeq xx^T$$

  or, equivalently,

  $$\begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix} \succeq O,$$

  where O is the null matrix and $A \succeq B$ denotes that matrix $A-B$ is positive semidefinite (see, e.g., [14]).

A combination of the two relaxations can also be considered (see, e.g., [1]). It is important to remark that, independently of how we replace the rank one constraint, the separation problem for the path constraints

$$y_{ij} \le \sum_{r\in V(P)} X_{rr}$$

remains identical.

The above relaxation can be improved by strengthening the path constraints. Let

$$i_0 \to \cdots \to i_s, \quad \text{with } i_0 = i \text{ and } i_s = j,$$

be some path $P \in \mathcal{P}(i,j)$. Then, we can replace the path constraint

$$y_{ij} \leq \sum_{r=0}^{s} x_{i_r}$$

with the stronger constraint

$$y_{ij} \leq \sum_{r=0}^{s} x_{i_r} - \sum_{r=0}^{s-1} x_{i_r} x_{i_{r+1}}.$$

The second summation is such that if we remove both i_r and i_{r+1}, we count them only once by subtracting $x_{i_r} x_{i_{r+1}}$. Therefore, recalling that $X = xx^T$, we obtain the constraint

$$y_{ij} \leq \sum_{r=0}^{s} X_{i_r i_r} - \sum_{r=0}^{s-1} X_{i_r i_{r+1}}. \tag{51}$$

If we impose the RLT constraints (50), the separation problem for these constraints is still solvable through the solution of a shortest path problem. Indeed, we only need to redefine the weights of the edges by taking into account also the $X_{i_r i_{r+1}}$ values. If (x^*, X^*, y^*) is the solution of the current relaxation, the arc weights are $w_{ij} = x_j^* - X_{ij}^*$, which are nonnegative in view of (50).

Constraint (51) can be further strengthened. Indeed, we can replace it by any set of constraints

$$y_{ij} \leq \sum_{r=0}^{s} X_{i_r i_r} - \sum_{j_r j_t \in E_T(P)} X_{j_r j_t}, \tag{52}$$

where $E_T(P)$ is the edge set of a spanning tree T over the complete subgraph with vertex set $V(P)$. The following observation proves that these are valid inequalities for our problem.

Proposition 11 *Inequalities (52) are valid for problem* (42)–(49).

Proof. Let $(\bar{x}, \bar{X}, \bar{y})$ be a feasible solution for (42)–(49). If $\bar{x}_i = 0$ for all $i \in V(P)$, then $\bar{y}_{ij} = 0$ and the right-hand side of inequalities (52) is also equal to 0, so that inequalities (52) are satisfied. Otherwise, let

$$V'(P) = \{i \in V(P) \ : \ \bar{x}_i = 1\}, \quad E'_T(P) = \{j_r j_t \colon j_r, j_t \in V'(P),\ j_r j_t \in E_T(P)\}.$$

Notice that

$$\bar{X}_{j_r j_t} = 1 \quad \Longleftrightarrow \quad \bar{x}_{j_r} = \bar{x}_{j_t} = 1.$$

Since $F = (V'(P), E'_T(P))$ is a forest, the right-hand side of (52) is equal to

$$|V'(P)| - |E'_T(P)| \geq 1,$$

and the inequality is then satisfied. $\square$

Constraint (51) is a special case of constraints (52) in which the spanning tree has edges

$$E_T = \{i_r i_{r+1} \colon \ r = 0, \ldots, s-1\}.$$

In fact, by denoting with (x^*, X^*, y^*) the optimal solution of the current relaxation and assigning weight X^*_{ij} to all edges ij with $i, j \in V(P)$, we can find the most violated among constraints (52) by solving a maximum spanning tree problem. Therefore, we can first detect the most violated among constraints (51) by solving a shortest path problem, and then we can strengthen each violated constraint by solving a maximum spanning tree problem.

Finally, we notice that all the valid inequalities introduced in Section 3 can also be added to these new relaxations, and it is not difficult to see that the new relaxations are always at least as good as the corresponding ones discussed in Sections 2-3 if the same set of valid inequalities is added. Indeed, if (x^*, X^*, y^*) is an optimal solution of one of the above relaxations, possibly reenforced through the addition of some of the valid inequalities discussed in Section 3, then (x^*, y^*) is a feasible solution for problem (8)-(11) reenforced with the same valid inequalities. Moreover, easy examples, such as the following one, can be built for which strict dominance holds.

Example 1 *Consider the graph $G = (V, E)$ with $V = \{1, \ldots, 10\}$ and*

$$E = \{12, 13, 14, 25, 26, 37, 38, 49, 4\text{-}10\},$$

which is a tree. Let $K = 2$. The optimal value for this problem is 39, obtained by deleting node 1 and any one of the three nodes 2, 3, 4. The relaxation (8)–(10) yields a bound equal to 40.5 for the root node of the branch-and-bound tree (note that for a tree the clique and cycle inequalities have no effect, while there is a single path inequality for each node pair), while both the RLT and semidefinite relaxation give a bound equal to 39.

We also notice that one might strengthen the valid inequalities introduced in Section 3. E.g., if we have the following subgraph (a cycle of length 4)

$$V' = \{i, j, k, p\} \qquad E' = \{ij, jk, kp, ip\},$$

then, it is easily seen that inequality (38)

$$y_{ik} + y_{jp} \leq x_i + x_j + x_k + x_p$$

can be replaced by the following pair of inequalities:

$$y_{ik} \leq x_i + x_k - X_{ik} + X_{jp}, \quad y_{jp} \leq x_j + x_p - X_{jp} + X_{ik}. \tag{53}$$

Indeed, to see that the first one holds, notice that nodes i and k are disconnected if we remove at least one of such nodes or if we remove both nodes j and p (similar for the second inequality). The first inequality can also be introduced if edge jp is in the graph, while the second one would not need to be added in this case, because it is already implied by constraint $y_{jp} = x_j + x_p - X_{jp}$. Similarly, the second inequality can also be introduced if we add edge ik, while the first one does not need to be added.

5 Computational experiments

In order to test our approaches, we considered two different classes of test instances. The first class of instances consists of Barabasi graphs (see the generator [7]) with 100 nodes and, respectively, 194, 285 and 380 edges. Such instances are denoted by `Barabasixx` where `xx` is the number of edges. The second class consists of random graphs with 50 and 100 nodes and density equal to 0.1, 0.2 and 0.3. Such instances are denoted by `Randxx-yy` where `xx` is the number of nodes and `yy` is the density. For each graph type we considered ten instances and values of K equal to $0.1N, 0.2N, 0.3N, 0.4N$ (recall that N is the number of nodes). For each set of instances we report:

- the average computation times $\mathrm{T_{avg}}$;
- the average gap $\mathrm{GAP_{avg}}$, where the gap for an instance is equal to

$$100 \cdot \frac{\texttt{Final Upper Bound} - \texttt{Best}}{\texttt{Final Upper Bound}}\%,$$

  where `Best` is the best observed value for the instance;
- the number of successes within the imposed time limit equal to 1,000 seconds, denoted by #OPT.

All tests have been performed by running `CPLEX` 12.1 over a `Xeon` processor at 2.33 GHz with 8 GB RAM.

We performed three different sets of experiments. In the first set we compared the basic version of our model (only including the path inequalities (10)), from now on denoted by `Base`, with the model (1)–(7) proposed in [2]. Constraints (4)–(6) were efficiently handled, in our implementation, as lazy constraints, separating them like cuts. Though we only considered the Barabasi graphs, the results, reported in Table 1 for the model proposed in [2] and in Table 2 for the `Base` version, appear to be quite clear: the basic model seems to perform better than the model proposed in [2] in terms of average computation time and/or average gap. It is worthwhile to remark that instances which turned out to be challenging for one model were also challenging for the other model.

In the second set of experiments we compared the `Base` version with the versions with the additional 4-cycle inequalities (38) (denoted by `Base+4cyc`), with the additional 4-clique inequalities (32) with $q = 4$ and $t = 2, 3$ (denoted by `Base+4clq`), and with both the 4-cycle and 4-clique inequalities (denoted by `Base+4cyc+4clq`). All results are reported in Tables 2–5. From the tables we can observe the following.

- Overall the impact of the 4-clique inequalities appears to be quite mild. The versions `Base` and `Base+4clq` behave very similarly, while the versions `Base+4cyc` and `Base+4cyc+4clq` also behave very similarly with the exception of the `Rand100-0.3` instances, where the latter performs much worse. In fact, looking at the behavior of `Base+4cyc+4clq` over such instances, we could observe that the bad behavior was due to the fact that this version was even unable to conclude the bound computation for the root node within the time limit.
- The versions `Base` and `Base+4cyc` behave very similarly over the `Barabasi` instances. The only significative improvement of `Base+4cyc` with respect to `Base` is in the computation times for the `Barabasi380` instance with $K = 40$.

- The improvement of version `Base+4cyc` with respect to `Base` becomes clear over the `Randxx-yy` instances. We can also observe that such improvement becomes more evident as the density increases (the only exception is the case `Rand50-0.3` with $K = 5$). A possible explanation for this is that as the density increases, the number of 4-cycles also grows. This makes the model heavier (as also confirmed by the decrease of the number of branch-and-bound nodes explored) but the bounds much tighter.

We point out that at each node of the branch-and-bound tree only violated path, 4-cycle and 4-clique inequalities, detected through proper separation procedures, have been inserted. The separation procedure for the path inequalities is based on the solution of a shortest path problem, as already pointed out in Section 2, while the separation procedure for the 4-cycle and 4-clique inequalities is based on a simple enumeration.

We also performed tests with 3-clique inequalities (but their impact was quite mild), while we did not performed experiments with cliques of cardinality larger than 4. Moreover, as already commented in Section 3.3, we also tested the neighborhood inequalities (as well as other inequalities not described here), but these did not appear to be very effective.

Finally, in the third set of experiments we compared the `Base+4cyc` version with the version based on the RLT relaxation with the path inequalities strengthened through inequalities (52), obtained by solving maximum spanning tree subproblems, and with the additional 4-cycle inequalities strengthened by (53). Such version is simply denoted by `RLT`. We restricted the attention to the `Rand50` instances. In Table 6 we report the average number of nodes visited when using the two different approaches. We notice that this value is usually much lower for `RLT`, in particular as the density and K increase. Therefore the `RLT` relaxation appears to be much tighter with respect to the version `Base+4cyc`. On the other hand, the computation times are usually larger for the `RLT` version. This is a consequence of the fact that the additional X_{ij} variables make the `RLT` relaxations much more expensive. The results for the `RLT` version are reported in Table 7. Overall, `Base+4cyc` is clearly superior with respect to `RLT` in terms of number of successes, average gaps and average times. However, in spite of the much lower number of visited nodes, approach `RLT` tends to become better with respect to `Base+4cyc` when we increase the density and the value of K. For $K = 15$ and density equal to 0.3, `RLT` obtains three successes with respect to the zero successes of `Base+4cyc`, and the average gap for `RLT` is clearly lower than that for `Base+4cyc`. For $K = 20$ and density equal to 0.3, both `Base+4cyc` and `RLT` have no success within the time limit, but again the average gap for `RLT` is clearly lower than that for `Base+4cyc`. Thus, it seems that `RLT` becomes advisable as soon as the density of the graph and the value of K increase.

For what concerns the SDP relaxation, we have only performed a few preliminary experiments with the solver `CSDP` [4] through `YALMIP`, from which we could observe that the bound computation was quite slow even on relatively small graphs. We did not further investigate such relaxation but this could be an issue for future research. A possible approach is that of using SDP-cuts to take advantage of the tightness offered by the SDP relaxations, while maintaining manageably-sized LP relaxations (see, e.g., [14, 17]). As another possible direction for future developments, it might be convenient to use the cheaper but weaker relaxations of Sections 2-3 during the first iterations of the branch-and-bound approach, and to use the more expensive but stronger relaxations of Section 4 at later stages.

Finally, we have also performed a few tests with model (24)–(29). Similarly to what we did for model (1)–(7), the large set of constraints (16)–(18) was efficiently handled as a set of lazy constraints, by separation. Such model did not compare favorably with respect to the other two models considered. In a preliminary testing, it was able to optimally solve a batch of ten instances with $|V| = 30$, $K = 4$ and density 10% within 324 seconds on average (530 in the worst case). Its performances worsened on instances with density 20%, where it could solve at optimality only 4 out of 10 instances within a time limit of 1000 seconds. In both cases the model was easily outperformed by model (1)–(7), that was able to solve at optimality all such instances with computation times usually below 30 seconds. In view of these results we did not pursue investigation on the bilevel approach.

Acknowledgments

The authors would like to thank two anonymous referees, whose constructive comments helped to improve the initial version of the paper.

Instance	$K = 0.1N$			$K = 0.2N$			$K = 0.3N$			$K = 0.4N$		
	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT
`Barabasi194`	1000.0	37.706	0	369.3	0.15093	8	2.76	0	10	0.5	0	10
`Barabasi285`	1000.0	59.564	0	1000.0	25.781	0	777.2	0.74527	3	4.5	0	10
`Barabasi380`	1000.0	63.36	0	1000.0	44.821	0	1000.0	10.48	0	505.9	0.071781	7

Table 1: Average time, average gap and number of successes within the time limit over all the instances for the model in [2].

Instance	$K = 0.1N$			$K = 0.2N$			$K = 0.3N$			$K = 0.4N$		
	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT	$\mathrm{T_{avg}}$	$\mathrm{GAP_{avg}}$	#OPT
`Barabasi194`	724.97	10.978	4	1.488	0	10	0.302	0	10	0.05	0	10
`Barabasi285`	1000.1	27.906	0	1000.1	19.043	0	127.3	0.17389	9	0.601	0	10
`Barabasi380`	1000.2	27.434	0	1000.1	34.069	0	1000.1	8.2691	0	17.746	0	10
`Rand50-0.1`	34.52	0	10	338.65	0	10	605.6	3.6413	6	43.786	0	10
`Rand50-0.2`	54.637	0	10	888.04	2.5521	5	1000.1	14.595	0	1000	16.552	0
`Rand50-0.3`	1000.3	1.2605	0	758.5	0.80965	8	1000.2	13.588	0	1000.1	18.913	0
`Rand100-0.1`	1000.3	26.44	0	909.25	32.511	0	1000.2	38.936	0	1000.2	27.596	0
`Rand100-0.2`	1000.2	26.111	0	1000.2	31.466	0	1000.2	35.681	0	1000.1	31.875	0
`Rand100-0.3`	1000.1	21.202	0	1000.1	29.881	0	1000.1	34.668	0	1000.1	31.189	0

Table 2: Average time, average gap and number of successes within the time limit over all the instances for the `Base` version.

Instance	$K = 0.1N$			$K = 0.2N$			$K = 0.3N$			$K = 0.4N$		
	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT
`Barabasi194`	729.39	11.184	4	1.532	0	10	0.308	0	10	0.052	0	10
`Barabasi285`	1000.1	27.505	0	1000.1	18.5	0	123	0.18584	9	0.539	0	10
`Barabasi380`	1000.5	27.411	0	1000.1	32.577	0	957.96	7.696	1	8.655	0	10
`Rand50-0.1`	37.894	0	10	295.21	0	10	562.43	3.19	6	14.796	0	10
`Rand50-0.2`	27.995	0	10	452.37	0.33984	9	976.32	6.8859	1	1000	10.532	0
`Rand50-0.3`	1000.1	2.4896	0	289.05	0.020619	9	1000.1	5.5766	0	1000.1	10.048	0
`Rand100-0.1`	1000.7	22.732	0	909.27	28.182	0	1000.2	34.204	0	1000.2	25.225	0
`Rand100-0.2`	906.53	5.1262	3	1000.2	9.941	0	1000.2	15.13	0	1000.1	20.186	0
`Rand100-0.3`	1000.7	4.5455	0	1000.6	14.118	0	1000.6	14.674	0	1000.3	19.567	0

Table 3: Average time, average gap and number of successes within the time limit over all the instances for the `Base+4cyc` version.

Instance	$K = 0.1N$			$K = 0.2N$			$K = 0.3N$			$K = 0.4N$		
	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT
`Barabasi194`	718.43	10.891	4	1.467	0	10	0.309	0	10	0.057	0	10
`Barabasi285`	1000.1	27.046	0	1000.1	19.082	0	126.93	0.17238	9	0.612	0	10
`Barabasi380`	1000.1	31.533	0	1000.1	34.252	0	1000.1	8.3262	0	21.492	0	10
`Rand50-0.1`	35.536	0	10	321.45	0	10	594.09	3.5791	6	44.113	0	10
`Rand50-0.2`	69.811	0	10	904.82	2.3058	4	1000.1	15.289	0	1000.1	16.162	0
`Rand50-0.3`	1000.1	1.3834	0	809.91	1.6337	7	1000.1	13.02	0	1000.1	18.901	0
`Rand100-0.1`	1000.2	28.989	0	1000.1	35.721	0	1000.2	38.935	0	1000.2	27.577	0
`Rand100-0.2`	1000.1	26.09	0	1000.2	31.466	0	1000.2	35.447	0	1000.1	31.145	0
`Rand100-0.3`	1000.1	23.534	0	1000.1	29.883	0	1000.2	34.303	0	1000.1	29.281	0

Table 4: Average time, average gap and number of successes within the time limit over all the instances for the `Base+4clq` version.

Instance	$K = 0.1N$			$K = 0.2N$			$K = 0.3N$			$K = 0.4N$		
	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT
`Barabasi194`	754.37	12.058	4	1.646	0	10	0.313	0	10	0.057	0	10
`Barabasi285`	1000.2	29.113	0	1000.1	18.86	0	117.93	0.18833	9	0.56	0	10
`Barabasi380`	1000.1	31.074	0	1000.1	32.98	0	996.5	7.6364	1	12.419	0	10
`Rand50-0.1`	36.804	0	10	275.68	0	10	557.13	3.1299	6	14.804	0	10
`Rand50-0.2`	32.214	0	10	454.16	0.21249	9	1000.1	7.3819	0	1000.1	10.407	0
`Rand50-0.3`	1000.5	2.4139	0	341.57	0.020619	9	1000	5.019	0	1000	9.075	0
`Rand100-0.1`	1000.1	24.369	0	1000.1	31.005	0	1000.2	34.146	0	1000.2	25.188	0
`Rand100-0.2`	945.99	5.5417	1	1000.3	10.077	0	1000.2	15.073	0	1000.1	19.486	0
`Rand100-0.3`	1000.7	11.584	0	1000.8	27.703	0	1000.9	36.487	0	1000.6	31.757	0

Table 5: Average time, average gap and number of successes within the time limit over all the instances for the `Base+4cyc+4clq` version.

	Base+4cyc				RLT			
Instance	$K = 0.1N$	$K = 0.2N$	$K = 0.3N$	$K = 0.4N$	$K = 0.1N$	$K = 0.2N$	$K = 0.3N$	$K = 0.4N$
`Rand50-0.1`	103	708	1672	460	78	284	445	458
`Rand50-0.2`	181	2240	2085	1952	72	254	173	191
`Rand50-0.3`	13371	4669	4808	3472	203	133	104	94

Table 6: Average number of nodes visited over the `Rand50` instances for the `Base+4cyc` and `RLT` versions

Instance		K = 0.1N			K = 0.2N			K = 0.3N			K = 0.4N	
	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT	T_{avg}	GAP_{avg}	#OPT
`Rand50-0.1`	155.1	0	10	753.2	2.65	4	734.5	5.34	3	141.3	0	10
`Rand50-0.2`	117.3	0	10	885.3	3.9	4	1000	8.74	0	1000	10.85	0
`Rand50-0.3`	1000	3.43	0	613.6	0.32	5	891.4	3.46	3	1000	6.86	0

Table 7: Average time, average gap and number of successes within the time limit over the `Rand50` instances for the `RLT` version.

References

[1] K. M. Anstreicher. Semidefinite programming versus the reformulation-linearization technique for nonconvex quadratically constrained quadratic programming. *Journal of Global Optimization*, 43:471–484, 2009.

[2] A. Arulselvan, C. W. Commander, L. Elefteriadou, and P. M. Pardalos. Detecting critical nodes in sparse graphs. *Computers & Operations Research*, 36:2193–2200, 2009.

[3] V. Boginski and C. W. Commander. Identifying critical nodes in protein-protein interaction networks. In S. Butenko, W. Art Chaovalitwongse, and P. M. Pardalos, editors, *Clustering challenges in biological networks*, pages 153–167. World Scientific, 2009.

[4] B. Borchers. CSDP, a C library for semidefinite programming. *Optimization Methods and Software*, 11:613–623, 1999.

[5] T. Christof. PORTA —a POlyhedron Representation Transformation Algorithm. Free software. Revised by A. Löbel.

[6] T. N. Dinh, Y. Xuan, M. T. Thai, E. K. Park, and T. Znati. On approximation of new optimization methods for assessing network vulnerability. In *Proceedings of the 29th IEEE Conference on Computer Communications (INFOCOM)*, pages 105–118, 2010.

[7] D. Dreier. Barabasi graph generator v1.4. `http://www.cs.ucr.edu/ddreier`.

[8] N. Fan and P. M. Pardalos. Robust optimization of graph partitioning and critical node detection in analyzing networks. In *COCOA 2010*, pages 170–183, 2010.

[9] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer, 2000.

[10] L. Lovász. Graph theory and integer programming. *Annals of Discrete Mathematics*, 4:141–158, 1979.

[11] T. C. Matisziw and A. T. Murray. Modeling s-t path availability to support disaster vulnerability assessment of network infrastructure. *Computers and Operations Research*, 36:16–26, 2009.

[12] Y.-S. Myung and H.-J. Kim. A cutting plane algorithm for computing k-edge survivability of a network. *European Journal of Operational Research*, 156(3):579–589, 2004.

[13] H. D. Sherali and W. P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, 3:411–430, 1990.

[14] H. D. Sherali and B. M. Fraticelli. Enhancing RLT relaxations via a new class of semidefinite cuts. *Journal of Global Optimization*, 22:233–261, 2002.

[15] H. D. Sherali and C. H. Tuncbilek. A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. *Journal of Global Optimization*, 2:101–112, 1992.

[16] H. D. Sherali and C. H. Tuncbilek. A reformulation-convexification approach for solving nonconvex quadratic programming problems. *Journal of Global Optimization*, 7:1–31, 1995.

[17] H.D. Sherali, E. Dalkiran, and J. Desai. Enhancing RLT-based relaxations for polynomial programming problems via a new class of ν-semidefinite cuts. *Computational Optimization and Applications*, to appear, 2011.

[18] J. C. Smith and C. Lim. Algorithms for discrete and continuous multicommodity flow network interdiction problems. *IIE Transactions*, 39:15–26, 2007.

[19] R. Wollmer. Removing arcs from a network. *Operations Research*, 12:934–940, 1964.

[20] R. K. Wood. Deterministic network interdiction. *Mathematical and Computer Modelling*, 17:1–18, 1993.

Appendix

A Convex hull description

Ignoring the cardinality constraint $\sum_{i \in V} x_i \le K$, formulation (8)–(11) reads as follows:

$$\begin{aligned}
\max \quad & \sum_{\substack{i,j \in V \\ i<j}} y_{ij} & & \\
\text{subject to} \quad & \sum_{r \in V(P)} x_r \ge y_{ij} & i,j \in V,\ P \in \mathcal{P}(i,j),\ i<j & \quad (54)\\
& x_i, y_{ij} \in \{0,1\} & i,j \in V,\ i<j. & \quad (55)
\end{aligned}$$

When the underlying graph G is a path, the constraint matrix of system (54) is totally unimodular, as (ignoring unit columns) the 1s appear in consecutive positions on each row. Therefore in this case the linear relaxation of the above set provides a description of the convex hull of integer solutions. However, the linear relaxation of (54)–(55) has fractional extreme points already for $G = K_3$ or when G is a tree. In the following subsections we give a linear-inequality description of the convex hull of (54)–(55) for some special cases. We first discuss here the proof technique and give some results that will be used later.

First of all we observe that the convex hull of (54)–(55) is a full-dimensional polyhedron, as the following $N + N(N-1)/2 + 1$ feasible points are affinely independent (recall that $N = |V|$ and the number of variables is $N + N(N-1)/2$):

- for $i \in V$, the point defined by setting $x_i = 1$ and all other variables to 0;
- for $i, j \in V, i < j$, the point defined by setting $x_i = y_{ij} = 1$ and all other variables to 0;
- the origin.

The convex hull of (54)–(55) will be described by some candidate linear system $Ax + By \ge d$. In order to prove that this system actually describes the convex hull of (54)–(55), we will use a well-known technique due to Lovász [10]: first we will observe that $Ax + By \ge d$ is a valid relaxation for our set; then we will show that for every optimization problem of the form

$$\max \left\{ \sum_{i \in V} p_i x_i + \sum_{\substack{i,j \in V \\ i<j}} q_{ij} y_{ij} : (x,y) \text{ satisfies (54)–(55)} \right\} \quad (56)$$

with $(p,q) \ne \mathbf{0}$, there is an inequality in the system $Ax + By \ge d$ that is satisfied at equality by all the optimal solutions of problem (56). Since our set is full-dimensional, this proves the result, as when the objective function is parallel to a facet of the convex hull of (54)–(55), an inequality that is tight for all optimal solutions must induce that facet; thus all the facets of the convex hull of (54)–(55) appear in $Ax + By \ge d$.

We now discuss some preliminary results. In the following we assume that an objective function $(p,q) \ne \mathbf{0}$ is fixed. Also, expressions such as "optimal solution" or "optimal value" will always implicitly refer to problem (56).

Lemma 12 *If $p_i > 0$ for some $i \in V$, then all optimal solutions satisfy $x_i = 1$. If $q_{ij} < 0$ for some $i, j \in V, i < j$, then all optimal solutions satisfy $y_{ij} = 0$.*

Proof. If a feasible solution satisfies $x_i = 0$, we can increase x_i to 1: this gives a feasible solution with better objective value. Similarly, if a feasible solution satisfies $y_{ij} = 1$, we can decrease y_{ij} to 0. $\square$

Therefore, from now on we assume that $p \le \mathbf{0}$ and $q \ge \mathbf{0}$.

Note that if (x, y) is an optimal solution, then

$$y_{ij} = \min\left\{1, \min\{\textstyle\sum_{r\in V(P)} x_r : P \in \mathcal{P}(i,j)\}\right\} \tag{57}$$

for every ij such that $q_{ij} > 0$. A feasible solution (x, y) with y_{ij} satisfying (57) for *all* ij, (including those for which $q_{ij} = 0$), will be called a *standard solution.* Since a standard solution is uniquely determined by its x-components, when constructing a standard solution we will only specify x.

Lemma 13 *If $p = \mathbf{0}$ or $q = \mathbf{0}$, then one of the bounds $0 \le x_i \le 1$ is tight for all optimal solutions.*

Proof. If $p = \mathbf{0}$, then, since $(p, q) \ne \mathbf{0}$, there exist i, j such that $q_{ij} > 0$. We claim that all optimal solutions satisfy $y_{ij} = 1$. Assume this is not true, i.e., there is an optimal solution with $y_{ij} = 0$. Then the standard solution $x = \mathbf{1}$ has a larger objective value, a contradiction.

If $q = \mathbf{0}$, then, since $(p, q) \ne \mathbf{0}$, there exists i such that $p_i < 0$. We claim that all optimal solutions satisfy $x_i = 0$. Assume this is not true, i.e., there is an optimal solution with $x_i = 1$. Then the standard solution $x = \mathbf{0}$ has a larger objective value, a contradiction. $\square$

Lemma 14 *If $i \in V$ is the only node such that $p_i < 0$, then $x_i = 0$ for all optimal solutions.*

Proof. If a feasible solution satisfies $x_i = 1$, we can decrease x_i to 0 and set all other components of x to 1 (without changing y): this gives a better feasible solution. $\square$

Therefore, from now on we assume that $q \ne \mathbf{0}$ and $p_i < 0$ for at least two nodes.

For a given objective function as in (56), we denote by Q the set of pairs ij such that $q_{ij} > 0$. We also define graph $G_Q = (V, Q)$. The following result will be used several times.

Lemma 15 *Let $i \in V$ be a node of degree 1 in G_Q and let j be its unique neighbor in G_Q. If $p_i < 0$ and $ij \in E$, then all optimal solutions satisfy $x_i + x_j = y_{ij}$.*

Proof. Assume that there is an optimal solution (x, y) such that $x_i + x_j > y_{ij}$. Since $q_{ij} > 0$, it follows that $x_i = x_j = y_{ij} = 1$. Then we find a better solution by decreasing x_i to 0 and taking the corresponding standard solution. $\square$

We now consider two configurations in detail.

A.1 Clique with three nodes

Here we consider the case $G = K_3$ and prove that the convex hull of (54)–(55) is obtained by adding the following inequality to the linear relaxation of (54)–(55):

$$y_{12} + y_{23} + y_{13} \le x_1 + x_2 + x_3 + 1. \tag{58}$$

Note that inequality (58) is valid, as it is a clique inequality.

Let $px+qy$ be an objective function satisfying the above assumptions. Recall that $|Q| \geq 1$. If $1 \leq |Q| \leq 2$, we can use Lemma 15 (the existence of an index i as required in Lemma 15 is guaranteed by the fact that $p_i < 0$ for at least two nodes). So we only have to consider the case $|Q| = 3$, i.e., $q_{ij} > 0$ for all ij.

If $q_{ij} > 0$ for all ij, every optimal solution is a standard solution. Thus we have eight candidates, one for each possible choice of $x \in \{0,1\}^3$. It can be checked that the only candidates for which (58) is not tight are $x = \mathbf{0}$ and $x = \mathbf{1}$. However, $x = \mathbf{1}$ cannot be optimal, as by choosing an index i such that $p_i < 0$ and decreasing x_i to 0, a feasible solution with larger objective value is obtained. It follows that if $x = \mathbf{0}$ is not optimal, then all optimal solutions satisfy (58) at equality.

It only remains to consider the case in which $x = \mathbf{0}$ is an optimal solution. Clearly in this case the optimal value is 0. If all optimal solutions satisfy $x_1 + x_2 = y_{12}$, the proof is complete. So we assume that there is an optimal solution such that $x_1 + x_2 > y_{12}$. The only candidate with this property is the standard solution $x = (1,1,0)$. The optimality of this solution implies

$$p_1 + p_2 + q_{12} + q_{23} + q_{13} = 0. \tag{59}$$

Now, if we consider the two standard solutions $x = (1,0,0)$ and $x = (0,1,0)$, since their objective values cannot exceed 0, we have:

$$p_1 + q_{12} + q_{13} \leq 0, \quad p_2 + q_{12} + q_{23} \leq 0.$$

If we take the sum of these two inequalities and subtract equation (59), we obtain $q_{12} \leq 0$, a contradiction.

A.2 Cycle of length 4

Let $V = \{1,2,3,4\}$ and $E = \{12,23,34,14\}$. We prove that in this case the convex hull of (54)–(55) is obtained by adding the following five inequalities to the linear relaxation of (54)–(55):

$$y_{ij} + y_{jk} + y_{ik} \leq x_1 + x_2 + x_3 + x_4 + 1, \quad i,j,k \in V,\ i < j < k \tag{60}$$

$$y_{13} + y_{24} \leq x_1 + x_2 + x_3 + x_4. \tag{61}$$

The validity of (60) and (61) has been already proven in Section 3.2.

Lemma 16 *Suppose that $p_i < 0$ for exactly two nodes (i and j, say). (a) If $ij \in E$, then all optimal solutions satisfy $x_i + x_j = y_{ij}$. (b) If $ij \notin E$, then all optimal solutions satisfy $x_i = 0$.*

Proof. (a) Without loss of generality, $i = 1$ and $j = 2$. If there is an optimal solution (x,y) such that $x_1 + x_2 > y_{12}$, then at least one of x_1, x_2 is equal to 1, say $x_1 = 1$. Note that $y_{12} = 1$ only if $x_2 = 1$. Then we obtain a better solution by decreasing x_1 to 0 and setting (or leaving) x_3 and x_4 to 1 (other components unchanged).

(b) Without loss of generality, $i = 1$ and $j = 3$. If there is an optimal solution (x,y) with $x_1 = 1$, then the standard solution $x = (0,1,0,1)$ would have a better objective value. □

Therefore, from now on we assume that $p_i < 0$ for at least three nodes.

Lemma 17 *If there is an optimal solution with more than two x-components equal to 1, then there exists $ij \in E$ such that all optimal solutions satisfy $x_i + x_j = y_{ij}$.*

Proof. First of all note that no optimal solution satisfies $x = \mathbf{1}$ (otherwise one can obtain a better solution by choosing an index i such that $p_i < 0$ and decreasing x_i to 0).

It is easy to see that there cannot be an optimal solution with three x-components equal to 1 if $p_i < 0$ for all i. Thus we assume that $p_i = 0$ for some i, say $p_1 = 0$ without loss of generality (thus $p_2, p_3, p_4 < 0$).

Now we claim that if $q_{23} > 0$ then all optimal solutions satisfy $x_2 + x_3 = y_{23}$. If this is not true, there is an optimal solution with $x_2 = x_3 = y_{23} = 1$. But then the standard solution $x = (1, 0, 1, 0)$ would have a better objective value.

It remains to consider the case in which $p_1 = q_{23} = 0$ and there is an optimal solution with three x-components equal to 1. Such a solution must satisfy $x_1 = x_2 = x_4 = 1$. However, we obtain a better feasible solution by decreasing x_2 to 0 and setting (or leaving) y_{23} to 0. □

Therefore, from now on we assume that all optimal solutions have at most two x-components equal to 1.

Lemma 18 *If $q_{13} > 0$ and $q_{24} > 0$, then all optimal solutions satisfy* (61) *at equality.*

Proof. Since $q_{13} > 0$ and $q_{24} > 0$, for every optimal solution the components y_{13}, y_{24} satisfy (57), i.e., their value is maximal. Now it can be checked that every feasible solution with at most two x-components equal to 1 and with y_{13}, y_{24} maximal satisfies (61) at equality. □

Therefore, from now on we assume that $q_{13}q_{24} = 0$. We now analyze several cases, depending on the cardinality of Q. Note that $1 \leq |Q| \leq 5$.

Case 1: $|Q| = 1$. Let us assume that $Q = \{ij\}$. If $ij \in E$, we can apply Lemma 15 (as at least one of p_i, p_j is negative). So we assume that $ij \notin E$, say $ij = 13$ without loss of generality. Since, by our assumptions, at least three components of p are negative, we can assume without loss of generality that $p_1, p_2 < 0$. We claim that then $x_1 + x_2 + x_3 = y_{13}$ for all optimal solutions. Assuming by contradiction that this is not true, consider an optimal solution such that $x_1 + x_2 + x_3 > y_{13}$. Note that since $q_{13} > 0$, variable y_{13} satisfies condition (57). We now distinguish two possibilities.

If $y_{13} = 0$, then $x_1 = x_3 = 0$ (by (57)) and $x_2 = 1$. Then we can obtain a better feasible solution by decreasing x_2 to 0 and taking the corresponding standard solution.

If $y_{13} = 1$, at least two of x_1, x_2, x_3 are equal to 1. If $x_2 = 1$, we can decrease x_2 and take the corresponding standard solution; otherwise we have $x_1 = x_3 = 1$ and we can decrease x_1 and take the corresponding standard solution. In both cases we obtain a better feasible solution.

Case 2: $|Q| = 2$. If Q is either a maximum matching with both edges in E or a path of length 2 with both edges in E, we can apply Lemma 15. Since $Q \neq \{13, 24\}$ (as $q_{13}q_{24} = 0$), the only remaining case is when Q is a path with one edge in E and the other edge not in E, say $Q = \{12, 13\}$ without loss of generality. If $p_1 < 0$, we can apply Lemma 15. So we assume $p_1 = 0$, which implies $p_2, p_3, p_4 < 0$. Now, proceeding exactly as in Case 1, one can show that $x_1 + x_2 + x_3 = y_{13}$ for all optimal solutions.

Case 3: $|Q| = 3$. The only case that cannot be treated with the above lemmas and assumptions is when Q is a 3-clique, say $Q = \{12, 23, 13\}$ without loss of generality.

If $p_4 = 0$, we claim that $x_1 + x_2 = y_{12}$ for every optimal solution. If this is not true, there is an optimal solution with $x_1 = x_2 = y_{12} = 1$. We obtain a better solution by decreasing x_1 to 0 (note that $p_1 < 0$) and setting (or leaving) x_4 to 1. So we now assume $p_4 < 0$.

Assume that the origin is an optimal solution. If all optimal solutions satisfy $x_1+x_2 = y_{12}$, the analysis of this case is complete. So we assume that there is an optimal solution such that $x_1 + x_2 > y_{12}$. Since, by our assumption, there is no optimal solution with more than two component of x equal to 1, all optimal solutions such that $x_1+x_2 > y_{12}$ satisfy $x = (1, 1, 0, 0)$, with objective value

$$p_1 + p_2 + q_{12} + q_{23} + q_{13} = 0. \tag{62}$$

Now, if we consider the two standard solutions $x = (1, 0, 0, 0)$ and $x = (0, 1, 0, 0)$, since their objective values cannot exceed 0, we have:

$$p_1 + q_{12} + q_{13} \le 0, \quad p_2 + q_{12} + q_{23} \le 0.$$

If we take the sum of these two inequalities and subtract equation (62), we obtain $q_{12} \le 0$, a contradiction.

Thus we now assume that the origin is not optimal. We claim that then all optimal solutions satisfy (60) at equality (with $i = 1, j = 2, k = 3$).

We first consider optimal solutions with exactly two x-components equal to 1. Up to symmetries, constraint (60) is not satisfied at equality only if $x_4 = 1$ and $x_2 = 0$. However, since one of x_1 and x_3 is equal to 1, we find a better feasible solution by decreasing x_4 to 0 and taking the corresponding standard solution.

We now consider optimal solutions with exactly one x-component equal to 1. Constraint (60) is not satisfied at equality only if $x_4 = 1$. However, in this case the origin would be a better solution.

Case 4: $4 \le |Q| \le 5$. The only cases that cannot be treated with the above lemmas and assumptions are (up to symmetries) the following: (a) $Q = \{12, 23, 34, 14\}$ and (b) $Q = \{12, 23, 34, 14, 13\}$. For the most part, these two cases can be analyzed together.

First of all we show that if $p_i = 0$ for some i, then $x_i = 1$ for all optimal solutions. Assume that $p_i = 0$ and there is an optimal solution with $x_i = 0$. Note that x can have at most one x-component equal to 1: otherwise, by increasing x_i we would obtain an optimal solution with more than two x-components equal to 1, a contradiction to our assumptions. So there is at most one x-component equal to 1. Now, if we increase x_i to 1 and construct the corresponding standard solution, we obtain a better solution (as at least one y_{ij} with $q_{ij} > 0$ can be increased from 0 to 1), a contradiction. Therefore, from now on we assume that $p_i < 0$ for all $i \in V$.

We will show that there is $ij \in E$ such that all optimal solutions satisfy $x_i + x_j = y_{ij}$. The proof is by contradiction, so from now on we assume that for every $ij \in E$ there is an optimal solution such that $x_i + x_j > y_{ij}$.

Note that there is always a standard solution that is optimal. In the following we consider all possible standard solutions. Recall that we can assume that the number of x-components equal to 1 is at most two in any optimal solution.

Assume that the standard solution $x = \mathbf{0}$ is optimal (thus the optimal value is 0). Take an optimal solution with $x_1 + x_2 > y_{12}$. Such a solution satisfies $x_1 = x_2 = 1$, thus

$$p_1 + p_2 + q_{12} + q_{13} + q_{14} + q_{23} = 0. \tag{63}$$

If we consider the two standard solutions $x = (1, 0, 0, 0)$ and $x = (0, 1, 0, 0)$, we have $p_1 + q_{12} + q_{13} + q_{14} \le 0$ and $p_2 + q_{12} + q_{23} \le 0$. If we sum these two inequalities and subtract equation (63), we find $q_{12} \le 0$, a contradiction.

Assume that the standard solution $x = (1, 0, 0, 0)$ is optimal. Then the optimal value is $\alpha = p_1 + q_{12} + q_{13} + q_{14}$. Take an optimal solution such that $x_1 + x_2 > y_{12}$: its objective value is $p_1 + p_2 + q_{12} + q_{13} + q_{14} + q_{23} = \alpha$, hence $p_2 + q_{23} = 0$. Similarly, considering an optimal solution such that $x_1 + x_4 > y_{14}$, we find $p_4 + q_{34} = 0$. Now, if we consider the standard solution $x = (0, 1, 0, 1)$, we have $p_2 + p_4 + q_{12} + q_{13} + q_{14} + q_{23} + q_{34} \le \alpha$, which, together with the conditions obtained above, gives $p_1 \ge 0$, a contradiction.

Note that the above case also covers the situation in which the standard solution $x = (0, 0, 1, 0)$ is optimal (by symmetry arguments); moreover, if $13 \notin Q$, the above analysis also covers the cases in which $x = (0, 1, 0, 0)$ and $x = (0, 0, 0, 1)$ are optimal. However, if $13 \in Q$ these two cases need a different argument. So let us assume that $13 \in Q$ and the standard solution $x = (0, 1, 0, 0)$ is optimal (the other case is analogous). Then the optimal value is $\alpha = p_2 + q_{12} + q_{23}$. Take an optimal solution such that $x_1 + x_4 > y_{14}$: its objective value is $p_1 + p_4 + q_{12} + q_{13} + q_{14} + q_{34} = \alpha$, hence $p_1 + p_4 + q_{13} + q_{14} + q_{34} = p_2 + q_{23}$. Now, if we consider the standard solution $x = (0, 1, 0, 1)$, we find $p_2 + p_4 + q_{12} + q_{13} + q_{14} + q_{23} + q_{34} \le \alpha$, thus $p_2 + q_{23} \le p_1$ also holds. If we consider the standard solution $x = (1, 0, 0, 0)$, we find $p_1 + q_{12} + q_{13} + q_{14} \le \alpha$, thus $p_1 + q_{13} + q_{14} \le p_2 + q_{23} \le p_1$. We then obtain $q_{13} + q_{14} \le 0$, a contradiction.

Assume that the standard solution $x = (1, 1, 0, 0)$ is optimal. Then the optimal value is $\alpha = p_1 + p_2 + q_{12} + q_{13} + q_{14} + q_{23}$. Take an optimal solution such that $x_1 + x_4 > y_{14}$: its objective value is $p_1 + p_4 + q_{12} + q_{13} + q_{14} + q_{34} = \alpha$, hence $p_4 + q_{34} = p_2 + q_{23}$. Now, if we consider the standard solution $x = (0, 1, 0, 1)$, we find $p_4 + q_{34} \le p_1$. Finally, with $x = (1, 0, 0, 0)$ we find $p_2 + q_{23} \ge 0$. The last three conditions together imply $p_1 \ge 0$, a contradiction. Note that this case also covers the situation in which one of the following standard solutions is optimal: $(0, 1, 1, 0)$, $(0, 0, 1, 1)$, $(1, 0, 0, 1)$.

Assume that the standard solution $x = (1, 0, 1, 0)$ is optimal. Then the optimal value is $\alpha = p_1 + p_3 + q_{12} + q_{13} + q_{14} + q_{23} + q_{34}$. Take an optimal solution such that $x_1 + x_2 > y_{12}$: its objective value is $p_1 + p_2 + q_{12} + q_{13} + q_{14} + q_{23} = \alpha$, hence $p_2 = p_3 + q_{34}$. Similarly, if we consider an optimal solution such that $x_3 + x_4 > y_{34}$, we find $p_4 = p_1 + q_{12}$. Now, if we take the standard solution $x = (0, 1, 0, 1)$, we find $p_2 + p_4 \le p_1 + p_3$. The last three conditions together imply $q_{12} + q_{34} \le 0$, a contradiction.

If $13 \notin Q$, the above case also covers the situation in which $x = (0, 1, 0, 1)$ is optimal. It only remains to consider the case when $13 \in Q$ and the standard solution $x = (0, 1, 0, 1)$ is optimal. In this case the optimal value is $\alpha = p_2 + p_4 + q_{12} + q_{13} + q_{14} + q_{23} + q_{34}$. Take an optimal solution such that $x_1 + x_2 > y_{12}$: its objective value is $p_1 + p_2 + q_{12} + q_{13} + q_{14} + q_{23} = \alpha$, hence $p_1 = p_4 + q_{34}$. Similarly, if we consider an optimal solution such that $x_1 + x_4 > y_{14}$, we find $p_1 = p_2 + q_{23}$. Now, with the standard solution $x = (1, 0, 0, 0)$, we find $p_1 \le p_2 + p_4 + q_{23} + q_{34}$. The last three conditions together imply $p_1 \ge 0$, a contradiction.