The Quadratic Cycle Cover Problem: special cases and efficient bounds

Frank de Meijer * Renata Sotirov †

Abstract

The quadratic cycle cover problem is the problem of finding a set of node-disjoint cycles visiting all the nodes such that the total sum of interaction costs between incident arcs is minimized. In this paper we study the linearization problem for the quadratic cycle cover problem and related lower bounds.

In particular, we derive various sufficient conditions for the quadratic cost matrix to be linearizable, and use these conditions to compute bounds. We also show how to use a sufficient condition for linearizability within an iterative bounding procedure. In each step, our algorithm computes the best equivalent representation of the quadratic cost matrix and its optimal linearizable matrix with respect to the given sufficient condition for linearizability. Further, we show that the classical Gilmore-Lawler type bound belongs to the family of linearization based bounds, and therefore apply the above mentioned iterative reformulation technique. We also prove that the linearization vectors resulting from this iterative approach satisfy the constant value property.

The best among here introduced bounds outperform existing lower bounds when taking both quality and efficiency into account.

Keywords: quadratic cycle cover problem, linearization problem, equivalent representations, Gilmore-Lawler bound

1 Introduction

A cycle cover in a directed graph is a set of node-disjoint cycles such that every node is on exactly one cycle. The quadratic cycle cover problem (QCCP) is the problem of finding a cycle cover in a graph such that the total sum of interaction costs between incident arcs is minimized. This problem is proven to be \text{NP}-hard, see [18]. The corresponding linear problem is called the cycle cover problem (CCP), in which one wants to find a minimum cycle cover with respect to linear arc costs. It is well known that the CCP is solvable in polynomial time.

In literature several special cases with respect to the objective function of the QCCP are considered. In the angular metric cycle cover problem (ANGLE-CCP) the quadratic costs represent the change of the direction induced by two arcs that are incident. The goal of ANGLE-CCP is to find a cycle cover of the graph while minimizing the total angular costs. The ANGLE-CCP has applications in robotics, see [3]. In the same paper it is shown that ANGLE-CCP is \text{NP}-hard. Only recently Galbiati et al. [13] introduced another special case of the QCCP: the minimum reload cost cycle cover problem (MinRC3). The MinRC3 problem asks for a minimum cycle cover in an arc-colored graph under the reload cost model. A reload cost is an interaction cost that is paid when two arcs of different colors are placed in succession on a cycle. The goal of the MinRC3 problem is to find a cycle cover such that the total reload cost is minimized. The problem is proven to be \text{NP}-hard in the strong

*CentER, Department of Econometrics and OR, Tilburg University, The Netherlands, f.j.j.demeijer@uvt.nl
†Department of Econometrics and OR, Tilburg University, The Netherlands, r.sotirov@uvt.nl
The notion of reload costs is introduced by Wirth and Steffan [34], and it has many applications in various fields, e.g. in cargo, energy and telecommunication networks, see e.g. [3, 34]. A detailed overview of the MinRC3 problem and its applications can be found in [6]. Several other combinatorial optimization problems including these reload costs have been investigated, see [6, 12, 14, 17, 34].

The QCCP is closely related to the quadratic traveling salesman problem (QTSP) which is introduced in [22]. The QTSP is the problem of finding a Hamiltonian cycle in a graph minimizing a quadratic cost function. It has applications in bioinformatics, robotics and telecommunication, see [19]. When we remove the subtour elimination constraints, the QTSP boils down to QCCP. Therefore, the QCCP is often used to provide lower bounds for the QTSP, see e.g., [10, 22, 33]. For this reason, the quadratic cycle cover problem is an interesting optimization problem that has received more attention in the past few years.

Several papers have been written about solution methods for the QCCP or its related problems. Jäger and Molitor [22] introduced the QCCP in order to use the QCCP bounds as lower bounds in a branch-and-bound algorithm for the QTSP. Staněk et al. [33] use the QCCP in combination with a rounding procedure to construct heuristics for the QTSP. Fischer [18] studies the polyhedral properties of the QCCP by proving that some triangle inequalities are facet-defining. Galbiati et al. [13] derive various integer programming formulations for the MinRC3 problem. They exploit one of those formulations together with a column generation approach to compute lower bounds for the original problem. Moreover, in [13] a local search algorithm based on 2-exchange and 3-exchange neighbourhoods is constructed to compute upper bounds for the MinRC3 problem. Büyükoğlan et al. [6] study the MinRC3 problem on complete graphs with an equitable or nearly equitable 2-edge coloring. For these types of graphs (except some special cases) a polynomial time algorithm is derived that constructs a monochromatic cycle cover.

We focus here on the linearization problem of the QCCP and its applications. An instance of the quadratic cycle cover problem is called linearizable if there exists an instance of the linear cycle cover problem such that the associated costs for both problems are equal for all feasible cycle covers. The linearization problem of the quadratic cycle cover problem asks whether a given instance of the QCCP is linearizable. To the best of our knowledge, this is the first paper about the linearization problem of the QCCP.

In the past few years linearization problems have become an active field of research for many combinatorial optimization problems. Kabadi and Punnen [23, 28] study the linearization problem of the quadratic assignment problem (QAP) and provide polynomial time algorithms that solves it. In particular, they present in [23] (resp. [28]) an $O(n^4)$ (resp. $O(n^3)$) algorithm for the general (resp. Koopmans-Beckmann) QAP linearization problem, where $n$ is the size of the problem. The linearization problem for the quadratic minimum spanning tree problem and the quadratic traveling salesman problem are studied in [10] and [30], respectively. Hu and Sotirov [21] develop a polynomial time algorithm that solves the linearization problem of the quadratic shortest path problem on directed graphs.

**Main results and outline.** In this paper, we first provide an elegant and compact proof that the quadratic cycle cover problem is $NP$-hard. Then, we consider the linearization problem of the QCCP and derive various conditions for an instance of the QCCP to be linearizable. In particular, we provide three different types of weak sum conditions on the data matrix for which the corresponding instance can be solved in polynomial time. Further, we present a general framework in which each sufficient condition of linearizability can be used to construct a lower bound on the optimal objective value. Each of these bounds can be computed by a solving a linear programming problem, as long as the set of linearizable matrices is a polyhedron. These types of bounds are called linearization based bounds (LBB), and were recently introduced in [21] for general binary quadratic problems. However, our LBBs exploit sufficient conditions of linearizability suited for the QCCP.

Furthermore, we show how to use a sufficient condition of linearizability within an itera-
tive bounding procedure. In each iteration, we search for the best equivalent representation of the objective and its optimal linearizable matrix that satisfies a particular sufficient condition of linearizability. We refer to the resulting bound as the reformulation based bound (RBB). Our iterative bounding procedure can be seen as a generalization of similar iterative procedures.

Finally, we consider the classical Gilmore-Lawler (GL) type bound. First, we show that the GL type bound for the QCCP can be obtained by solving a single linear programming problem instead of solving \(m\) (integer) subproblems, where \(m\) equals the number of arcs in the graph. Then, we prove that the GL type bound belongs to the family of linearization based bounds by providing the appropriate sufficient condition. We implement our iterative bounding procedure to compute the RBB using the GL type bound. In literature, iterative approaches for various problems that are based on the GL type bounds use dual variables to obtain bounds, and do not search for equivalent reformulations that provide best bounds in each iteration. Clearly, our approach outperforms others. Another interesting result is that the linearization vectors resulting from this iterative procedure satisfy the constant value property. Yet another important property for linearizability.

Our numerical results show that the introduced bounding approaches are efficient and provide strong bounds compared to several classical methods from literature. In particular, our most prominent bound can be computed within 60 seconds for instances up to 15000 arcs. Interestingly, the GL type bound that is known to be one of the computationally cheapest bound for quadratic optimization problems cannot be computed on such large instances.

This paper is organized as follows. In Section 2, we formally introduce the QCCP, and prove its \(\mathcal{NP}\) hardness. In Section 3, the linearization problem for the QCCP is introduced and several sufficient conditions for linearizability are derived. The general framework for the computation of the linearization based bounds is discussed in Section 4. These bounds are used in Section 5 to construct an iterative bounding procedure for each sufficient condition. In Section 6, we consider the classical GL type bound and prove that it belongs to the family of linearization based bounds. We also show how the iterative procedure for this linearization based bound boils down to the computation of the strongest GL type bound in each step. In Section 7, we briefly discuss several other bounds from literature. Numerical results are given in Section 8.

**Notation**

A directed graph \(G = (N, A)\) is given by a node set \(N\) and an arc set \(A \subseteq N \times N\). For all nodes \(i \in N\) we denote by \(\delta^+(i)\) the set of arcs that are leaving \(i\). Similarly, \(\delta^-(i)\) denotes the set of arcs that are entering \(i\). For all arcs \(e \in A\) we let \(e^+\) and \(e^-\) denote the starting and ending node of \(e\), respectively.

For any square matrix \(M\), we introduce the operator \(\text{diag} : \mathbb{R}^{n \times n} \to \mathbb{R}^n\) that maps a matrix to a vector consisting of its diagonal elements. Moreover, we denote by \(\text{Diag} : \mathbb{R}^n \to \mathbb{R}^{n \times n}\) its adjoint operator. That is, for any \(v \in \mathbb{R}^n\) the matrix \(\text{Diag}(v)\) equals a diagonal matrix with the entries of \(v\) on its main diagonal.

### 2 The Quadratic Cycle Cover Problem

In this section, we formally introduce the quadratic cycle cover problem.

An instance \(I\) of the QCCP is specified by the pair \(I = (G, Q)\), where \(G = (N, A)\) is a directed graph with \(n\) vertices and \(m\) arcs and \(Q = (Q_{ef}) \in \mathbb{R}^{m \times m}\) is a quadratic cost matrix. The entries in \(Q\) are such that \(Q_{ef} = 0\) if \(f\) is not a successor of \(e\). In other words, the quadratic cost of two arcs \(e\) and \(f\) can be nonzero only if \(e\) and \(f\) are incident. In case we also consider linear arc costs, i.e. we have a cost function \(p : A \to \mathbb{R}\), we can put these
arc costs on the diagonal of the quadratic cost matrix. Therefore, we assume that the cost structure of an instance of the QCCP is fully determined by its quadratic cost matrix. Now let \( x \in \{0, 1\}^m \) be a vector with \( x_e = 1 \) if arc \( e \) belongs to a cycle cover, and 0 otherwise. Then the QCCP can be formulated as

\[
\text{OPT}(Q) := \min_{x} x^T Q x \\
\text{s.t. } x \in X,
\]

where \( X \) denotes the set consisting of all feasible cycle covers in \( G \), i.e.

\[
X := \left\{ x \in \{0, 1\}^m \mid \sum_{e \in \delta^+(i)} x_e = \sum_{e \in \delta^-(i)} x_e = 1, \forall i \in N \right\}.
\]

The above set equals the set of directed 2-factors in \( G \). For the existence of such a directed 2-factor in a directed graph, see e.g. Chiba and Yamashita [8].

The quadratic cycle cover problem is \( \mathcal{NP} \)-hard in the strong sense. This was already proven in [18]. Also, the \( \text{ANGLE-CCP} \) and the \( \text{MINRC3} \) problem are shown to be \( \mathcal{NP} \)-hard, see [3] and [13], respectively. We now provide an alternative proof which is based on a reduction from the quadratic assignment problem. We consider the Koopmans-Beckmann form of the QAP, see [24]. Let \( F \) and \( P \) be a set of \( n \) facilities and \( n \) locations, respectively, \( d : P \times P \to \mathbb{R} \) a distance function and \( w : F \times F \to \mathbb{R} \) a weight function. Without loss of generality, we assume that \( d_{ii} = w_{ii} = 0 \) for all \( i \in \{1, \ldots, n\} \). Then we search for a bijection \( \pi : F \to P \) such that \( \sum_{i=1}^{n} \sum_{j=1}^{n} d_{\pi(i) \pi(j)} w_{ij} \) is minimized.

**Theorem 1.** The QCCP is \( \mathcal{NP} \)-hard.

**Proof.** Let an instance \( I \) of QAP be given, i.e. we consider the sets \( P \) and \( F \) with \( |F| = |P| = n \), a distance function \( d : P \times P \to \mathbb{R} \), a weight function \( w : F \times F \to \mathbb{R} \) and a positive integer \( K \). We create an instance \( I' \) of QCCP.

Let us denote the set of facilities \( F \) by \( F = \{1, \ldots, n\} \) and the set of locations \( P \) by \( P = \{a, b, \ldots, n'\} \) where \( n' \) denotes the \( n \)th character. For the reduction we create a directed graph \( G = (N, A) \) that consists of cells. A cell belongs to a single facility and consists of \( n \) nodes, each of them corresponding to an assignment with one of the \( n \) locations. For each facility \( i \in F \), we define a set of \( n - 1 \) identical cells. Such a set we call a group. The nodes corresponding to the same assignment within a group are placed on a cycle. In this way we create \( n \) cycles per group. In Figure 1 the group corresponding to facility 1 is given together with the arcs between the nodes in the group.

![Figure 1: Group consisting of \( n - 1 \) cells corresponding to facility 1.](image)

We specify the interaction cost between each of the arcs within a group (the arcs drawn in Figure 1) to be zero for each group.

Since we have \( n \) facilities, we specify \( n \) such groups, each of them containing \( n - 1 \) cells. Let \( C'_j \) be the \( i \)th cell belonging to group \( j \). For each group we connect one cell to exactly
one cell from another group. In this way, each group is connected to each other group via a single cell, which results in \( \binom{n}{2} \) connections.

Connecting two cells is done by introducing a connection node and a relink node. Starting from the first cell, we draw an arc from every node in the first cell to the connection node. Successively, we draw an arc from the connection node to all the nodes of the second cell. The same is done for the relink node, now in the other direction. The connection between the cells \( C_1^1 \) and \( C_2^1 \) is shown in Figure 2. Note that in this figure the arcs between the nodes within a group are omitted.

The interaction cost between the arcs within a group (the ones drawn in Figure 1) and the arcs from and to the relink node (the ones drawn in Figure 2) is set to \( \infty \). We now determine the cost between arcs entering and leaving the connection node. Suppose we are considering the connected cells \( C_j^i \) and \( C_{j'}^{i'} \) with \( j \neq j' \). A node of \( C_j^i \) can be given by \((jk)\) with \( k \in P \). Similarly, a node of \( C_{j'}^{i'} \) can be given by \((j'k')\) with \( k' \in P \). Let \( e_{jk} \) be the arc from \((jk)\) to the connection node for all \( k \in P \) and let \( f_{j'k'} \) be the arc from the connection node to \((j'k')\) for all \( k' \in P \). Then the interaction cost \( Q_{e_{jk},f_{j'k'}} \) for all \( k, k' \in P \) is defined as follows:

\[
Q_{e_{jk},f_{j'k'}} := \begin{cases} 
  d_{k,k'} w_{j,j'} + d_{k',k} w_{j',j} & \text{if } k \neq k', \\
  \infty & \text{otherwise.}
\end{cases}
\]

The interaction cost between any other pair of successive arcs drawn in Figure 2 is set equal to zero. We repeat this construction for any two connected cells \( C_j^i \) and \( C_{j'}^{i'} \).

In total, we have \( \binom{n}{2} \) connections and, thus, \( \binom{n}{2} \) connection and \( \binom{n}{2} \) relink nodes. Figure 3 gives a simplified overview of the graph \( G \) when \( n = 4 \). The four groups are indicated by the large rectangles. Each group contains 3 cells. The arcs within a group are as given in Figure 1 but are omitted in Figure 3. The arcs to and from the connection node and relink node are drawn for the pair \( C_1^1 \) and \( C_2^1 \). For simplicity, the connections of all other pairs are indicated by a single arc.

![Figure 2: Connection between \( C_1^1 \) and \( C_2^1 \) via the connection and relink node.](image-url)
Note that the constructed graph $G$ has $n^2(n-1)+2\binom{n}{2} = \mathcal{O}(n^3)$ nodes and $n^2(n-1)+4\binom{n}{2} = \mathcal{O}(n^3)$ edges.

It remains to show that there exists a cycle cover in $G$ with cost at most $K$ if and only if there exists a feasible assignment with cost at most $K$.

First, it follows from the construction that a cycle cover in $G$ with finite cost corresponds to a feasible assignment of facilities and locations. Note that the cycles that contain a relink and a connection node never contain more than one node from each group, since this would induce an interaction cost of $\infty$. Therefore, these cycles that we call connection cycles from now on, consist of a relink node, a connection node and two nodes that each correspond to an assignment of a different facility. Moreover, these facilities must be assigned to a different location, otherwise this implies a cost of $\infty$. For the nodes in a cell that are not on a connection cycle there is only one possibility left to be covered, namely by a cycle within the same group, i.e. a cycle given in Figure 3. Consequently, all nodes on this cycle cannot be on a connection cycle. Therefore, in each group (i.e. each facility) exactly one location is ’chosen’ to be on a connection cycle. Moreover, no two facilities are assigned to the same location, since this would imply a cost of $\infty$ at the connection node connecting these groups. We conclude that a cycle cover with finite cost corresponds to a feasible assignment and vice versa.

Next, we show that the total cost of an assignment in the QAP instance equals the total cost of the corresponding cycle cover in the QCCP instance. By construction, the total cost of the cycle cover equals the sum of quadratic costs that is incurred at the $\binom{n}{2}$ connection nodes. For each pair of assigned locations and facilities, say $(jk)$ and $(j'k')$ with $j, j' \in F, j \neq j'$ and $k, k' \in P, k \neq k'$, this cost equals $d_{k,k'}w_{j,j'} + d_{k',k}w_{j',j}$. Taking the sum over all connection cycles, the total cost of the cycle cover equals $\sum_{i=1}^{n}\sum_{j=1}^{n}d_{\pi(i),\pi(j)}w_{i,j}$ where $\pi : F \to P$ is the bijection that corresponds to the assignment.

We conclude that there exists a feasible assignment for the QAP having cost at most $K$ if and only if there exists a feasible cycle cover in the corresponding instance of the QCCP having cost at most $K$. Thus, the QCCP is $\mathcal{NP}$-hard. \qed
3 The QCCP Linearization Problem

In this section, we formally introduce the linearization problem for the QCCP and derive various sufficient conditions for an instance of the QCCP to be linearizable. Several of these conditions are used later on to construct lower bounds for the optimal value of the QCCP.

Let us consider the (linear) cycle cover problem. Given a cost function $p : A \rightarrow \mathbb{R}$, the goal of the CCP is to find a cycle cover of minimum cost. It can be written as follows:

$$\min_{x \in \{0, 1\}^m} \left\{ p^T x \mid x \in X \right\},$$  \hspace{1cm} (3)

where $X$ is given in (2). Since the constraint set of $X$ is totally unimodular, it follows that the CCP is solvable in polynomial time. We call an instance $I = (G, Q)$ of the QCCP linearizable if there exists a cost vector $p \in \mathbb{R}^m$ such that $x^T Q x = p^T x$ for all cycle covers $x \in X$. If such a vector $p$ exists, we call $p$ a linearization vector of $Q$ for the QCCP.

The QCCP linearization problem can be stated as follows: Given an instance $I = (G, Q)$ of the QCCP, verify whether it is linearizable and, if yes, compute a linearization vector $p$ of $Q$.

In the remaining part of this section we provide sufficient conditions on the quadratic cost matrix $Q$ to be linearizable. The first type of sufficient conditions for linearizability are related to the constant value property (CVP) for cost vectors or cost matrices. The definition associated with the CCP is stated below.

**Definition 1.** A cost matrix $p$ satisfies the constant value property if $p^T x = p^T \bar{x}$ for all cycle covers $x, \bar{x} \in X$.

A similar definition holds for the quadratic version of the problem.

**Definition 2.** A cost matrix $Q$ satisfies the constant value property if $x^T Q x = \bar{x}^T Q \bar{x}$ for all cycle covers $x, \bar{x} \in X$.

When $Q$ satisfies the constant value property then $Q$ is linearizable, as stated by the following proposition.

**Proposition 1.** Assume that $Q$ satisfies the constant value property, i.e. $x^T Q x = \xi$ where $\xi \in \mathbb{R}$ for all $x \in X$, then $Q$ is linearizable with cost vector $p$ defined as $p_e = \xi/n$ for all $e \in A$.

**Proof.** For all $x \in X$ we have $x^T Q x = \xi = n \xi/n = x^T p$ since $x$ has exactly $n$ nonzero elements. \hfill \Box

A more restricted version of the CVP is obtained when the interaction cost of a single arc with its successor or predecessor is constant for all cycle covers $x \in X$. We refer to these properties as the row and column constant value property, respectively. These definitions are based on similar definitions by Punnen and Kabadi [30] for the QTSP.

**Definition 3.** A cost matrix $Q$ satisfies the row CVP if there exists some $b \in \mathbb{R}^m$ such that for all arcs $e \in A$ we have $Q_{ef} = Q_{eg} = b_e$ for all $f, g \in \delta^+(e^-)$ and $Q_{ef} = 0$ otherwise. A cost matrix $Q$ satisfies the column CVP if there exists some $c \in \mathbb{R}^m$ such that for all arcs $e \in A$ we have $Q_{fe} = Q_{ge} = c_e$ for all $f, g \in \delta^-(e^+_e)$ and $Q_{fe} = 0$ otherwise.

It is not hard to verify that an instance of the QCCP is linearizable if the cost matrix $Q$ satisfies the row or column CVP.

**Proposition 2.** If $Q$ satisfies the row CVP or the column CVP, then $Q$ is linearizable.
Proof. We prove the case when $Q$ satisfies the row CVP. Assume that $b \in \mathbb{R}^m$ is such that for all arcs $e \in A$, $Q_{ef} = Q_{eg} = b_e$ for all $f, g \in \delta^+(e^-)$ and $Q_{ef} = 0$ otherwise. Since $Q_{ef} = 0$ when $e$ and $f$ are not successors, we know that $x^T Q x = \sum_{e \in A} \sum_{f \in \delta^+(e^-)} Q_{ef} x_e x_f$. We have

$$\sum_{e \in A} \sum_{f \in \delta^+(e^-)} Q_{ef} x_e x_f = \sum_{e \in A} x_e b_e \sum_{f \in \delta^+(e^-)} x_f = \sum_{e \in A} x_e b_e = x^T b.$$ 

The proof for the column CVP is similar. 

A matrix $Q \in \mathbb{R}^{m \times m}$ is called a sum matrix if there exist $b, c \in \mathbb{R}^m$ such that $Q_{ef} = b_e + c_f$ for all $e, f$. A weak sum matrix is a matrix for which this property holds except for the entries on the diagonal, i.e. $Q_{ef} = b_e + c_f$ for all $e \neq f$. The weak sum property is used as a condition for linearizability for several quadratic problems, see e.g., [20, 30]. Since in this work we only incur a cost when two arcs are incident, we use a different form of the weak sum condition in which we only put a restriction on incident arcs. We call this condition the incident weak sum property.

Definition 4. A matrix $Q$ is called incident weak sum if there exist vectors $b, c \in \mathbb{R}^m$ such that $Q_{ef} = b_e + c_f$ for all $e, f$ and $Q_{ef} = 0$ otherwise, i.e. $Q_{ef} = b_e + c_f$ for all pairs of arcs $e, f$ such that $f$ is a successor of $e$. If such vectors $b$ and $c$ exist, these vectors are called supporting vectors of $Q$.

If the quadratic cost matrix $Q$ is an incident weak sum matrix, then $Q$ is linearizable as stated in the following proposition.

Proposition 3. Let $Q$ be an incident weak sum matrix with supporting vectors $b, c \in \mathbb{R}^m$. Then $Q$ is linearizable with cost vector $p = b + c$.

Proof. We show that for all $x \in X$ we have $x^T Q x = p^T x$ where $p = b + c$. Note that since $Q_{ef} = 0$ for all arcs that are not successors, we have $x^T Q x = \sum_{e \in A} \sum_{f \in \delta^+(e^-)} Q_{ef} x_e x_f$. Now,

$$\sum_{e \in A} \sum_{f \in \delta^+(e^-)} Q_{ef} x_e x_f = \sum_{e \in A} \sum_{f \in \delta^+(e^-)} (b_e + c_f) x_e x_f = \sum_{e \in A} b_e x_e \sum_{f \in \delta^+(e^-)} x_f + \sum_{f \in A} c_f x_f \sum_{e \in A} x_e = \sum_{e \in A} b_e x_e + \sum_{f \in A} c_f x_f = \sum_{e \in A} p_e x_e.$$ 

Here we use the fact that $\sum_{f \in \delta^+(e^-)} x_f = \sum_{e \in \delta^-(f^+)} x_e = 1$, since $x$ is a cycle cover. 

From Proposition 3 it follows that the incident weak sum property is a sufficient condition for $Q$ to be linearizable. By including linear arc costs, this result remains valid, since we only increase the entries on the diagonal of $Q$.

Moreover, note that when $Q$ satisfies the row or column CVP, then $Q$ is an incident weak sum matrix. Next, we provide a special type of instance for which the cost matrix is not by definition linearizable, but for which we can still obtain its optimal value by solving a linear cycle cover problem.

Definition 5. A matrix $Q \in \mathbb{R}^{m \times m}$ is called a symmetric product matrix if $Q = aa^T$ for some vector $a \in \mathbb{R}^m$.

Equivalently, we can say that $Q$ is a symmetric product matrix if it is a symmetric positive semidefinite matrix of rank one. Instances with such a quadratic cost matrix can be solved in polynomial time, as stated in the following proposition.
Proposition 4. If the quadratic cost matrix $Q$ is a symmetric product matrix, i.e. $Q = aa^T$ for some $a \in \mathbb{R}^m$, then the optimal cycle cover can be computed in polynomial time.

Proof. Let $Q$ be such that $Q = aa^T$ for some $a \in \mathbb{R}^m$. For all feasible cycle covers $x \in X$, we have:

$$x^T Q x = x^T a a^T x = (a^T x)^2 = (a^T x)^2.$$

Minimizing $x^T Q x$ over all $x \in X$ is then equivalent to minimizing $a^T x$ over all cycle covers $x \in X$. \hfill \Box

Until now we considered instances for which $Q$ is of the desired type, i.e., $Q_{ef} = 0$ when $f$ is not a successor of $e$. Below we derive two sufficient conditions for linearizability of a matrix $Q$ which can have nonzero interaction cost between non-incident arcs. Although these cost matrices do not meet the assumptions of the QCCP, we can still use them to derive strong bounds for the objective value of the original problem. This is addressed in Section 4.

Punnen et al. [30] introduce a generalized version of the weak sum property for the QTSP. Their approach can be applied to the QCCP. However, since the authors of [30] prove the condition to hold for complete graphs, we provide a proof for general digraphs.

First, we define some new terminology. Instead of writing $Q_{ij}$ for $i, j \in A$, we write $Q_{ij,kl}$ with $(i, j), (k, l) \in A$. Let $N_i^+$ denote the set of nodes $j$ for which there exists an arc $(i, j) \in A$, i.e. $N_i^+ := \{ j \in N | (i, j) \in A \}$. Similarly, let $N_i^-$ be the set of nodes $j$ for which an arc $(j, i) \in A$ exists, i.e. $N_i^- := \{ j \in N | (j, i) \in A \}$. Now we can introduce the notion of a generalized weak sum matrix and prove that it is linearizable.

Definition 6. $Q$ is called a generalized weak sum matrix if there exist $B, C \in \mathbb{R}^{m \times n}$ and $D, T \in \mathbb{R}^{n \times m}$ such that $Q_{ij,kl} = b_{ij,k} + c_{ij,t} + d_{ij,kl} + t_{ij,kl}$ for all $(i, j, k, l) \in A$. If such $B, C, D$ and $T$ exist, these matrices are called supporting matrices of $Q$.

Now we can prove the following proposition.

Proposition 5. If $Q$ is a generalized weak sum matrix with supporting matrices $B, C \in \mathbb{R}^{m \times n}$ and $D, T \in \mathbb{R}^{n \times m}$, then $Q$ is linearizable with cost vector $p$ given by $p_{ij} = \sum_{k=1}^n b_{ij,k} + \sum_{l=1}^n d_{ij,l} + \sum_{k=1}^n t_{ij,kl}$.

Proof. Let $\tilde{b}_{ij} := b_{ij,k}$, $\tilde{c}_{ij} := \sum_{k=1}^n c_{ij,k}$, $\tilde{d}_{ij} := \sum_{k=1}^n d_{ij,k}$, $\tilde{t}_{ij} := \sum_{k=1}^n t_{ij,kl}$ and $p'_{ij} = b_{ij} + c_{ij} + d_{ij} + t_{ij}$ for all $(i, j) \in A$. Then, for $x \in X$, we have:

$$x^T Q x = \sum_{i \in N} \sum_{j \in N_i^+} \sum_{k \in N_j^+} \sum_{l \in N_l^+} Q_{ij,kl} x_{ij} x_{kl},$$

where we use the fact that $\sum_{l \in N_j^+} x_{kl} = \sum_{k \in N_i^-} x_{kl} = \sum_{j \in N_i^+} x_{ij} = \sum_{i \in N_j^-} x_{ij} = 1$ since $x$ is a cycle cover. \hfill \Box
Note that an incident weak sum matrix can be seen as a special case of a generalized weak sum matrix. That is, for all \((i,j)\in A\) we set \(b_{ij,j} = b_{ij}\) and \(b_{ij,k} = 0\) for all \(k \neq j\) and for all \((k,l) \in A\) we set \(t_{k,kl} = t_{kl}\) and \(t_{j,kl} = 0\) for all \(j \neq k\). Moreover, let \(C\) and \(D\) be the zero matrix. Then we have \(Q_{ij,kl} = b_{ij,j} + c_{ij,k} + d_{ij,l} + t_{j,kl} = b_{ij} + t_{j}t_{kl}\) for all \((i,j),(j,l) \in A\) and \(Q_{ij,kl} = 0\) otherwise.

When \(Q\) is a generalized weak sum matrix, we need \(4mn\) parameters to describe \(Q\). This number can be reduced by considering a more restricted version of a generalized weak sum matrix.

**Definition 7.** \(Q\) is called a restricted generalized weak sum matrix if there exist \(C \in \mathbb{R}^{m \times m}\), \(D \in \mathbb{R}^{n \times m}\) and \(b, t \in \mathbb{R}^m\) such that \(Q_{ij,kl} = b_{ij,j} + c_{ij,k} + d_{ij,l} + t_{j,kl}\) for all \(i,j, l \in A\) with \((i,j),(j,l) \in A\) and \(Q_{ij,kl} = c_{ij,k} + d_{ij,l}\) otherwise. If such \(C, D, b, t\) exist, these are called supporting matrices and vectors, respectively.

We can show that restricted generalized weak sum matrices are linearizable.

**Proposition 6.** If \(Q\) is a restricted generalized weak sum matrix with supporting matrices \(C, D\) and supporting vectors \(b, t\), then \(Q\) is linearizable with vector \(p\) given by \(p_{ij} = b_{ij} + \sum_{k=1}^n c_{ij,k} + \sum_{k=1}^n d_{ij,k} + t_{ij}\).

**Proof.** Define \(B \in \mathbb{R}^{m \times n}\) and \(T \in \mathbb{R}^{n \times m}\) as follows:

\[
B_{ij,k} = \begin{cases} b_{ij} & \text{if } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall (i,j) \in A, \forall k \in N
\]

\[
T_{k,ij} = \begin{cases} t_{ij} & \text{if } k = i \\ 0 & \text{otherwise} \end{cases} \quad \forall (i,j) \in A, \forall k \in N
\]

Now the matrices \(B, C, D\) and \(T\) are such that they satisfy the conditions of Proposition 5. This implies that \(Q\) is linearizable with vector \(p'\) given by \(p'_{ij} = \sum_{k=1}^n b_{ij,k} + \sum_{k=1}^n c_{ij,k} + \sum_{k=1}^n d_{ij,k} + \sum_{k=1}^n t_{k,ij}\). Since \(\sum_{k=1}^n b_{ij,k} = b_{ij}\) and \(\sum_{k=1}^n t_{k,ij} = t_{ij}\), it follows that \(Q\) is linearizable with vector \(p := b_{ij} + \sum_{k=1}^n b_{ij,k} + \sum_{k=1}^n d_{ij,k} + t_{ij}\).

\[\square\]

### 4 Linearization Based Bounds for QCCP

In this section we show how the sufficient conditions for linearizability can be used to derive bounds for the optimal value of the QCCP. The construction of these bounds is provided in Section 4.1. Section 4.2 shows some preliminary numerical results of these bounding procedures.

#### 4.1 Construction of Linearization Based Bounds

When an instance of the QCCP is linearizable, we can solve the problem in polynomial time by solving the corresponding linear cycle cover problem. When a quadratic cost matrix \(Q\) is not linearizable, we can still use the sufficient conditions for linearizability to find lower bounds for the optimal value of the problem. This approach is introduced by Hu and Sotirov [21] for general binary quadratic problems. We here use tailor made sufficient conditions for the QCCP, which leads to efficient lower bounds as we show later in the numerical results.

Before we proceed, let us recall the linear cycle cover problem. We introduce the matrix \(U \in \mathbb{R}^{n \times m}\) with \(U_{ie} = 1\) if node \(i\) is the starting node of arc \(e\) and \(0\) otherwise. Similarly, we define \(V \in \mathbb{R}^{n \times m}\) with \(V_{ie} = 1\) if node \(i\) is the ending node of arc \(e\) and \(0\) otherwise. Since the matrix \([U^T, V^T]^T\) is totally unimodular, the optimal value of the CCP using cost vector \(p\) equals

\[
OPT(p) := \min_{x \in \mathbb{R}^m} \{p^T x | \begin{bmatrix} U \\ V \end{bmatrix} x = \mathbb{1}_{2n}, x \geq 0\}
\]
where $\mathbb{1}_{2n} \in \mathbb{R}^{2n}$ equals the vector of ones. Note that (3) and (4) are equivalent optimization problems. The corresponding dual problem is given in (5).

When $Q$ is linearizable with linearization vector $p$, we can find the optimal value for the QCCP by computing $OPT(p)$ using (4) or (5). If $Q$ is not linearizable, we can search for a linearizable matrix $\hat{Q}$ that is as close as possible to $Q$. To guarantee that $\hat{Q}$ is indeed linearizable, it should satisfy one of the sufficient conditions for linearizability derived in Section 3. We define the sets $S_i(Q)$, for $i = 1, 2, 3$, consisting of cost matrices $\hat{Q}$ such that $\hat{Q}$ is linearizable w.r.t. a sufficient condition for linearizability and $Q - \hat{Q}$ is elementwise nonnegative. We have

$$S_1(Q) := \{ \hat{Q} \in \mathbb{R}^{m \times m} \mid \hat{Q} \text{ is incident weak sum and } Q - \hat{Q} \geq 0 \},$$

$$S_2(Q) := \{ \hat{Q} \in \mathbb{R}^{m \times m} \mid \hat{Q} \text{ is restricted generalized weak sum and } Q - \hat{Q} \geq 0 \},$$

$$S_3(Q) := \{ \hat{Q} \in \mathbb{R}^{m \times m} \mid \hat{Q} \text{ is generalized weak sum and } Q - \hat{Q} \geq 0 \}.$$

**Remark 1.** We do not consider the sets of cost matrices $Q$ satisfying the row or column CVP, since these are special types of incident weak sum matrices. These type of matrices are contained in $S_1$.

$S_i(Q)$ can be seen as the set of all the linearizable cost matrices of a specific type that are suitable for obtaining lower bounds for the optimal value of the problem. For this purpose, we define for $i = 1, 2, 3$ the set $\tau_i(Q)$ of cost vectors $\hat{p} \in \mathbb{R}^m$ as

$$\tau_i(Q) := \{ \hat{p} \in \mathbb{R}^m \mid x^T \hat{p} = x^T \hat{Q} x \text{ for all } x \in X, \hat{Q} \in S_i(Q) \}.$$  

It is clear that for all $i$ and all $\hat{p} \in \tau_i(Q)$ we have

$$OPT(Q) = \min_{x \in X} \{ x^T Q x \} \geq \min_{x \in X} \{ x^T \hat{Q} x \} = \min_{x \in X} \{ x^T \hat{p} \} = OPT(\hat{p}).$$

So, indeed, $OPT(\hat{p})$ is a lower bound for the optimal objective value of the QCCP for all $i$ and $\hat{p} \in \tau_i(Q)$. By maximizing over all cost vectors in $\tau_i(Q)$, we obtain the strongest linearization based bound with respect to the set $S_i(Q)$, which we denote by $v^i_{LBB}$, see also [21]:

$$v^i_{LBB} := \max_{\hat{p} \in \tau_i(Q)} \{ OPT(\hat{p}) \} = \max_{\hat{p} \in \mathbb{R}^m} \{ \mathbb{1}_{2n}^T y \mid [U^T, V^T] y \leq \hat{p}, \hat{p} \in \tau_i(Q) \}.$$  

The corresponding bounding approaches are denoted by $LBB1$, $LBB2$ and $LBB3$, respectively.

**Remark 2.** Recall that the matrices in $S_2(Q)$ and $S_3(Q)$ can have nonzero interaction cost for non-incident arcs, so they do not satisfy the assumptions on the quadratic cost matrix of the QCCP. Nevertheless, they can still be used to derive lower bounds for the original problem. In other words, we search for the general quadratic cost matrix that is linearizable and as close as possible to $Q$ that gives us the best lower bound.

As long as the set $\tau_i(Q)$ is a polyhedron, the corresponding bound $v^i_{LBB}$ can be calculated by solving the linear programming problem [9]. The sets $\tau_i(Q)$ for $i = 1, 2, 3$ are indeed nonempty polyhedra, since they can be described by a finite number of linear equalities and inequalities. These polyhedral descriptions are provided in Table 1.
matrix and the generalized weak sum matrix.

Proof.

Theorem 2. Consequently, we can establish the following result about the quality of the corresponding linearization based bounds.

Hu and Sotirov [21] argue that the linearization based bounds can be improved by extending the set $\tau(Q)$ using a skew symmetric matrix $M$. That is, since each skew symmetric matrix is linearizable, a matrix $\hat{Q}$ is linearizable if and only if $\hat{Q} + M$ is linearizable for all $M$ with $M + M^T = 0$. Using this, the set $\tau(Q)$ can be extended to:

$$\tau_1^{skew}(Q) := \{ \hat{p} \in \mathbb{R}^m | b_e + c_e + M_e \leq Q_{ef}, \forall e \in A, f \in \delta^+(e^-), \hat{p}_e = b_e + c_e, \forall e \in A, M_e + M_f = 0 , \forall e \in A, f \in \delta^+(e^-), b, c \in \mathbb{R}^m, M \in \mathbb{R}^{m \times m} \} \quad (7)$$

Note that in $\tau_1^{skew}(Q)$ we include only incident skew symmetric matrices, since adding dense skew symmetric matrices would increase computational complexity. Since $\tau_1(Q) \subseteq \tau_1^{skew}(Q)$, it follows that we can obtain a stronger bound by maximizing over $\tau_1^{skew}(Q)$, see Section 8. The same extension can be applied to any set $\tau_i(Q)$.

### 4.2 Preliminary Results

In order to check the quality of the bounds derived above, we perform a preliminary numerical study. We create instances according to the $G(n, p)$ Erdős-Rényi model, see [11]. Here $n$ equals the number of nodes and $p$ equals the probability that an arc is included. We create instances for various values of $n$ and $p$. The interaction cost between any two incident arcs is drawn uniformly at random as an integer from $\{1,...,100\}$. For a more detailed explanation of the instances, we refer to Section 8. In Table 2 we present the bounds $v_{LBB}^1$, $v_{LBB}^2$, and the computational times required for computing them. Experiments are performed using a pc with an Intel(R) Core(TM) i7-6500 CPU, 3.20 GHz and 8 GB memory using CPLEX 12.6 as solver. The maximum computation time is set to 3600 seconds and we put ‘n.a.’ in the table when this maximum is reached before a solution is obtained.

| Set         | Type of cost matrix | (In)equalities which describe the set                                                                 |
|-------------|---------------------|--------------------------------------------------------------------------------------------------------|
| $\tau_1(Q)$ | Incident weak sum matrix | $b_e + c_e \leq Q_{ef}$, $\forall e \in A, f \in \delta^+(e^-)$                                       |
|             | restricted generalized weak sum matrix | $b_e = b_e + c_e$, $b, c \in \mathbb{R}^m$                                                            |
|             | Generalized weak sum matrix | $\hat{p}_e = b_e + c_e$, $\forall e \in A, f \in \delta^+(e^-)$                                       |

Table 1: Polyhedral descriptions of the sets $\tau_1(Q), \tau_2(Q)$ and $\tau_3(Q)$.

By construction, we have $\tau_1(Q) \subseteq \tau_2(Q) \subseteq \tau_3(Q)$ for all quadratic cost matrices $Q$. Consequently, we can establish the following result about the quality of the corresponding linearization based bounds.

**Theorem 2.** For all instances of the QCCP, we have $v_{LBB}^1 \leq v_{LBB}^2 \leq v_{LBB}^3$.

**Proof.** The proof follows immediately from the construction of the linearization based bounds and the definitions of the incident weak sum matrix, the restricted generalized weak sum matrix and the generalized weak sum matrix.

---

12
obtain a sequence of bounds. However, it is in general not possible to find a vector
which this bound has strictly improved. That is, since $p_0 \leq p_1$ for all cycle covers
of this new objective function into account. By construction we have
reformulated as $x^T Q x$ be seen as the largest linearization of the matrix
of this type with $\hat{Q}_0$. In this paper we introduce a new approach that is different in two ways. First,
resulting sequence of bounds remains constant after the first iteration. To overcome this
bound. It is important to note that one can construct such a bounding procedure using any
In literature, various iterative bounding procedures have been proposed, see e.g. \cite{5, 29, 31, 32}. In this paper we introduce a new approach that is different in two ways. First,

\begin{table}[h]
\centering
\begin{tabular}{c|ccc|ccc|ccc}
$p$ & $n$ & $m$ & $OPT$ & $LBB1$ & $LBB2$ & $LBB3$
\hline
0.3 & 15 & 61 & 485 & 478 & 0.010 & 478 & 0.635 & 485 & 0.714 \\
20 & 118 & 438 & 377 & 0.031 & 384 & 1.265 & 390 & 77.21 \\
25 & 172 & 382 & 291 & 0.050 & 295 & 2.531 & 295 & 869.0 \\
0.5 & 15 & 116 & 226 & 215 & 0.034 & 215 & 1.407 & 215 & 90.39 \\
20 & 177 & 255 & 189 & 0.059 & 190 & 12.05 & 190 & 2105 \\
25 & 306 & n.a. & 172 & 0.516 & 173 & 353.6 & n.a. & 3600 \\
\end{tabular}
\caption{Bounds and computation times of linearization based bounds on Erdős-Rényi instances.}
\end{table}

By construction, the optimal solution has always an integer objective value. Therefore, we
round up all bounds to the nearest integer. The results of Table 2 show that the linearization
based bounds $LBB1$, $LBB2$ and $LBB3$ do not differ significantly, especially for the larger
instances. At the same time, the computation times differ significantly. It turns out that
$LBB1$ is most efficient. Therefore, this bound can be preferred when taking both quality
and efficiency into account.

\section{Reformulated LBB Approach}

In this section we discuss how a reformulation of the quadratic cost matrix can be used to
obtain a non-decreasing sequence of lower bounds that are based on the linearization based
bound. In other words, we search for a linearizable matrix in $S(Q_0)$. In other words, we search for a linearizable matrix in $S(Q_0)$, we know there exists some
$p_1 \in \tau(Q_0)$ such that $x^T Q_0 x = x^T p_1$ for all $x \in X$. This leads to the following reformulation
of the objective function
\begin{equation}
x^T Q_0 x = x^T Q_0 x + x^T (Q_0 - \hat{Q}_0) x = x^T p_1 + x^T (Q_0 - \hat{Q}_0) x
\end{equation}
for all cycle covers $x \in X$. By letting $p_0$ be the $m \times 1$ zero vector and $Q_1 := Q_0 - \hat{Q}_0$, this
relation can be written as $x^T p_0 + x^T Q_0 x = x^T p_1 + x^T Q_1 x$ for all $x \in X$. The vector $p_1$ can
be seen as the largest linearization of the matrix $Q_0$ (see \cite{4}) with $Q_1$ being the residual
quadratic part. Recall that the bound $u_{\text{LBB}}$ is calculated by only taking the linear part
of this new objective function into account. By construction we have $x^T p_0 \leq x^T p_1$ for all
$x \in X$ and $0 \leq Q_1 \leq Q_0$.

Now we can proceed in a similar way by considering the linearization problem of the
residual cost matrix $Q_1$. In other words, we search for a linearizable matrix in $S(Q_1)$ and
its corresponding linearization vector $p_2$. If we let $Q_2$ be the new residual matrix, then the
objective function can be reformulated as $x^T p_0 + x^T Q_0 x = x^T (p_1 + p_2) + x^T Q_2 x$. Since
$x^T p_1 \leq x^T (p_1 + p_2)$, a stronger bound can be obtained. This procedure can be repeated to
obtain a sequence of bounds. However, it is in general not possible to find a vector $p_2$ for
which this bound has strictly improved. That is, since $p_1 + p_2 \in \tau(Q_0)$ this would imply
that $p_1$ is not the optimal solution to \cite{4}. Thus, applying this procedure iteratively, the
resulting sequence of bounds remains constant after the first iteration. To overcome this
issue, we need to reformulate the residual cost matrix in each step.

In literature, various iterative bounding procedures have been proposed, see e.g. \cite{5, 29, 31, 32}. In this paper we introduce a new approach that is different in two ways. First,
the existing bounding procedures are mainly based on the classical Gilmore-Lawler type bound. Our approach is based on general sufficient conditions for linearizability and we can show that the Gilmore-Lawler type bounding procedure is a special case of this approach, see Section 6. Second, the existing bounding procedures mostly use a fixed reformulation of the cost matrix in each iteration. However, using a fixed reformulation is in general not the best one can do. Here, we search for the reformulation of the cost matrix that results in the strongest bound in the next iteration. For this purpose, we define the notion of an equivalent representation of a matrix, see e.g. [29].

**Definition 8.** Let \((G, Q)\) be an instance of the QCCP. Then \((G, R)\) is an equivalent representation of \((G, Q)\) if \(x^T Q x = x^T R x\) for all \(x \in X\).

If there is no confusion about the graph \(G\) under consideration, we say that \(R\) is an equivalent representation of \(Q\). It is easy to verify that a matrix \(R\) is an equivalent representation of \(Q\) if for all \(e, f \in A\) we have \(R_{ef} + R_{fe} = Q_{ef} + Q_{fe}\). Here, we focus on a specific type of equivalent representation, which we call an \(\eta\)-equivalent representation of \(Q\).

**Definition 9.** Given \(\eta \in [0, 1]\), an equivalent representation \(Q^\eta := \eta Q + (1 - \eta)Q^T\) of \(Q\) is called an \(\eta\)-equivalent representation.

In other words, an \(\eta\)-equivalent representation is obtained by taking a convex combination of \(Q\) and \(Q^T\). Moreover, it follows that if \(R\) and \(Q\) are equivalent representations, then a linearization of \(R\) is also a linearization of \(Q\) and vice versa.

Instead of considering the linearization problem of the residual matrix \(Q_1\), we can consider the linearization problem of \(Q^n_1\) for some \(\eta \in [0, 1]\). Since \(Q^n_1\) has a different structure than \(Q_1\), it is in general possible to find a linearizable matrix \(Q_1 \in S(Q^n_1)\) and a corresponding linearization vector that result in a strictly stronger bound.

As already mentioned above, many approaches in literature are based on taking a fixed value for \(\eta\), e.g. \(\eta = \frac{1}{2}\) which corresponds to the case of symmetrizing. This does not give the best bound in general. Instead, we search for \(\eta \in [0, 1]\) that results in the strongest bound in each iteration. Suppose we are in step \(k\) of the algorithm in which we consider the linearization problem of the residual matrix \(Q_{k-1}\). Then the optimal equivalent representation of \(Q_{k-1}\) and its corresponding vector \(p_k \in \tau(Q^n_{k-1})\) can be computed simultaneously by solving the following problem:

\[
r_k := \max \{ \frac{1}{2} y^T y \mid [U^T, V^T] y \leq p_k, p_k \in \tau(Q^n_{k-1})\},
\]  

(9)

which equals the additional amount of quadratic cost that can be linearized in iteration \(k\). Note that if the set \(\tau(Q_{k-1})\) is a polyhedron, then \(\tau(Q^n_{k-1})\) is also a polyhedron and the corresponding problem (9) can be solved in polynomial time. For the sufficient conditions mentioned in Section 6 this is indeed the case.

Finally, we provide a new bounding procedure that is based on iteratively finding the best equivalent representation of the residual cost matrix and its optimal linearizable matrix. Starting with \(Q_0 = Q\), the goal is to find the best linearizable matrix \(Q_k\) of an equivalent representation of the residual matrix \(Q_k\) and its corresponding linearization vector \(p_k\). In each iteration we compute \(r_k\) by (9) and let \(d_k = d_{k-1} + p_k\) which equals the total linearization vector of \(Q\). The final bound is given by the sum of all \(r_k\)'s, which we call the reformulation based bound. The procedure is given in Algorithm 1.
Algorithm 1 LBB Reformulation Algorithm

1: Initialize: \( Q_0 = Q, d_0 = 0, k = 1, r_0 = \infty \)
2: while \( r_{k-1} > 0 \) do
3: Compute \( r_k, p_k \) and \( \eta \) using (9).
4: Obtain \( ˆQ_{k-1} \) from the structure of \( \tau(Q_{k-1}) \).
5: Update \( Q_k \) as \( \eta Q_{k-1} + (1 - \eta) ˆQ_{k-1} - ˆQ_{k-1} \).
6: Update \( d_k \) as \( d_{k-1} + p_k \).
7: Update \( k \) as \( k + 1 \).
8: end while
9: Compute \( v_{RBB} = \sum_{i=1}^{k-1} r_i \).
10: return \( d_k, v_{RBB} \).

Remark 3. Note that step 4 of Algorithm 1 depends on the specific sufficient condition for linearizability.

Hu and Sotirov [21] show that in the case that the linearizable matrix \( ˆQ \) is in the form
\( \hat{Q} = [U^T, V^T]Y + \text{Diag}(z) \) for some \( Y \in \mathbb{R}^{2m \times m} \) and \( z \in \mathbb{R}^m \), the bound \( v_{RBB} \) is dominated by the solution of the first level RLT relaxation introduced by Adams and Sherali [1, 2]. Here RLT stands for reformulation linearization technique. In [21] it is moreover shown that the first level RLT bound, denoted by \( v_{RLT1} \), can be obtained by searching for the optimal linearizable matrix \( \hat{Q} \) of the form \( \hat{Q} = [U^T, V^T]Y + M + \text{Diag}(z) \) where \( M \) is a skew symmetric matrix. Our preliminary numerical results show that the above algorithm does not improve significantly the LBB1 bound. However, in the next section we show that our approach outperforms known iterative approaches related to the Gilmore-Lawler bounds.

6 The Gilmore-Lawler Type Bound

In this section we consider the classical Gilmore-Lawler type bound. The GL procedure is a well-known approach to construct lower bounds for quadratic 0-1 optimization problems, see e.g. [15, 26, 31, 32]. We provide a compact formulation of the GL type bound that can be used to compute the bound by a single LP-problem, instead of solving \( m \) subproblems. Moreover, we show that this bound in fact belongs to the family of linearization based bounds. Therefore, based on the results of Section 5, we provide a bounding procedure that computes the best GL type bound in each step of the algorithm. We conclude this section by testing this new bounding procedure on some preliminary test instances.

6.1 The classical GL type bound

In the objective function of the QCCP, see (1), we have the quadratic term \( x_e x_f \) for each two arcs \( e, f \in A \) placed in succession on a cycle. To get rid of this quadratic term, for each given arc \( e \in A \), potentially in the solution, we consider the cycle cover containing \( e \) with minimum interaction cost with \( e \). We denote this minimum contribution of arc \( e \) to a solution by \( z_e \). In particular, for all \( e \in A \) we have

\[
(P_e) \quad z_e := \min \{ Q_e, x \mid x \in X, x_e = 1 \},
\]

where \( Q_e \) denotes the \( e \)th row of the cost matrix \( Q \). The feasible set of \((P_e)\) equals the set of all node-disjoint cycle covers containing arc \( e \). If this set is empty, then we set \( z_e \) equal to 0 since arc \( e \) can not contribute to a cycle cover. Let \( z \in \mathbb{R}^m \) be the vector consisting of the elements \( z_e \) for all \( e \in A \). Then the classical GL type bound is obtained by solving the following CCP:

\[
(GL) \quad v_{GL} := \min \{ z^T x \mid x \in X \}.
\]
Note that the constraint matrices of \((P_e)\) and \((GL)\) are totally unimodular. For this reason, we can drop the integrality constraints and solve \((GL)\) and \((P_e)\) for all \(e \in A\) as linear programming problems.

Besides computing the GL type bound by solving \((GL)\) and \((P_e)\) for all \(e \in A\), we can also obtain its value by solving an integer linear programming (ILP) problem. The problem \((GL_{ILP})\) is defined as follows:

\[
(GL_{ILP}) : \min_{e \in A} \sum_{f \in A} Q_{ef} y_{ef} \\
\text{s.t.} \sum_{f \in \delta^{+}(i)} y_{ef} = \sum_{f \in \delta^{-}(i)} y_{ef} = x_e \quad \forall i \in N, \forall e \in A \\
y_{ee} = x_e \quad \forall e \in A \\
y_{ef} \in \{0, 1\}, \quad x \in X \quad \forall e, f \in A.
\]

It follows from the constraints that if \(x_e = 1\), then \(y_{e,:} := \begin{bmatrix} y_{e1}, \ldots, y_{em} \end{bmatrix}\) is the characteristic vector of the cheapest cycle cover containing arc \(e\) and if \(x_e = 0\), then \(y_{e,:}\) equals the zero vector.

Let \((CGL_{ILP})\) be the continuous relaxation of \((GL_{ILP})\). In this continuous relaxation we can omit the upper bounds on \(x_e\) and \(y_{ef}\) for all \(e, f \in A\), since it is never optimal to set the value of these variables larger than one. We can compute the GL type bound by solving \((CGL_{ILP})\) as stated by the following theorem. This theorem is based on a similar result for the QMST, see [31].

**Theorem 3.** The optimal value of \((CGL_{ILP})\) equals \(v_{GL}\).

**Proof.** Let \(\lambda_{e,i}\) and \(\alpha_{e,i}\) be the dual variables corresponding to constraints \([11]\), i.e. \(\lambda_{e,i}\) corresponds to \(\sum_{f \in \delta^{+}(i)} y_{ef} = x_e\) and \(\alpha_{e,i}\) corresponds to \(\sum_{f \in \delta^{-}(i)} y_{ef} = x_e\). Similarly, let \(\mu_i\) and \(\gamma_i\) be the dual variables corresponding to the first and second equalities of constraints \([2]\), and \(\theta_e\) the dual variable corresponding to constraints \([12]\). The dual problem of \((CGL_{ILP})\) is as follows:

\[
(DCGGL_{ILP}) : \max \sum_{i \in N} \mu_i + \sum_{i \in N} \gamma_i \\
\text{s.t.} \lambda_{e,f^+} + \alpha_{e,f^-} \leq Q_{ef} \quad \forall e, f \in A, \ f \neq e \\
\lambda_{e,e^+} + \alpha_{e,e^-} + \theta_e \leq Q_{ee} \quad \forall e \in A \\
- \sum_{i \in N} \lambda_{e,i} - \sum_{i \in N} \alpha_{e,i} + \gamma_{e^-} + \mu_{e^+} - \theta_e \leq 0 \quad \forall e \in A.
\]

Constraint \([17]\) can be rewritten as \(\gamma_{e^-} + \mu_{e^+} \leq \sum_{i \in N} \lambda_{e,i} + \sum_{i \in N} \alpha_{e,i} + \theta_e\) for all \(e \in A\). In order to maximize the objective function of \((DCGGL_{ILP})\), we maximize the right hand side of this inequality subject to constraints \([15]-[16]\). This gives for each \(e \in A\) the following subproblem:

\[
(DCP'_e) : z'_e := \max \left\{ \sum_{i \in N} \lambda_{e,i} + \sum_{i \in N} \alpha_{e,i} + \theta_e \mid ([15],[16]) \right\}.
\]

For each fixed \(e \in A\) the subproblem given above equals the dual of the continuous relaxation of \((P_e)\). By strong duality we know \(z'_e = z_e\) for all \(e \in A\). Substitution of this term into constraint \([17]\) gives a rewritten formulation for \((DCGGL_{ILP})\):

\[
\max \left\{ \sum_{i \in N} \mu_i + \sum_{i \in N} \gamma_i \mid \mu_{e^+} + \gamma_{e^-} \leq z_e \quad \forall e \in A \right\}.
\]

This problem exactly equals the dual of the continuous relaxation of \((GL)\). Because of strong duality, it follows that the optimal objective value of \((CGL_{ILP})\) equals \(v_{GL}\). \(\square\)
We can show that the Gilmore-Lawler type bound for the QCCP in fact belongs to the family of linearization based bounds introduced in Section [4]. That is, we can obtain \( v_{GL} \) by searching for a linearizable quadratic cost matrix \( Q \) of a specific type that is as close as possible to \( Q \). The required linearizability condition on \( Q \) is given below, and it differs from the sufficient conditions presented in Section [3].

**Proposition 7.** If there exists \( B, C \in \mathbb{R}^{m \times n} \) and \( t \in \mathbb{R}^m \) such that \( Q_{ef} = B_{e,f}^+ + C_{e,f}^- \) for \( e \neq f \) and \( Q_{ee} = B_{e,e}^+ + C_{e,e}^- + t_e \) for all \( e \in A \), then \( Q \) is linearizable with vector \( p \) given by \( p_e = t_e + \sum_{i=1}^n B_{e,i} + \sum_{i=1}^n C_{e,i} \).

**Proof.** Let \( \hat{Q} \) be defined as \( \hat{Q}_{ef} = B_{e,f}^+ + C_{e,f}^- \) for all \( e, f \in A \). Then \( \hat{Q} \) can be seen as a generalized weak sum matrix where \( D \) and \( T \) are equal to the zero matrix, see Definition [6]. According to Proposition [3], \( Q \) is linearizable with vector \( \hat{p} = \sum_{i=1}^n B_{e,i} + \sum_{i=1}^n C_{e,i} \). Since \( Q = \hat{Q} + \text{Diag}(t) \), it follows that \( Q \) is linearizable with vector \( p \) given by \( p_e = t_e + \sum_{i=1}^n B_{e,i} + \sum_{i=1}^n C_{e,i} \). \( \square \)

Similar to the notation used in Section [3] let \( S_{GL}(Q) \) denote the set of all linearizable cost matrices \( \hat{Q} \in \mathbb{R}^{m \times m} \) that satisfy the conditions of Proposition [7]. Moreover, let \( \tau_{GL}(Q) \) be the following polyhedron:

\[
\tau_{GL}(Q) := \{ \hat{p} \in \mathbb{R}^m \mid x^T \hat{p} = x^T \hat{Q}x \text{ for all } x \in X, \hat{Q} \in S_{GL}(Q) \}, \tag{18}
\]

and

\[
v_{LBB}^{GL} := \max_{\hat{p} \in \mathbb{R}^m} \{ \frac{1}{2} y^T \hat{p} \mid [U^T, V^T]y \leq \hat{p} \} \tag{19}
\]

Now we prove the main result of this section which states that the classical Gilmore-Lawler type bound can be seen as a special case of linearization based bound.

**Theorem 4.** We have \( v_{LBB}^{GL} = v_{GL} \).

**Proof.** By using the polyhedral description of \( S_{GL}(Q) \) following from Proposition [7], the optimization problem in (19) can be written as follows:

\[
v_{LBB}^{GL} = \max_{\hat{y} \in \mathbb{R}^{2n}} \sum_{i=1}^{2n} y_i \tag{20}
\]

s.t. \([U^T, V^T]y \leq \hat{p} \)

\[
B_{e,f}^+ + C_{e,f}^- \leq Q_{ef} \quad \forall e, f \in A, f \neq e \tag{21}
\]

\[
B_{e,e}^+ + C_{e,e}^- + t_e \leq Q_{ee} \quad \forall e \in A \tag{22}
\]

\[
\hat{p}_e = t_e + \sum_{i=1}^n B_{e,i} + \sum_{i=1}^n C_{e,i} \quad \forall e \in A \tag{23}
\]

\[
y \in \mathbb{R}^{2n}, \hat{p}, t \in \mathbb{R}^m, B, C \in \mathbb{R}^{m \times n}. \tag{24}
\]

We show that this optimization problem is equivalent to \((DCGL_{ILP})\), the dual problem of the continuous relaxation of \((GL_{ILP})\). Take \( B_{e,i} = \lambda_{e,i} \) and \( C_{e,i} = \alpha_{e,i} \) for all \( e \in A \) and \( i \in N \), where \( \lambda \) and \( \alpha \) denote the dual vectors belonging to constraints [11]. Similarly, let \( t = \theta \) where \( \theta \) equals the dual vector to constraints [12]. Finally, let \( y = [\mu^T, \gamma^T]^T \), where \( \mu \) and \( \gamma \) are the dual variables belonging to constraints [2]. By substitution of these variables and combining constraints [21] and [24], we obtain the problem \((DCGL_{ILP})\), i.e. the dual of \((GL_{ILP})\). Thus, we have \( v_{LBB}^{GL} = v_{GL} \). \( \square \)

Theorem 4 shows that the GL type bound belongs to the family of linearization based bounds. This is also shown in [21, 32], however our proof is very different as we exploit the fact that \( v_{GL} \) can be obtained by solving an LP problem i.e. \((GL_{ILP})\). Additionally, we show here that the computation of the GL type bound is equivalent to the search for the optimal linearizable cost matrix \( \hat{Q} \) satisfying the properties of Proposition [7].
6.2 The best Gilmore-Lawler type bound

Section 6.1 shows that the calculation of the classical GL type bound fits in the general framework discussed in Section 4. In this section we apply the reformulation procedure of Section 5 to the GL type bound. We also show that our approach outperforms several iterative approaches from literature.

In order to apply Algorithm 1 to the sufficient condition for linearizability of Proposition 7, we need to define how to calculate $r_k$ for each iteration $k$, see [9]. We rewrite the set \( \tau_{GL}(Q) \), see [18], as follows:

\[
\tau_{GL}(Q) = \{ \hat{p} \in \mathbb{R}^m | t \in \mathbb{R}^m, B, C \in \mathbb{R}^{m \times n}, [22, 23, 24] \},
\]

which is clearly a polyhedron. Then for all $k \geq 1$ we calculate the additional amount of quadratic cost that is linearized by solving:

\[
r_k := \max_{y \in \mathbb{R}^m, p_k \in \mathbb{R}^m} \{ \bar{T}^T y | [U^T, V^T]y = p_k, p_k \in \tau_{GL}(Q^k) \}.
\]

(26)

Note that as opposed to the constraints in (9), we have replaced the constraint \([U^T, V^T]y \leq p_k\) by an equality constraint. This does not change the value of $r_k$. To verify this, suppose we solve (26) using the inequality constraint \([U^T, V^T]y \leq p_k\) and let \(\hat{y}, \hat{p}_k\) and \(\hat{t}\) be the corresponding optimal solutions. Let $e \in A$ be such that the inequality constraint is satisfied with strict inequality. Then without changing \(\hat{y}\), we can reduce \(t_e\) (and thus \(\hat{p}_e\)) such that we get equality for $e \in A$. Although it changes the linearization vector \(\hat{p}\), the resulting bound remains equal. To verify this, notice that only the left hand side of constraint (23) is decreased, so the solution is still feasible and the optimal value $r_k$ remains unchanged. From this, it follows that one may replace \([U^T, V^T]y \leq p_k\) by an equality constraint and solve $r_k$ as in (26).

Algorithm 1 using (25) to compute $r_k$, $p_k$, and $\eta$ gives a new bounding procedure for the QCCP. We call the resulting bound the reformulated GL-type bound (RGL) and denote its value by $v_{RGL}$. By construction, it iteratively computes the best Gilmore-Lawler type bound among all equivalent representations of the quadratic cost matrix.

The algorithm proposed in this section satisfies another interesting property, namely the vectors $d_k$ satisfy the constant value property for all $k \geq 0$. This is an important property for linearizability because the set of linearizable cost matrices for combinatorial optimization problems with interaction costs can be characterized by the constant value property, under certain conditions, see [25].

**Theorem 5.** All $d_k$ where $k \geq 0$ computed during the RGL approach, satisfy the constant value property, i.e., we have $x^T d_k = \bar{x}^T d_k$ for all feasible cycle covers $x, \bar{x} \in X$.

**Proof.** We apply a proof by induction on $k$. Note that the vector $d_0$ equals the $m \times 1$ vector of zeros which trivially satisfies the constant value property.

Now assume that the induction hypothesis is true for iteration $k - 1$, i.e. $x^T d_{k-1} = \bar{x}^T d_{k-1}$ for all feasible cycle covers $x, \bar{x} \in X$. In iteration $k$ we solve (26). Let $\hat{y} \in \mathbb{R}^{2n}$ and $\hat{p} \in \mathbb{R}^m$ be the optimal variables for this problem and split $\hat{y} = [\mu^T, \lambda^T]^T$ with $\mu, \lambda \in \mathbb{R}^n$. It follows that $[U^T, V^T]\hat{y} = U^T \mu + V^T \lambda = \hat{p}$. Now let $x \in X$ be any feasible cycle cover in $G$. Then we can sum up the rows of this system of equalities for all arcs $e \in A$ in the cycle cover implied by $x$:

\[
\sum_{e \in A \mid x_e = 1} (\mu_e^+ + \lambda_e^-) = \sum_{e \in A \mid x_e = 1} \hat{p}_e \iff \sum_{i \in N} \mu_i + \sum_{i \in N} \lambda_i = x^T \hat{p}
\]

where we use the fact that each vertex is visited exactly once on a cycle cover. So the quantity $x^T \hat{p}$ is equal for all $x \in X$. As a result, $\hat{p}$ satisfies the constant value property. The vector $d_k$ is constructed as $d_{k-1} + p_k$ with $p_k = \hat{p}$. Since $d_{k-1}$ and $\hat{p}$ satisfy the constant value property, it follows that $d_k$ satisfies the constant value property. \(\square\)
Remark 4. Since the GL type bound can be computed both as a linearization based bound and by solving \((CGL_{ILP})\) (see Theorem 4), the iterative approach derived in this section can also be defined in terms of \((CGL_{ILP})\). In that case, we iteratively compute \(v_{GL}\) and reformulate the quadratic cost matrix using the dual variables of \((CGL_{ILP})\). The details of this equivalent approach can be found in [27].

Since the linearizable matrix \(\hat{Q}\) of Proposition 7 can be written in the form \(\hat{Q} = [U^T, V^T]Y + \text{Diag}(z)\) for some \(Y \in \mathbb{R}^{2n \times m}\) and \(z \in \mathbb{R}^m\), it follows from [21] that we have \(v_{RGL} \leq v_{RLT1}^*\).

6.3 Preliminary Results

For the instances considered in the preliminary results of Section 4, we now test our Gilmore-Lawler type bounds. First, we compute the classical GL type bound, after symmetrizing the quadratic cost matrix \(Q\). This bound is denoted by \(GL\). Moreover, we consider the iterative GL type bounding approach where we symmetrize the quadratic cost matrix in each iteration. That is, we apply Algorithm 1 using (26) where instead of optimizing over \(\eta\), we set \(\eta = \frac{1}{2}\). We denote this bound by \(RGL_{sym}\). Finally, we report the bound \(RGL\) which is introduced in Section 6.2. The maximum computation time is set at 1800 seconds. The results are given in Table 3.

| \(p\) | \(n\) | \(m\) | \(OPT\) | \(GL\) | \(RGL_{sym}\) | \(RGL\) |
|------|------|------|--------|-------|-------------|-------|
|      |      |      |        | bound | time | bound | time |
| 0.3  | 15   | 61   | 485    | 347   | 0.061 | 368   | 1.481 |
|      | 20   | 118  | 438    | 223   | 0.068 | 263   | 3.482 |
|      | 25   | 172  | 382    | 176   | 0.136 | 190   | 5.105 |
| 0.5  | 15   | 116  | 226    | 102   | 0.336 | 110   | 2.602 |
|      | 20   | 177  | 255    | 93    | 0.118 | 103   | 5.365 |
|      | 25   | 306  | n.a.   | 66    | 0.296 | 75    | 13.80 |
|      |      |      |        | 75    | 13.80 | 93    | 1800 |

Table 3: Bounds and computation times of GL type bounds on Erdős-Rényi instances.

From Table 3 it follows that the iterative approaches significantly improve the classical GL type bound. Among these iterative approaches, \(RGL\) provides much stronger bounds than \(RGL_{sym}\). We conclude that this new approach of calculating the best GL type bound in each step provides better bounds than when setting \(\eta = \frac{1}{2}\) in the reformulation. However, it turns out that this improvement in the quality comes at the cost of efficiency. Clearly, we can stop our algorithm after a pre-specified number of iterations and/or time.

7 Other bounds for the QCCP

In this section we present several known bounding approaches from literature that can be applied to the QCCP. In the next section, we compare those bounds with the bounds introduced in this paper. We consider a column generation approach and a mixed integer linear programming (MILP) based bound.

Galbiati et al. [13] construct a column generation approach for the MinRC3. This approach can be extended to the QCCP. To the best of our knowledge, this is the only implemented lower bounding approach for the MINRC3 in literature.

Let \(\mathcal{C}\) be the set of all directed cycles in \(G\). Moreover, let \(\mathcal{C} \subseteq \mathcal{C}\) be a subset of cycles such that it contains at least one cycle cover. Let \(w_c\) be the cost of a cycle \(c \in \mathcal{C}\). Then the restricted master problem \((RMP)\) is given by:

\[
\min \sum_{c \in \mathcal{C}} w_c y_c
\]
Let $\pi \in \mathbb{R}^n$ be the vector of dual variables corresponding to constraint (28). Then the subproblem $(SP)$ searches for the cycle in $\mathcal{C}$ with the smallest (negative) reduced costs with respect to $\pi$, i.e.

$$(SP) \quad \min_{x,z} \{ x^T Q x - z^T \pi \mid \sum_{e \in \delta^+(i)} x_e = \sum_{e \in \delta^-(i)} x_e = z_i \ \forall i \in N, \sum_{e \in A} x_e \geq 2, x \in \{0,1\}^m, z \in \{0,1\}^n \},$$

where $z_i = 1$ if vertex $i$ is on the cycle and 0 otherwise. As stated in [13], the problem $(SP)$ is $\mathcal{NP}$-hard itself. The quadratic objective function can be linearized by standard linearization techniques. A lower bound on the optimal value of the QCCP can be obtained by iteratively solving the master problem and its corresponding subproblem. If a cycle with negative reduced cost is found, we add it to the set $\mathcal{C}$. This procedure is repeated until no more cycle with negative reduced cost is found or after some predefined stopping criteria. The obtained bound is denoted by $v_{CG}$.

Based on a procedure introduced by Glover [16], we can present the QCCP as an MILP problem. Let $z_e$ be computed as in $(P_e)$ for all $e \in A$, see Section 6.1. Moreover, define for all $e \in A$ the value $q_{e}^{\text{max}} := \max_{f \in A} Q_{ef}$ as the maximum quadratic cost associated with leaving arc $e$. Then the QCCP can be formulated as an MILP:

$$(\text{MILP}) \quad \min_{x,y} \sum_{e \in A} y_e$$

s.t. $y_e \geq z_e x_e \quad \forall e \in A$

$y_e \geq Q_{e,}\! x - q_{e}^{\text{max}}(1 - x_e) \quad \forall e \in A$

$x \in X, y \in \mathbb{R}^m.$

If we relax the binary constraint on $x$, then we obtain a lower bound for the QCCP. We call this bound the MILP-Based Bound and we denote its value by $v_{\text{MILP}}$.

### 8 Numerical Results

In this section we test our bounding approaches on a set of test instances and compare them with several approaches from literature. We take into account the linearization based bound $LBB1$ from Section 4.1, the classical GL type bound $GL$ from Section 6.1, the reformulated GL type bound $RGL$, discussed in Section 6.2, the column generation approach $CG$ and the MILP-based bound $MILP$ from Section 7, and the first level RLT bound $RLT1$, see [11, 12]. Note that we do not take into account $LBB2$ and $LBB3$, since our preliminary experiments from Section 4.2 show that $LBB1$ is preferred when taking both quality and efficiency into account.

All lower bounds are implemented in Matlab on a pc with an Intel(R) Core(TM) i5-6500 CPU, 3.20 GHz and 8 GB memory using CPLEX 12.6 as solver.

We consider the following types of instances:

- **Erdős-Rényi instances**: These instances are created via the $G(n, p)$ Erdős-Rényi model, see [11]. The number of nodes is fixed to $n$ and each arc is included with probability $p$ independent of the other arcs. The quadratic cost between any pair of incident arcs is chosen discrete uniformly at random out of $\{0, ... , 100\}$.
• **Manhattan instances:** The Manhattan instances are introduced in [9] and resemble the street pattern of modern cities like Manhattan or Barcelona. Given a finite set of positive integers \((n_1, n_2, \ldots, n_k)\), the graph consists of a \(n_1 \times n_2 \times \cdots \times n_k\) directed grid. Each node in the interior is adjacent to its \(2k\) direct neighbours. The nodes on the boundary are also incident to the corresponding nodes on the opposite boundary. For each dimension \(k\), the arcs belonging to the same layer of the grid point in the same direction. However, the arcs of two consecutive layers point in the opposite direction. This results in a graph containing a lot of cycles. The quadratic cost between any pair of incident arcs is chosen discrete uniformly at random out of \(\{0, \ldots, 10\}\).

• **Angle-distance instances:** The Angle-distance instances are originally constructed for the QTSP in [18]. The number of nodes \(n\) and the graph density \(p\) are given. The \((x,y)\)-coordinates of each node is chosen discrete uniformly at random out of \(\{0, \ldots, 500\}^2\). Exactly \(\lceil pn(n-1)\rceil\) arcs are chosen uniformly at random from the total set of arcs. For each arc \(e\), let \(d_e\) denote the Euclidean distance between the endpoints of \(e\). Moreover, for each two incident arcs \(e\) and \(f\), let \(\alpha_{ef}\) denote the turning angle (in radians). Given some constant \(\rho \in \mathbb{R}_+\), the quadratic cost of two incident arcs \(e\) and \(f\) is calculated as:

\[
Q_{ef} = \left\lceil 0.1 \left( \rho \cdot \alpha_{ef} + \frac{d_e + d_f}{2} \right) \right\rceil.
\]

Similar as in [18], we take \(\rho = 40\).

For Erdős-Rényi and Angle-distance instances, preliminary experiments show that instances up to approximately 300 arcs can be solved to optimality within one hour. For the Manhattan instances the limit is around 2000 arcs, due to the small density of these types of graphs.

In total we consider two sets of experiments: experiments on small instances and experiments on large instances. Since the optimum, \(RLT_1\) and \(CG\) cannot be calculated for large instances, we only test these approaches on the smaller instances. Moreover, we include the bounds introduced in this paper, namely \(LBB_1\) and \(RGL\). The value and computation times (in seconds) on small Erdős-Rényi instances can be found in Table 4. This table contains 6 instances for \(n = 20, 25, 30\) and \(p = 0.3, 0.5\). The results on Manhattan and Angle-distance instances are reported in Table 5 and 6 respectively. For the Angle-distance instances we take the same values for \(n\) and \(p\) as for the Erdős-Rényi instances, while for the Manhattan instances we consider several two- and three-dimensional instances. The maximum computation time is set to 3600 seconds. When after this time no bound is computed, we report ‘n.a.’ in the tables. Since the optimal value is always integer, we round up all bounds.

For the smaller instances, we see that \(RLT_1\) performs best in quality. When it can be computed, it is often close to the optimal solution and it dominates the other bounds. \(LBB_1\) is often very close to \(RLT_1\), but can be computed much more efficient. Namely, for the Erdős-Rényi and the Angle-distance instances the computation time of \(LBB_1\) for all small instances is below 0.4 seconds, whereas \(RLT_1\) cannot be computed within one hour for some of these instances. The column generation approach provides strong bounds, but in most cases it is not able to compute a bound in a time span of one hour. The reformulated \(GL\) type bound performs well on the Manhattan and Angle-distance instances, see Table 5 and 6. Although its total computation time is large, the advantage of this approach is that it provides a bound in a short time and then iteratively improves the value. This makes it possible to stop the procedure at any given time and still obtain a bound. The bounds computed by \(RGL\) are in all cases dominated by \(LBB_1\).

When taking both efficiency and quality into account, we conclude that the linearization based bound \(LBB_1\) outperforms the other approaches. Based on Table 4, 5 and 6, the value of \(LBB_1\) is at least 75% of the optimal value for the Erdős-Rényi instances. For the
Angle-distance and Manhattan instances, this percentage equals 98% and 96%, respectively.

For the larger instances, we only compute the bounds that can be computed efficiently. That is, we do not consider the iterative approaches, but only the bounds GL, MILP and LBB1. We also investigate the effect of a reformulation by adding an optimal incident skew symmetric matrix to the incident entries of the cost matrix, see Section 4.1. We apply this reformulation to LBB1, which implies that we optimize over the set $\tau_{1}^{skew}(Q)$, see (7), instead of $\tau_{1}(Q)$. The resulting bound is denoted by LBB1$^{skew}$. For the Manhattan instances this bound is omitted, since preliminary experiments showed that this reformulation does not improve the bounds for most Manhattan instances. This could be due to the sparsity of Manhattan instances. The bounds and computation times (in seconds) for the Erdős-Rényi, Manhattan and Angle-distance instances are reported in Table 7, 8 and 9, respectively. For the Erdős-Rényi and Angle-distance instances we take for $n$ values between 30 and 100 nodes and consider $p = 0.3$ and $p = 0.5$. For the Manhattan instances we consider large two-dimensional instances and one large three-dimensional instance. The maximum computation time for these bounds is set to 1800 seconds. Again, we round up all values.

### Table 4: Bounds and computation times of $RLT_{1}$, $CG$, $LBB_{1}$ and $RGL$ on small Erdős-Rényi instances.

| $p$ | $n$ | $m$ | $OPT$ | $RLT_{1}$ | $CG$ | $LBB_{1}$ | $RGL$ |
|-----|-----|-----|-------|----------|------|----------|-------|
|     |     |     | value | time     | value | time     | value | time |
| 0.3 | 20  | 119 | 319   | 10.28    | 301   | 4.825    | 289   | 102.3|
|     | 25  | 177 | 386   | 19.04    | 331   | 20.09    | 331   | 928.9|
|     | 30  | 280 | n.a.  | 3600     | 284   | 70.62    | n.a.  | 3600 |
| 0.5 | 20  | 195 | 236   | 211.0    | 181   | 10.28    | 175   | 129.0|
|     | 25  | 327 | n.a.  | 3600     | 141   | 82.52    | n.a.  | 3600 |
|     | 30  | 442 | n.a.  | 3600     | 168   | 385.0    | n.a.  | 3600 |

### Table 5: Bounds and computation times of $RLT_{1}$, $CG$, $LBB_{1}$ and $RGL$ on small Manhattan instances.

### Table 6: Bounds and computation times of $RLT_{1}$, $CG$, $LBB_{1}$ and $RGL$ on small Angle-distance instances.
For the larger instances, we see that LBB1 in all cases dominates GL and MILP in both quality and efficiency. The difference in quality is most present for the Erdős-Rényi instances, see Table 7. For the Manhattan instances, we see that GL and MILP can be calculated efficiently for instances up to 3000 arcs. However, LBB1 remains efficient even for larger instances. In particular, bounds for Manhattan instances up to 15000 arcs can be computed within 60 seconds.

Moreover, we conclude from Table 7 and 9 that the addition of an incidence skew symmetric matrix to the set $\tau_1(Q)$ only improves the bounds for some of the instances. In general, it turns out that the Erdős-Rényi instances can successfully be improved by this method, whereas for the Angle-distance instances only in a few cases there is an improvement. Although the computation times of LBB1 skew are larger than those of LBB1, bounds can still be computed in a reasonable time span.

| p  | n   | m   | value | time | value | time | value | time |
|----|-----|-----|-------|------|-------|------|-------|------|
| 0.3| 30  | 284 | 1.272 | 0.254| 230   | 0.083| 232   | 0.766|
|    | 40  | 468 | 0.645 | 0.565| 265   | 0.179| 278   | 1.711|
|    | 50  | 754 | 1.598 | 1.430| 267   | 0.404| 274   | 4.184|
|    | 60  | 1062| 4.068 | 3.283| 272   | 0.726| 272   | 8.048|
|    | 70  | 1481| 8.910 | 7.896| 255   | 1.660| 258   | 15.38 |
|    | 80  | 1842| 14.26 | 11.89 |263   | 2.740| 267   | 24.67 |
|    | 90  | 2385| 23.25 | 22.54 |259   | 5.296| 261   | 41.74 |
| 0.5| 30  | 434 | 0.557 | 0.456| 161   | 0.182| 163   | 9.218 |
|    | 40  | 793 | 1.607 | 1.367| 166   | 0.554| 169   | 10.38 |
|    | 50  | 1197| 4.323 | 3.827| 165   | 1.185| 167   | 15.74 |

Table 7: Bounds and computation times of GL, MILP, LBB1 and LBB1 skew on large Erdős-Rényi instances.

| Instance | n   | m   | value | time | value | time | value | time |
|----------|-----|-----|-------|------|-------|------|-------|------|
| (20, 20) | 400 | 800 | 1237  | 5.31 | 1493  | 5.197| 1537  | 0.100|
| (30, 30) | 900 | 1800| 2813  | 56.24| 3375  | 55.97| 3517  | 0.410|
| (40, 40) | 1600| 3200| 5101  | 346.8| 6055  | 345.6| 6302  | 1.388|
| (50, 50) | 2500| 5000| 7983  | 1225.3|9524  | 1220.5|9828  | 2.838|
| (17, 17) | 4913| 14739| n.a. | 1800 | n.a. | 1800| 15398 | 54.79|

Table 8: Bounds and computation times of GL, MILP and LBB1 on large Manhattan instances.
Table 9: Bounds and computation times of \textit{GL}, \textit{MILP}, \textit{LBB}_1 and \textit{LBB}^{*skew} on Angle-distance instances.

| p   | n   | m   | \textit{GL} | \textit{MILP} | \textit{LBB}_1 | \textit{LBB}^{*skew} |
|-----|-----|-----|-------------|---------------|----------------|----------------------|
|     |     |     | value       | time          | value         | time                | value       | time          | value         | time          |
| 0.3 | 30  | 261 | 0.238       | 0.232         | 0.054        | 0.054              | 6.957      | 6.957         |
| 40  | 468 | 507 | 0.093       | 0.512         | 0.463        | 0.463              | 7.795      | 7.795         |
| 50  | 735 | 622 | 1.567       | 1.148         | 0.317        | 0.317              | 9.982      | 9.982         |
| 60  | 1062| 609 | 3.684       | 2.599         | 0.694        | 0.694              | 13.97      | 13.97         |
| 70  | 1449| 656 | 7.436       | 5.885         | 1.331        | 1.331              | 21.63      | 21.63         |
| 80  | 1896| 749 | 13.03       | 11.16         | 2.613        | 2.613              | 31.96      | 31.96         |
| 90  | 2403| 815 | 20.33       | 19.35         | 4.838        | 4.838              | 48.81      | 48.81         |
| 100 | 2970| 810 | 30.35       | 29.94         | 12.50        | 12.50              | 78.62      | 78.62         |
| 0.5 | 30  | 435 | 0.516       | 0.406         | 0.168        | 0.168              | 16.59      | 16.59         |
| 40  | 780 | 411 | 1.156       | 1.178         | 0.474        | 0.474              | 16.21      | 16.21         |
| 50  | 1225| 466 | 4.159       | 3.582         | 0.353        | 0.353              | 22.34      | 22.34         |

9 Conclusion

In this paper we consider the linearization problem for the QCCP and its applications. We provide several sufficient conditions for linearizability, and show how these conditions can be used to obtain strong lower bounds for the QCCP. The linearization based bound \textit{LBB}_1, resulting from the incident weak sum property, is the most efficient LBB in terms of complexity and quality, see Table 2. We show here that the GL type bound for the QCCP also belongs to the family of linearization based bounds, see Theorem 4, by providing the appropriate sufficient condition, see Proposition 7.

The first level RLT bounds and/or the GL type bounds are the only linearization based bounds for quadratic binary optimization problems that are implemented for various binary quadratic optimization problems up to date. This paper shows potential in other than those two linearization based bounds.

Here, we also present how each sufficient condition can be used in an iterative bounding procedure. In particular, we introduce a new reformulation technique in which we search for the best equivalent representation of the residual cost matrix and its optimal linearizable matrix, see Algorithm 4. We show how the resulting iterative procedure computes the best GL type bound in each iteration. Our approach outperforms known iterative bounding procedures that use the GL type bounds, see Table 3. Moreover, we prove that the resulting linearization vectors in each step satisfy the constant value property, see Theorem 5.

Finally, our numerical results show that our approach outperforms several other bounds from literature if we take into account both quality and efficiency. Although the linearization based bounds \textit{LBB}_1 are dominated by the well known first level RLT bounds, they can be computed extremely fast. For the Manhattan instances, \textit{LBB}_1 bounds for instances up to 15000 arcs can be computed within 60 seconds. However, other approaches fail to provide bounds for instances of this large sizes.

We expect that similar bounding procedures can be successfully applied for other quadratic optimization problems, such as the quadratic assignment problem, the quadratic minimum spanning tree, and the quadratic traveling salesman problem. However, this is a subject of our future research.

References

[1] W.P. Adams and H.D. Sherali. A tight linearization and an algorithm for zero-one quadratic programming problems. \textit{Management Science}, 32(10):1274-1290, 1986.

[2] W.P. Adams and H.D. Sherali. Linearization strategies for a class of zero-one mixed integer programming problems. \textit{Operations Research}, 38(2):217-226, 1990.
[3] A. Aggarwal, D. Coppersmith, S. Khanna, R. Motwani and B. Schieber. The angular-
metric traveling salesmans problem. *SIAM Journal on Computing*, 29:697–711, 1999.

[4] E. Amaldi, G. Galbiati and F. Maffioli. On minimum reload cost paths, tours and flows.
*Networks*, 57:254–260, 2011.

[5] R. Burkard, M. Dell’Amico and S. Martello. Assignment Problems, *SIAM*, Philadelphia,
USA, 2009.

[6] Y. Büyükçolak, D. Gözüpek and S. Özkan. Minimum reload cost cycle cover in complete
graphs, [arXiv:1706.05225](https://arxiv.org/abs/1706.05225), 2017.

[7] P. Carrera, P. Fêluci and F. Malucelli. A new lower bound for the quadratic assignment problem.
*Operations Research*, 40:22–27, 1992.

[8] S. Chiba and T. Yamashita. On directed 2-factors in digraphs and 2-factors containing
perfect matchings in bipartite graphs. *SIAM Journal of Discrete Mathematics*, 32(1):
394–409, 2018.

[9] F. Comellas, C. Dalfó and M.A. Fiol. Multidimensional Manhattan street networks.
*Journal on Discrete Mathematics*, 22(4):1428–1447, 2008.

[10] A. Ćustić and A.P. Punnen. A characterization of linearizable instances of the quadratic
minimum spanning tree problem. *Journal of Combinatorial Optimization*, 35(2):436–
453, 2018.

[11] P. Erdős and A. Rényi. On random graphs. *Publicationes Mathematicae*, 6(2):290–297,
1959.

[12] G. Galbiati. The complexity of a minimum reload cost diameter problem. *Discrete
Applied Mathematics*, 156:3494–3497, 2008.

[13] G. Galbiati, S. Gualandi, and F. Maffioli. On minimum reload cost cycle cover. *Discrete
Applied Mathematics*, 164:112–120, 2014

[14] I. Gamvros, L. Gouveia, S. Raghavan. Reload cost trees and network design. *Networks*,
59:365–379, 2012.

[15] P.C. Gilmore. Optimal and suboptimal algorithms for the quadratic assignment prob-
lem. *Journal of the Society for Industrial and Applied Mathematics*, 10(2):305–313,
1962.

[16] F. Glover. Improved linear integer programming formulations of nonlinear integer prob-
lems. *Management Science*, 22(4):455–460, 1975.

[17] L. Gourvès, A. Lyra, C. Martinhon and J. Monnot. The minimum reload s-t path, trail
and walk problems. *Discrete Applied Mathematics*, 158(13):1404–1417, 2010.

[18] A. Fischer. A polyhedral study of quadratic traveling salesman problems. *Dissertation*,
Chemnitz University of Technology, 2013.

[19] A. Fischer, F. Fischer, G. Jäger, J. Keilwagen, P. Molitor and I. Grosse. Exact algo-
rithms and heuristics for the quadratic traveling salesman problem with an application
in bioinformatics. *Discrete Applied Mathematics*, 166:87–114, 2014.

[20] H. Hu and R. Sotirov. Special cases of the quadratic shortest path problem. *Journal of
Combinatorial Optimization*, 35(3):754–777, 2018.

[21] H. Hu and R. Sotirov. The linearization problem of binary quadratic problems and its
applications, [arXiv:1802.02426](https://arxiv.org/abs/1802.02426), 2019.
[22] G. Jäger and P. Molitor. Algorithms and experimental study for the traveling salesman problem of second order. Lecture Notes in Computer Science, 5165:211-224, 2008.

[23] S.N. Kabadi and A.P. Punnen. An $O(n^4)$ algorithm for the QAP linearization problem. Mathematics of Operations Research, 36:754–761, 2011.

[24] T.C. Koopmans and M.J. Beckmann. Assignment problems and the location of economic activities. Econometrica, 25:53–76, 1957.

[25] S. Lendl, A. Ćustić, and A.P. Punnen. Combinatorial optimization problems with interaction costs: Complexity and solvable cases. arXiv preprint arXiv:1707.02428, 2017.

[26] E.L. Lawler. The quadratic assignment problem. Management Science, 9(4):586–599, 1963.

[27] F. de Meijer. Bounds on the minimum reload cost cycle cover problem. Master Thesis, Tilburg University, 2018.

[28] A.P. Punnen and S.N. Kabadi. A linear time algorithm for the Koopmans Beckmann QAP linearization and related problems. Discrete Optimization, 10(3): 200–209, 2013.

[29] A.P. Punnen and P. Pandey. Representations of quadratic combinatorial optimization problems: a case study using the quadratic set covering problem. arXiv: 1802.00897v2, 2018.

[30] A.P. Punnen, M. Walter and B.D. Woods. A characterization of linearizable instances of the quadratic traveling salesman problem. arXiv: 1708.07217v3, 2018.

[31] B. Rostami and F. Malucelli. Lower bounds for the quadratic minimum spanning tree problem based on reduced cost computation. Computers and Operations Research, 64:178–188, 2015.

[32] B. Rostami, A. Chassein, M. Hopf, D. Frey, C. Buchheim, F. Malucelli and M. Goerigk. The quadratic shortest path problem: complexity, approximability and solution methods. European Journal of Operations Research, 2018.

[33] R. Stanče, P. Greistorfer, K. Ladner and U. Pferschy. Geometric and LP-based heuristics for the quadratic travelling salesman problem. arXiv: 1803.03681v1, 2018.

[34] H. Wirth and J. Steffan. Reload cost problems: minimum diameter spanning tree. Discrete Applied Mathematics, 113:73–85, 2001.