Parametric Convex Quadratic Relaxation of the Quadratic Knapsack Problem

M. Fampa[a,1], D. Lubke[a], F. Wang[b], H. Wolkowicz[c]

[a] *Universidade Federal do Rio de Janeiro, Brazil*
[b] *Dept. of Mathematics, Royal Institute of Technology, Sweden*
[c] *Dept. of Combinatorics and Optimization, University of Waterloo, Waterloo, Canada*

Abstract

We consider a parametric convex quadratic programming, **CQP**, relaxation for the quadratic knapsack problem, **QKP**. This relaxation maintains partial quadratic information from the original **QKP** by perturbing the objective function to obtain a concave quadratic term. The nonconcave part generated by the perturbation is then linearized by a standard approach that lifts the problem to the matrix space. We present a primal-dual interior point method to optimize the perturbation of the quadratic function, in a search for the tightest upper bound for the **QKP**. We prove that the same perturbation approach, when applied in the context of semidefinite programming, **SDP**, relaxations of the **QKP** , cannot improve the upper bound given by the corresponding linear **SDP** relaxation. The result also applies to more general integer quadratic problems. Finally, we propose new valid inequalities on the lifted matrix variable, derived from cover and knapsack inequalities for the **QKP**, and present the separation problems to generate cuts for the current solution of the **CQP** relaxation. Our best bounds are obtained from alternating between optimizing the parametric quadratic relaxation over the perturbation and adding cutting planes generated by the valid inequalities proposed.

Keywords: quadratic knapsack problem, quadratic binary programming, convex quadratic programming relaxations, parametric optimization, valid inequalities, separation problem

2000 MSC: 90C26, 90C25, 65K05

Email addresses: `fampa@cos.ufrj.br` (M. Fampa), `danielalubke@cos.ufrj.br` (D. Lubke), `fewa@kth.se` (F. Wang), `hwolkowicz@uwaterloo.ca` (H. Wolkowicz)

[1] Corresponding author: Marcia Fampa, PESC/COPPE, Universidade Federal do Rio de Janeiro, Caixa postal 68511, Rio de Janeiro, RJ, 21941-972, Brazil. Email: fampa@cos.ufrj.br

1. Introduction

We study a convex quadratic programming, **CQP**, relaxation of the quadratic knapsack problem, **QKP**,

$$
\text{(1)} \qquad (\mathbf{QKP}) \qquad \begin{array}{rll} p^*_{\text{QKP}} := & \max & x^TQx \\ & \text{s.t.} & w^Tx \le c \\ & & x \in \{0,1\}^n, \end{array}
$$

where $Q \in \mathbb{S}^n$ is a symmetric $n \times n$ nonnegative integer profit matrix, $w \in \mathbb{Z}^n_{++}$ is the vector of positive integer weights for the items, and $c \in \mathbb{Z}_{++}$ is the knapsack capacity with $c \ge w_i$, for all $i \in N := \{1, \ldots, n\}$. The binary variable x indicates whether an item is chosen for the knapsack or not, and the inequality in the model, known as a knapsack inequality, ensures that the selection of items does not exceed the knapsack capacity. We note that any linear costs in the objective can be included on the diagonal of Q by exploiting the $\{0,1\}$ constraints and, therefore, are not considered.

The **QKP** was introduced in [12] and was proved to be NP-Hard in the strong sense by reduction from the clique problem. The quadratic knapsack problem is a generalization of the knapsack problem, **KP**, which has the same feasible set of the **QKP**, and a linear objective function in x. The **KP** can be solved in pseudo-polynomial time using dynamic programming approaches with complexity of $O(nc)$.

The **QKP** appears in a wide variety of fields, such as biology, logistics, capital budgeting, telecommunications and graph theory, and has received a lot of attention in the last decades. Several papers have proposed branch-and-bound algorithms for the **QKP** and the main difference between them is the method used to obtain upper bounds for the subproblems [7, 4, 6, 5, 15, 16]. The well known trade-off between the strength of the bounds and the computational effort required to obtain them is intensively discussed in [24], where semidefinite programming, **SDP**, relaxations proposed in [15] and [16] are presented as the strongest relaxations for the **QKP**. The linear programming, **LP**, relaxation proposed in [4], on the other side, is presented as the most computationally inexpensive.

Both the **SDP** and the **LP** relaxations have a common feature, they are defined in the symmetric matrix lifted space determined by the equation $X = xx^T$, and by the replacement of the quadratic objective function in (1) with a linear function in X, namely, $\text{trace}(QX)$. As the constraint $X = xx^T$ is nonconvex, it is relaxed by convex constraints in the relaxations. The well known McCormick inequalities [21], and also the semidefinite constraint, $X - xx^T \succeq 0$, have been extensively used to relax the nonconvex constraint $X = xx^T$, in relaxations of the **QKP**.

In this paper, we investigate a convex quadratic programming, **CQP**, relaxation for the **QKP**, where instead of linearizing the objective function, we perturb the objective function Hessian Q, and maintain the (concave) perturbed version of the quadratic function in the objective, linearizing only the remaining part derived from the perturbation. Our relaxation is a parametric convex quadratic problem, defined as a function of a matrix parameter Q_p, such that $Q - Q_p \preceq 0$. A similar approach to handle nonconvex quadratic functions consists in decomposing it as a difference of convex (DC) quadratic function [18]. DC decompositions have been extensively used in the literature to generate convex quadratic relaxations of nonconvex quadratic problems. See, for example, [10] and references therein. Unlike the approach used in DC decompositions, we do not necessarily decompose Q as a difference of convex functions, or equivalently, as a sum of a convex and a concave function. Instead, we decompose it as a sum of a concave function and a quadratic term derived from the perturbation applied to Q. This perturbation can be any symmetric matrix Q_p, which is iteratively optimized by a primal-dual interior point method, **IPM**, to generate the best possible bound for the **QKP**.

Although **SDP** relaxations are well known for being more expensive to solve in general, in an attempt to obtain even stronger bounds, we also investigated the parametric convex quadratic **SDP** problem, where we add to our **CQP** relaxation, the positive semidefinite constraint $X - xx^T \succeq 0$. An **IPM** could also be applied to this parametric problem in order to generate the best possible bound. Nevertheless, we prove an interesting result concerning the relaxations, in case the constraint $X - xx^T \succeq 0$ is imposed: the tightest bound generated by the parametric quadratic relaxation is obtained when the perturbation Q_p is equal to Q, or equivalently, when we linearize all the objective function, getting the standard linear **SDP** relaxation. We conclude, therefore, that keeping the (concave) perturbed version of the quadratic function in the objective of the **SDP** relaxation does not lead to a tighter bound.

Another contribution of this work is the development of valid inequalities for the **CQP** relaxation on the lifted matrix variable. The inequalities are first derived from cover inequalities for the **KP**, addressed in the next subsection. The idea is then extended to knapsack inequalities. Taking advantage of the lifting $X := xx^T$, we propose new valid inequalities that can also be applied to more general relaxations of binary quadratic programming problems that use the same lifting. We discuss how cuts for the quadratic relaxation can be obtained by the solution of separation problems, and investigate possible dominance relation between the inequalities proposed.

We finally present an algorithmic framework, where we iteratively improve the upper bound for the **QKP** by optimizing the choice of the perturbation of the objective function and adding cutting planes to the relaxation. At each iteration, lower bounds for the problem are also generated from feasible solutions constructed from a rank-one approximation of the solution of the **CQP** relaxation.

In Section 2, we introduce our parametric convex quadratic relaxation for the **QKP**. In Section 3, we explain how we optimize the parametric problem over the perturbation of the objective, i.e., we present the **IPM** applied to obtain the perturbation that leads to the best possible bound. In Section 4, we present our conclusion about the parametric quadratic **SDP** relaxation. In Section 5, we introduce new valid inequalities on the lifted matrix variable of the convex quadratic model, and we describe how cutting planes are obtained by the solution of separation problems. In Section 6, we present the heuristic procedure used to generate lower bounds to the **QKP** . In Section 7, we discuss our numerical experiments and in Section 8, we present our final remarks.

1.1. Preliminaries: knapsack polytope and cover inequalities

In the following we recall the concepts of knapsack polytopes and cover inequalities.

The knapsack polytope is the convex hull of the feasible points of the **KP** , **KF** $:= \{x \in \{0,1\}^n : w^T x \leq c\}$.

Definition 1 (zero-one knapsack polytope)**.**

$$\mathbf{KPol} := \mathrm{conv}(\mathbf{KF}) = \mathrm{conv}\{x \in \{0,1\}^n : w^T x \leq c\}.$$

Proposition 2. The dimension

$$\dim(\mathbf{KPol}) = n,$$

and **KPol** is an independence system, i.e.,

$$x \in \mathbf{KPol}, y \in \{0,1\}^n, y \leq x \implies y \in \mathbf{KPol}.$$

Proof. Recall that $w_i \leq c, \forall i$. Therefore, all the unit vectors $e_i \in \mathbb{R}^n$ are feasible and the first statement follows. The second statement is clear. ☐

Cover inequalities were originally presented in [2, 26]; see also [23, Section II.2]. These inequalities can be used in general optimization problems with binary variables and, particularly, in the knapsack problems, **KP** and **QKP** .

Definition 3 (cover inequality, **CI**)**.** The subset $C \subseteq N$ is a cover if it satisfies

$$\sum_{j \in C} w_j > c.$$

The (valid) **CI** is

$$\sum_{j \in C} x_j \leq |C| - 1. \tag{2}$$

The cover inequality is minimal if no proper subset of C is also a cover.

Definition 4 (extended **CI**, **ECI**)**.** Let $w^* := \max_{j \in C} w_j$ and define the extension of C as

$$E(C) := C \cup \{j \in N \backslash C : w_j \geq w^*\}.$$

The **ECI** is

$$\sum_{j \in E(C)} x_j \leq |C| - 1.$$

Definition 5 (lifted **CI**, **LCI**)**.** Given any minimal cover C, there exists at least one facet-defining lifted **CI** , **LCI**of the form

$$\sum_{j \in C} x_j + \sum_{j \in N \setminus C} \alpha_j x_j \leq |C| - 1, \tag{3}$$

where $\alpha_j \geq 0, \forall j \in N \setminus C$. Moreover, each such **LCI** dominates the extended **CI**.

Cover inequalities are extensively discussed in [14, 3, 2, 26, 23, 1]. Details about the computational complexity of **LCI** is presented in [28, 13]. Algorithm 1 [27, page 5], shows how to derive a facet-defining **LCI** from a given minimal cover C.

Algorithm 1: Procedure to find **LCI**

Sort the elements in ascending w_i order $i \in N \setminus C$, defining $\{i_1, i_2, \ldots, i_r\}$.
For: t=1 **to** r

$$\begin{array}{llll} & \zeta_t = & \max & \sum_{j=1}^{t-1} \alpha_{i_j} x_{i_j} + \sum_{i \in C} x_i \\ (4) & & \text{s.t.} & \sum_{j=1}^{t-1} w_{i_j} x_{i_j} + \sum_{i \in C} w_i x_i \leq c - w_{i_t} \\ & & & x \in \{0,1\}^{|C|+t-1}. \end{array}$$

Set $\alpha_{i_t} = |C| - 1 - \zeta_t$.

End

Notation

If $A \in \mathbb{S}^n$, then $\text{svec}(A)$ is a vector whose entries come from A by stacking up its lower half, i.e.,

$$\text{svec}(A) := (a_{11}, \ldots, a_{n1}, a_{22}, \ldots, a_{n2}, \ldots, a_{nn})^T \in \mathbb{R}^{n(n+1)/2} .$$

The operator sMat is the inverse of svec, i.e., $\text{sMat}(\text{svec}(A)) = A$.

We also denote by $\lambda_{\min}(A)$, the smallest eigenvalue of A and by $\lambda_i(A)$ the i^{th} largest eigenvalue of A.

2. A Parametric Convex Quadratic Relaxation

In order to construct a convex relaxation for the **QKP**, we start by considering the following standard reformulation of the problem in the lifted space of symmetric matrices, defined by the lifting $X := xx^T$.

$$(5) \qquad (\mathbf{QKP}_{\text{lifted}}) \qquad \begin{array}{rl} p^*_{\text{QKP}_{\text{LIFTED}}} := \max & \operatorname{trace}(QX) \\ \text{s.t.} & w^T x \le c \\ & X = xx^T \\ & x \in \{0,1\}^n. \end{array}$$

We consider an initial **LP** relaxation of the **QKP** , given by

$$(6) \qquad (\mathbf{LPR}) \qquad \begin{array}{ll} \max & \operatorname{trace}(QX) \\ \text{s.t.} & (x, X) \in \mathcal{P}, \end{array}$$

where $\mathcal{P} \subset [0,1]^n \times \mathbb{S}^n$ is a bounded polyhedron, such that

$$\{(x,X) \ : \ w^T x \le c, \ X = xx^T, \ x \in \{0,1\}^n\} \subset \mathcal{P}.$$

2.1. The perturbation of the quadratic objective

We then propose a convex quadratic relaxation with the same feasible set of **LPR** , but maintaining a concave perturbed version of the quadratic objective function of the **QKP**, and linearizing only the remaining nonconcave part derived from the perturbation. More specifically, we choose $Q_p \in \mathbb{S}^n$ such that

$$(7) \qquad Q - Q_p \preceq 0,$$

and get

$$\begin{aligned} x^T Q x &= x^T (Q - Q_p) x + x^T Q_p x = x^T (Q - Q_p) x + \operatorname{trace}(Q_p x x^T) \\ &= x^T (Q - Q_p) x + \operatorname{trace}(Q_p X). \end{aligned}$$

Finally, we define the parametric convex quadratic relaxation of the **QKP** :

$$(8) \qquad (\mathbf{CQP}_{Q_p}) \qquad \begin{array}{rl} p^*_{\text{CQP}}(Q_p) := \max & x^T (Q - Q_p) x + \operatorname{trace}(Q_p X) \\ \text{s.t.} & (x, X) \in \mathcal{P}, \end{array}$$

3. Optimizing the parametric problem over the parameter Q_p

The upper bound $p^*_{\text{CQP}}(Q_p)$ in the convex quadratic problem (8) depends on the feasible perturbation Q_p of the Hessian Q. To improve the upper bound we consider the parametric problem

$$(9) \qquad param^*_{\text{QKP}} := \min_{Q - Q_p \preceq 0} p^*_{\text{CQP}}(Q_p).$$

We solve (9) with a primal-dual interior-point approach, and describe in this section how the search direction of the algorithm is obtained at each iteration. We start with minimizing a log-barrier function. We use the barrier function, $B_\mu(Q_p, Z)$ with barrier parameter, $\mu > 0$, to obtain the barrier problem

$$(10) \qquad \begin{array}{ll} \min & B_\mu(Q_p, Z) := p^*_{\text{CQP}}(Q_p) - \mu \log\det Z \\ \text{s.t.} & Q - Q_p + Z = 0 \qquad\qquad (:\Lambda) \\ & Z \succ 0, \end{array}$$

where $\Lambda \in \mathbb{S}^n$ denotes the dual variable (matrix). Let us consider the Lagrangian function

$$L_\mu(Q_p, Z, \Lambda) := p^*_{\text{CQP}}(Q_p) - \mu \log\det Z + \operatorname{trace}((Q - Q_p + Z)\Lambda).$$

Note that the objective function for $p^*_{\text{CQP}}(Q_p)$ is linear in Q_p, i.e., this function is the maximum of linear functions over feasible points x, X. Therefore, this is a convex function. From standard sensitivity analysis results, e.g. [11, Corollary 3.4.2],[17], [9, Theorem 1], if the optimal solution x, X is unique, then the gradient is obtained by differentiating the Lagrangian. Since Q_p appears only in the objective function in (8), and

$$x^T(Q - Q_p)x + \operatorname{trace}(Q_p X) = x^T Q x + \operatorname{trace}(Q_p(X - xx^T)),$$

we get a directional derivative at Q_p in the direction ΔQ_p,

$$D(p^*_{\text{CQP}}(Q_p); \Delta Q_p) = \max_{\text{optimal } x, X} \operatorname{trace}((X - xx^T)\Delta Q_p).$$

In the case of a unique optimum $x = x(Q_p), X = X(Q_p)$, we get the gradient

$$\nabla p^*_{\text{CQP}}(Q_p) = X - xx^T. \tag{11}$$

The gradient of the barrier function, is then

$$\nabla B_\mu(Q_p) = (X - xx^T) - \mu Z^{-1}.$$

The optimality conditions for (10) are obtained by differentiating the Lagrangian L_μ with respect to Q_p, Λ, Z, respectively,

$$\begin{array}{rrcl} \frac{\partial}{\partial Q_p}: & \nabla p^*_{\text{CQP}}(Q_p) - \Lambda & = & 0, \\ \frac{\partial}{\partial \Lambda}: & Q - Q_p + Z & = & 0, \\ \frac{\partial}{\partial Z}: & -\mu Z^{-1} + \Lambda & = & 0, \quad \text{(or) } Z\Lambda - \mu I = 0. \end{array} \tag{12}$$

This gives rise to the nonlinear overdetermined system

$$G_\mu(Q_p, \Lambda, Z) = \begin{pmatrix} \nabla p^*_{\text{CQP}}(Q_p) - \Lambda \\ Q - Q_p + Z \\ Z\Lambda - \mu I \end{pmatrix} = 0, \qquad Z, \Lambda \succ 0. \tag{13}$$

We use a BFGS approximation for the Hessian of p^*_{CQP}, as if it is twice differentiable, and update it at each iteration (see [20]). We denote the approximation of $\nabla^2_{\text{BFGS}} p^*_{\text{CQP}}(Q_p)$ by B, and begin with the approximation $B_0 = I$. Recall that if Q_p^k, Q_p^{k+1} are two successive iterates with gradients $\nabla p^*_{\text{CQP}}(Q_p^k), \nabla p^*_{\text{CQP}}(Q_p^{k+1})$, respectively, with current Hessian approximation $B_k \in \mathbb{S}^{n(n+1)/2}$, then we set

$$Y_k := \nabla p^*_{\text{CQP}}(Q_p^{k+1}) - \nabla p^*_{\text{CQP}}(Q_p^k), \qquad S_k := Q_p^{k+1} - Q_p^k,$$

and,

$$\upsilon := \langle Y_k, S_k \rangle, \qquad \omega := \langle \operatorname{svec}(S_k), B_k \operatorname{svec}(S_k) \rangle.$$

We note that the curvature condition $\upsilon > 0$ should be verified.

Finally, we update the Hessian approximation with

$$B_{k+1} := B_k + \frac{1}{\upsilon}\left(\operatorname{svec}(Y_k)\operatorname{svec}(Y_k^T)\right) - \frac{1}{\omega}\left(B_k \operatorname{svec}(S_k)\operatorname{svec}(S_k)^T B_k\right).$$

The overdetermined equation for the search direction is

$$G'_\mu(Q_p, \Lambda, Z)\begin{pmatrix}\Delta Q_p \\ \Delta\Lambda \\ \Delta Z\end{pmatrix} = -G_\mu(Q_p, \Lambda, Z), \tag{14}$$

where

$$G_\mu(Q_p, \Lambda, Z) = \begin{pmatrix}\nabla p^*(Q_p) - \Lambda \\ Q - Q_p + Z \\ Z\Lambda - \mu I\end{pmatrix} =: \begin{pmatrix}R_d \\ R_p \\ R_c\end{pmatrix}. \tag{15}$$

If B is the current estimate of the Hessian, then the system becomes

$$\begin{cases} \operatorname{sMat}(B \operatorname{svec}(\Delta Q_p)) - \Delta\Lambda = -R_d, \\ -\Delta Q_p + \Delta Z = -R_p, \\ Z\Delta\Lambda + \Delta Z\Lambda = -R_c. \end{cases}$$

We can substitute for the variables $\Delta\Lambda$ and ΔZ in the third equation of the system. We note that, as the system is overdetermined, this substitution changes the least squares solution. Nevertheless, elimination gives us a simplified system, and therefore, we apply it, using the following two equations for elimination and backsolving,

$$\Delta\Lambda = \operatorname{sMat}(B \operatorname{svec}(\Delta Q_p)) + R_d, \quad \Delta Z = -R_p + \Delta Q_p. \tag{16}$$

Accordingly, we have a single equation to solve, and the system finally becomes

$$Z \operatorname{sMat}(B \operatorname{svec}(\Delta Q_p)) + (\Delta Q_p)\Lambda = -R_c - ZR_d + R_p\Lambda.$$

We emphasize that to compute the search direction at each iteration of our **IPM**, we need to update the residuals defined in (15), and therefore we need the optimal solution $x = x(Q_p), X = X(Q_p)$ of the convex quadratic relaxation **CQP** $_{Q_p}$ for the current perturbation Q_p. Problem **CQP** $_{Q_p}$ is thus solved at each iteration of the **IPM** method, each time for a new perturbation Q_p.

Moreover, we note that at each iteration of the **IPM**, we have $Z \succ 0$ and $Q - Q_p \prec 0$. Problem **CQP** $_{Q_p}$ then maximizes a strictly concave quadratic function, subject to linear constraints, and therefore has a unique optimal solution (see e.g. [25]). The result assures that the gradient in (11) is well defined.

In Algorithm 2, we present in details an iteration of the **IPM**. The algorithm is part of the complete framework used to generate bounds for the **QKP**, as described in Section 7.

Algorithm 2: Updating the perturbation Q_p

Input: k, Q_p^k, Z^k, Λ^k, $x(Q_p^k)$, $X(Q_p^k)$, $\nabla p^*_{\text{CQP}}(Q_p^k)$, B_k, μ^k,
$\tau_\alpha := 0.95$, $\tau_\mu := 0.9$.
Compute the residuals:

$$\begin{pmatrix} R_d \\ R_p \\ R_c \end{pmatrix} := \begin{pmatrix} \nabla p^*_{\text{CQP}}(Q_p^k) - \Lambda^k \\ Q - Q_p^k + Z^k \\ Z^k \Lambda^k - \mu^k I \end{pmatrix}.$$

Solve the linear system for ΔQ_p:

$$Z^k \,\text{sMat}(B_k \,\text{svec}(\Delta Q_p)) + (\Delta Q_p)\Lambda^k = -R_c - Z^k R_d + R_p \Lambda^k.$$

Set:

$$\Delta\Lambda := \text{sMat}(B_k \,\text{svec}(\Delta Q_p)) + R_d, \;\; \Delta Z := -R_p + \Delta Q_p.$$

Update Q_p, Z and Λ:

$$Q_p^{k+1} := Q_p^k + \hat{\alpha}_p \Delta Q_p, \;\; Z^{k+1} := Z_p^k + \hat{\alpha}_p \Delta Z, \;\; \Lambda^{k+1} := \Lambda^k + \hat{\alpha}_d \Delta\Lambda,$$

where

$$\begin{aligned} \hat{\alpha}_p &:= \tau_\alpha \times \min\{1, \text{argmax}_{\alpha_p}\{Z_p^k + \alpha_p \Delta Z \succeq 0\}\}, \\ \hat{\alpha}_d &:= \tau_\alpha \times \min\{1, \text{argmax}_{\alpha_d}\{\Lambda^k + \alpha_d \Delta\Lambda \succeq 0\}\}. \end{aligned}$$

Obtain the optimal solution $x(Q_p^{k+1})$, $X(Q_p^{k+1})$ of relaxation
CQP $_{Q_p}$, where $Q_p := Q_p^{k+1}$.
Update the gradient of p^*_{CQP}:

$$\nabla p^*_{\text{CQP}}(Q_p^{k+1}) := X(Q_p^{k+1}) - x(Q_p^{k+1})x(Q_p^{k+1})^T.$$

Update the Hessian approximation of p^*_{CQP}:

$$\begin{aligned} &Y_k := \nabla p^*_{\text{CQP}}(Q_p^{k+1}) - \nabla p^*_{\text{CQP}}(Q_p^k), \;\; S_k := Q_p^{k+1} - Q_p^k, \\ &\upsilon := \langle Y_k, S_k \rangle, \;\; \omega := \langle \text{svec}(S_k), B_k \,\text{svec}(S_k) \rangle, \\ &B_{k+1} := B_k + \frac{1}{\upsilon}\left(\text{svec}(Y_k)\,\text{svec}(Y_k^T)\right) - \frac{1}{\omega}\left(B_k \,\text{svec}(S_k)\,\text{svec}(S_k)^T B_k\right). \end{aligned}$$

Update μ:

$$\mu^{k+1} := \tau_\mu \frac{\text{trace}(Z^{k+1}\Lambda^{k+1})}{n}.$$

Output: Q_p^{k+1}, Z^{k+1}, Λ^{k+1}, $x(Q_p^{k+1})$, $X(Q_p^{k+1})$, $\nabla p^*_{\text{CQP}}(Q_p^{k+1})$,
B_{k+1}, μ^{k+1}.

4. The parametric quadratic SDP relaxation

In an attempt to obtain tighter bounds, a promising approach could seem to be to include the positive semidefinite constraint $X - xx^T \succeq 0$ in our parametric quadratic relaxation, and solve a parametric convex quadratic **SDP** relaxation, also using an **IPM**. Nevertheless, we show in this section that the convex quadratic **SDP** relaxation cannot generate a better bound than the linear **SDP** relaxation, obtained when we set Q_p equal to Q.

Consider the reformulation $\mathbf{QKP}_{\text{lifted}}$ in (5), of the **QKP**, and its **SDP** relaxation given by

$$(17) \qquad (\mathbf{LSDP}) \qquad \begin{array}{rll} p^*_{\text{LSDP}} := & \sup & \operatorname{trace}(QX) \\ & \text{s.t.} & (x, X) \in \mathcal{F} \\ & & X - xx^T \succeq 0, \end{array}$$

where $\mathcal{F}$ is any relaxation of the feasible set of $\mathbf{QKP}_{\text{lifted}}$.

We now consider the parametric **SDP** relaxation of $\mathbf{QKP}_{\text{lifted}}$ given by

$$(18) \qquad (\mathbf{QSDP}_{Q_p}) \qquad \begin{array}{rll} p^*_{\text{QSDP}_{Q_p}} := & \sup & x^T(Q - Q_p)x + \operatorname{trace}(Q_p X) \\ & \text{s.t.} & (x, X) \in \mathcal{F} \\ & & X - xx^T \succeq 0, \end{array}$$

where $Q - Q_p \preceq 0$.

Theorem 6. Let $\mathcal{F}$ be any subset of $\mathbb{R}^n \times \mathbb{S}^n$. For any choice of matrix Q_p satisfying $Q - Q_p \preceq 0$, we have

$$(19) \qquad p^*_{\text{QSDP}_{Q_p}} \geq p^*_{\text{LSDP}}.$$

Moreover, $\inf\{p^*_{\text{QSDP}_{Q_p}} \;:\; Q - Q_p \preceq 0\} = p^*_{\text{LSDP}}$.

Proof. Let $(\tilde{x}, \tilde{X})$ be a feasible solution for **LSDP**. We have

$$\begin{aligned}
(20) \qquad p^*_{\text{QSDP}_{Q_p}} &\geq \tilde{x}^T(Q - Q_p)\tilde{x} + \operatorname{trace}(Q_p\tilde{X}) \\
&= \operatorname{trace}((Q - Q_p)(\tilde{x}\tilde{x}^T - \tilde{X})) + \operatorname{trace}((Q - Q_p)\tilde{X}) \\
(21) \qquad &\quad + \operatorname{trace}(Q_p\tilde{X}) \\
(22) \qquad &= \operatorname{trace}((Q - Q_p)(\tilde{x}\tilde{x}^T - \tilde{X})) + \operatorname{trace}(Q\tilde{X}) \\
(23) \qquad &\geq \operatorname{trace}(Q\tilde{X}).
\end{aligned}$$

The inequality (20) holds because $(\tilde{x}, \tilde{X})$ is also a feasible solution for $\mathbf{QSDP}_{Q_p}$. The inequality in (23) holds because of the negative semidefiniteness of $Q - Q_p$ and $\tilde{x}\tilde{x}^T - \tilde{X}$. Because $p^*_{\text{QSDP}_{Q_p}}$ is an upper bound on the objective value of **LSDP** at any feasible solution, we can conclude that $p^*_{\text{QSDP}_{Q_p}} \geq p^*_{\text{LSDP}}$. Clearly, $Q_p = Q$ satisfies $Q - Q_p = 0 \preceq 0$ and **LSDP** is the same as **QSDP** for this choice of Q_p. Therefore, $\inf\{p^*_{\text{QSDP}_{Q_p}} \;:\; Q - Q_p \preceq 0\} = p^*_{\text{LSDP}}$. $\square$

Notice that in Theorem 6 we do not require that the relaxation $\mathcal{F}$ be convex nor need it have any relationship at all with the feasible region of **QKP** . Also, in principle, for some choices of Q_p, we could have $p^*_{\text{QSDP}_{Q_p}} = +\infty$ with $p^*_{\text{LSDP}} = +\infty$ or not.

5. Valid inequalities

We are now interested in finding valid inequalities to strengthen relaxations of the **QKP** in the lifted space determined by the lifting $X := xx^T$. Let us denote by **CRel**, any convex relaxation of the **QKP** in the lifted space, where the equation $X = xx^T$ was relaxed somehow, by convex constraints, i.e., any convex relaxation of $\mathbf{QKP}_{\text{lifted}}$.

We initially note that if the inequality

$$\tau^T x \leq \beta \tag{24}$$

is valid for the **QKP**, where $\tau \in \mathbb{Z}^n_+$ and $\beta \in \mathbb{Z}_+$, then, as x is nonnegative and $X := xx^T$,

$$(x\ X) \begin{pmatrix} -\beta \\ \tau \end{pmatrix} \leq 0 \tag{25}$$

is a valid inequality for $\mathbf{QKP}_{\text{lifted}}$. In this case, we say that (25) is a valid inequality for $\mathbf{QKP}_{\text{lifted}}$ derived from the valid inequality (24) for the **QKP**.

5.1. Adding cuts to the relaxation

Given a solution $(\bar{x}, \bar{X})$ of **CRel**, our initial goal is to obtain a valid inequality for $\mathbf{QKP}_{\text{lifted}}$ derived from a **CI**, which is violated by $(\bar{x}, \bar{X})$. A **CI** is formulated as $\alpha^T x \leq e^T \alpha - 1$, where $\alpha \in \{0,1\}^n$ and e denotes the vector of ones. We then search for the **CI** that maximizes the sum of the violations among the inequalities in $\bar{Y}\mathbf{cut}(\alpha) \leq 0$, where $\bar{Y} := \begin{pmatrix} \bar{x}\ \bar{X} \end{pmatrix}$ and

$$\mathbf{cut}(\alpha) = \begin{pmatrix} -e^T\alpha + 1 \\ \alpha \end{pmatrix}.$$

To obtain such **CI** , we solve the following linear knapsack problem,

$$v^* := \max_{\alpha} \{e^T \bar{Y} \mathbf{cut}(\alpha) : w^T\alpha \geq c + 1,\ \alpha \in \{0,1\}^n\}. \tag{26}$$

Let α^* solve (26). If $v^* > 0$, then at least one valid inequality in the following set of n scaled cover inequalities, denoted in the following by **SCI**, is violated by $(\bar{x}, \bar{X})$.

$$(x\ X) \begin{pmatrix} -e^T\alpha^* + 1 \\ \alpha^* \end{pmatrix} \leq 0. \tag{27}$$

Based on the following theorem, we note that to strengthen cut (27), we may apply Algorithm 1 to the **CI** obtained, lifting it to an **LCI** , and finally add the valid inequality (25) derived from the **LCI** to **CRel**.

Theorem 7. The valid inequality (25) for **QKP** $_{\text{lifted}}$, which is derived from a valid **LCI**, dominates all inequalities derived from a **CI** that can be lifted to the **LCI**.

Proof. Consider the **LCI** (3) derived from a **CI** (2) for the **QKP**. The corresponding scaled cover inequalities (25) derived from the **CI** and the **LCI** are, respectively,

$$\sum_{j\in C} X_{ij} \leq (|C|-1)x_i, \;\; \forall i \in N,$$

and

$$\sum_{j\in C} X_{ij} + \sum_{j\in N\setminus C} \alpha_j X_{ij} \leq (|C|-1)x_i, \;\; \forall i \in N,$$

where $\alpha_j \geq 0, \forall j \in N\setminus C$. Clearly, as all X_{ij} are nonnegative, the second inequality dominates the first, for all $i \in N$. □

5.2. New valid inequalities in the lifted space

As discussed, after finding any valid inequality in the form of (24) for the **QKP**, we may add the constraint (25) to **CRel** when aiming at better bounds. We observe now, that besides (25) we can also generate other valid inequalities in the lifted space by taking advantage of the lifting $X := xx^T$, and also of the fact that x is binary. In the following, we show how the idea can be applied to cover inequalities.

Let

$$\sum_{j\in C} x_j \leq \beta, \tag{28}$$

where $C \subset N$ and $\beta < |C|$, be a valid inequality for **KPol**.

Inequality (28) can be either a cover inequality, **CI**, an extended cover inequality, **ECI**, or a particular lifted cover inequality, **LCI**, where $\alpha_j \in \{0, 1\}, \forall j \in N\setminus C$ in (3). Furthermore, given a general **LCI**, where $\alpha_j \in \mathbb{Z}_+$, for all $j \in N\setminus C$, a valid inequality of type (28) can be constructed by replacing each α_j with $\min\{\alpha_j, 1\}$ in the **LCI**.

Definition 8 (Cover inequality in the lifted space, **CILS**)**.** Let $C \subset N$ and $\beta < |C|$ as in inequality (28), and also consider here that $\beta > 1$. We define

$$\sum_{i,j\in C, i<j} X_{ij} \leq \binom{\beta}{2}. \tag{29}$$

as the **CILS** derived from (28).

Theorem 9. If inequality (28) is valid for **QKP**, then the **CILS** (29) is a valid inequality for **QKP** $_{\text{lifted}}$.

Proof. Considering (28), we conclude that at most $\binom{\beta}{2}$ products of variables $x_i x_j$, where $i, j \in C$, can be equal to 1. Therefore, as $X_{ij} := x_i x_j$, the result follows. □

Remark 10. When $\beta > 1$, inequality (28) is well known as a clique cut, widely used to model decision problems, and frequently used as a cut in branch-and-cut algorithms. In this case, using similar idea to what was used to construct the **CILS**, we conclude that it possible to fix

$$X_{ij} = 0, \text{ for all } i, j \in C, i < j.$$

Given a solution $(\bar{x}, \bar{X})$ of **CRel**, the following mixed-integer linear program (**MILP**) is a separation problem, which searches for a **CILS** violated by $\bar{X}$.

$$\begin{array}{lll}
z^* := & \max_{\alpha,\beta,K} \operatorname{trace}(\bar{X}K) - \beta(\beta - 1), & (\textbf{MILP}_1) \\
\text{s.t.} & w'\alpha \geq c + 1, & \\
& \beta = e'\alpha - 1, & \\
& K(i,i) = 0, & i = 1, \dots, n, \\
& K(i,j) \leq \alpha_i, & i,j = 1, \dots, n,\ i < j, \\
& K(i,j) \leq \alpha_j, & i,j = 1, \dots, n,\ i < j, \\
& K(i,j) \geq 0, & i,j = 1, \dots, n,\ i < j, \\
& K(i,j) \geq \alpha_i + \alpha_j - 1, & i,j = 1, \dots, n,\ i < j, \\
& \alpha \in \{0,1\}^n,\ \beta \in \mathbb{R},\ K \in \mathbb{S}^n. &
\end{array}$$

If α^*, β^*, K^* solves **MILP** $_1$, with $z^* > 0$, the **CILS** given by $\operatorname{trace}(K^*X) \leq \beta^*(\beta^* - 1)$ is violated by $\bar{X}$. The binary vector α^* defines the **CI** from which the cut is derived. The **CI** is specifically given by ${\alpha^*}^T x \leq e^T\alpha^* - 1$ and $\beta^*(\beta^* - 1)$ determines the right hand side of the **CILS**. The inequality is multiplied by 2 because we consider the variable K as a symmetric matrix, in order to simplify the presentation of the model.

Theorem 11. The valid inequality **CILS** for **QKP** $_{\text{lifted}}$, which is derived from a valid **LCI** in the form (28), dominates any **CILS** derived from a **CI** that can be lifted to the **LCI**.

Proof. As X is nonnegative, it is straightforward to verify that if X satisfies a **CILS** derived from a **LCI**, X also satisfies any **CILS** derived from a **CI** that can be lifted to the **LCI**. □

Any feasible solution of **MILP** $_1$ such that $\operatorname{trace}(\bar{X}K) > \beta(\beta - 1)$ generates a valid inequality for **QKP** $_{\text{lifted}}$, which is violated by $\bar{X}$. Therefore, we do not need to solve **MILP** $_1$ to optimality to generate a cut. Moreover, to generate distinct cuts, we can solve **MILP** $_1$ several times (not necessarily to optimality), each time adding to it, the following "no-good" cut to avoid the previously generated cuts:

$$\sum_{i \in N} \bar{\alpha}(i)(1 - \alpha(i)) \geq 1, \tag{30}$$

where $\bar{\alpha}$ is the value of the variable α in the solution of **MILP** $_1$, when generating the previous cut.

We note that, if α^*, β^*, K^* solves $\mathbf{MILP}_1$, then ${\alpha^*}'x \leq e'\alpha^* - 1$ is a valid **CI** for our **QKP**, however it may not be a minimal cover. Aiming at generating stronger valid cuts, based in Theorem 11, we might add to the objective function of $\mathbf{MILP}_1$, the term $-\delta e'\alpha$, for some weight $\delta > 0$. The objective function would then favor minimal covers, which could be lifted to a **LCI**, that would finally generate the **CILS**. We should also emphasize that if the **CILS** derived from a **CI** is violated by a given $\bar{X}$, then clearly, the **CILS** derived from the **LCI** will also be violated by $\bar{X}$.

Now, we also note that, besides defining one cover inequality in the lifted space considering all possible pairs of indexes in C, we can also define a set of cover inequalities in the lifted space, considering in each inequality, a partition of the indexes in C into subsets of cardinality 1 or 2. In this case, the right hand side of the inequalities is never larger than $\beta/2$. The idea is better specified below.

Definition 12 (Set of cover inequalities in the lifted space, **SCILS**)**.** Let $C \subset N$ and $\beta < |C|$ as in inequality (28). Let

1. $C_s := \{(i_1, j_1), \ldots, (i_p, j_p)\}$ be a partition of C, if $|C|$ is even.
2. $C_s := \{(i_1, j_1), \ldots, (i_p, j_p)\}$ be a partition of $C \setminus \{i_0\}$ for each $i_0 \in C$, if $|C|$ is odd and β is odd.
3. $C_s := \{(i_0, i_0), (i_1, j_1), \ldots, (i_p, j_p)\}$, where $\{(i_1, j_1), \ldots, (i_p, j_p)\}$ is a partition of $C \setminus \{i_0\}$ for each $i_0 \in C$, if $|C|$ is odd and β is even.

In all cases, $i_k < j_k$ for all $k = 1, \ldots, p$.

The inequalities in the **SCILS** derived from (28) are given by

$$\sum_{(i,j) \in C_s} X_{ij} \leq \left\lfloor \frac{\beta}{2} \right\rfloor, \tag{31}$$

for all partitions C_s defined as above.

Theorem 13. If inequality (28) is valid for **QKP**, then the inequalities in the **SCILS** (31) are valid for $\mathbf{QKP}_{\text{lifted}}$.

Proof. The proof of the validity of **SCILS** is based on the lifting relation $X_{ij} = x_i x_j$. We note that if the binary variable x_i indicates whether or not the item i is selected in the solution, the variable X_{ij} indicates whether or not the pair of items i and j, are both selected in the solution.

1. If $|C|$ is even, C_s is a partition of C in exactly $|C|/2$ subsets with two elements each, and therefore, if at most β elements of C can be selected in the solution, clearly at most $\left\lfloor \frac{\beta}{2} \right\rfloor$ subsets of C_s can also be selected.
2. If $|C|$ and β are odd, C_s is a partition of $C \setminus \{i_0\}$ in exactly $|C - 1|/2$ subsets with two elements each, where i_0 can be any element of C. In this case, if at most β elements of C can be selected in the solution, clearly at most $\frac{\beta-1}{2}\left(= \left\lfloor \frac{\beta}{2} \right\rfloor\right)$ subsets of C_s can also be selected.

3. If $|C|$ is odd and β is even, C_s is the union of $\{(i_0, i_0)\}$ with a partition of $C \setminus \{i_0\}$ in exactly $|C-1|/2$ subsets with two elements each, where i_0 can be any element of C. In this case, if at most β elements of C can be selected in the solution, clearly at most $\frac{\beta}{2}\left(= \left\lfloor \frac{\beta}{2} \right\rfloor\right)$ subsets of C_s can also be selected.

□

Given a solution $(\bar{x}, \bar{X})$ of **CRel**, we now present a **MILP** separation problem, which searches for an inequality in **SCILS** that is most violated by $\bar{X}$. Let $A \in \{0,1\}^{n \times \frac{n(n+1)}{2}}$. In the first n columns of A we have the $n \times n$ identity matrix. In the remaining $n(n-1)/2$ columns of the matrix, there are exactly two elements equal to 1 in each column. All columns are distinct. For example, for $n = 4$,

$$A := \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

The columns of A represent all the subsets of items in N with one or two elements. Let

$$\begin{array}{lll} z^* := & \max_{\alpha,v,K,y} \operatorname{trace}(\bar{X}K) - 2v, & (\textbf{MILP}_2) \\ \text{s.t.} & w'\alpha \geq c+1, & \\ & K(i,i) = 2y(i), & i = 1,\ldots,n, \\ & \sum_{i=1}^{n} y(i) \leq 1, & \\ & K(i,j) = \sum_{t=n+1}^{n(n+1)/2} A(i,t)A(j,t))y(t), & i,j = 1,\ldots,n, i<j, \\ & v \geq (e'\alpha - 1)/2 - 0.5, & \\ & v \leq (e'\alpha - 1)/2, & \\ & y(t) \leq 1 - A(i,t) + \alpha(i), & i = 1,\ldots,n,\ t = 1,\ldots,\frac{n(n+1)}{2}, \\ & \alpha \leq Ay \leq\ \alpha + \left(\frac{n(n+1)}{2}\right)(1-\alpha), & \\ & \alpha \in \{0,1\}^n,\ y \in \{0,1\}^{\frac{n(n+1)}{2}}, & \\ & v \in \mathbb{Z},\ K \in \mathbb{S}^n. & \end{array}$$

If α^*, v^*, K^*, y^* solves **MILP** $_2$, with $z^* > 0$, then the particular inequality in **SCILS** given by

$$\operatorname{trace}(K^* X) \leq 2v^* \tag{32}$$

is violated by $\bar{X}$. The binary vector α^* defines the **CI** from which the cut is derived. As the **CI** is given by $\alpha^* x \leq e'\alpha^* - 1$, we can conclude that the cut generated either belongs to case (1) or (3) in Definition 12. This fact is considered in the formulation of **MILP** $_2$. The vector y^* defines a partition C_s as presented in case (3), if $\sum_{i=1}^{n} y(i) = 1$, and in case (1), otherwise. We finally note that the number 2 in the right hand side of (32) is due to the symmetry of the matrix K^*.

We now may repeat the observations made for **MILP** $_1$.

Any feasible solution of $\mathbf{MILP}_2$ such that $\text{trace}(\bar{X}K) > 2v$ generates a valid inequality for **CRel**, which is violated by $\bar{X}$. Therefore, we do not need to solve $\mathbf{MILP}_2$ to optimality to generate a cut. Moreover, to generate distinct cuts, we can solve $\mathbf{MILP}_2$ several times (not necessarily to optimality), each time adding to it, the following suitable "no-good" cut to avoid the previously generated cuts:

$$\sum_{i=1}^{\frac{n(n+1)}{2}} \bar{y}(i)(1 - y(i)) \geq 1, \tag{33}$$

where $\bar{y}$ is the value of the variable y in the solution of $\mathbf{MILP}_2$, when generating the previous cut.

The **CI** $\alpha^{*\prime}x \leq e'\alpha^* - 1$ may not be a minimal cover. Aiming at generating stronger valid cuts, we might add again to the objective function of $\mathbf{MILP}_2$, the term $-\delta e'\alpha$, for some weight $\delta > 0$. The objective function would then favor minimal covers, which could be lifted to a **LCI**. In this case, however, after computing the **LCI**, we have to solve $\mathbf{MILP}_2$ again, with α fixed at values that represent the **LCI**, and v fixed so that the right hand side of the inequality is equal to the right hand side of the **LCI**. All components of y that were equal to 1 in the previous solution of $\mathbf{MILP}_2$ should also be fixed at 1. The new solution of $\mathbf{MILP}_2$ would indicate the other subsets of N to be added to C_s. One last detail should be taken into account. If the cover C corresponding to the **LCI**, is such that $|C|$ is odd and the right hand side of the **LCI** is also odd, then the cut generated should belong to case (2) in Definition 12, and $\mathbf{MILP}_2$ should be modified accordingly. Specifically, the second and third constraints in $\mathbf{MILP}_2$, should be modified respectively to

$$\begin{array}{ll} K(i,i) = 0, & i = 1, \ldots, n, \\ \sum_{i=1}^{n} y(i) = 1. & \end{array}$$

Remark 14. Let $\gamma := |C|$. Then, the number of inequalities in the **SCILS** is

$$\frac{\gamma!}{2^{(\frac{\gamma}{2})}(\frac{\gamma}{2}!)},$$

if γ is even, or

$$\gamma \times \frac{(\gamma-1)!}{2^{(\frac{\gamma-1}{2})}(\frac{\gamma-1}{2}!)},$$

if γ is odd.

Finally, we extend the ideas presented above to the more general case of knapsack inequalities. We note that the following discussion applies to a general **LCI**, where $\alpha_j \in \mathbb{Z}_+, \forall j \in N \backslash C$.

Let

$$\sum_{j \in N} \alpha_j x_j \leq \beta. \tag{34}$$

be a valid knapsack inequality for **KPol**, with $\alpha_j, \beta \in \mathbb{Z}_+, \beta \geq \alpha_j, \forall j \in N$.

Definition 15 (Set of knapsack inequalities in the lifted space, **SKILS**)**.** Let α_j be the coefficient of x_j in (34). Let $\{C_1, \ldots, C_q\}$ be the partition of N, such that $\alpha_u = \alpha_v$, if $u, v \in C_k$ for some k, and $\alpha_u \neq \alpha_v$, otherwise. The knapsack inequality (34) can then be rewritten as

$$\sum_{k=1}^{q} \left(\tilde{\alpha}_k \sum_{j \in C_k} x_j \right) \leq \beta. \tag{35}$$

Now, for $k = 1, \ldots, q$, let $C_{l_k} := \{(i_{k_1}, j_{k_1}), \ldots, (i_{k_{p_k}}, j_{k_{p_k}})\}$, where $i < j$ for all $(i,j) \in C_{l_k}$, and

- C_{l_k} is a partition of C_k, if $|C_k|$ is even.
- C_{l_k} is a partition of $C_k \setminus \{i_{k_0}\}$, where $i_{k_0} \in C_k$, if $|C_k|$ is odd.

The inequalities in the **SKILS** corresponding to (34) are given by

$$\sum_{k=1}^{q} \left(\tilde{\alpha}_k X_{i_{k_0} i_{k_0}} + 2\tilde{\alpha}_k \sum_{(i,j) \in C_{l_k}} X_{ij} \right) \leq \beta, \tag{36}$$

for all partitions C_{l_k}, $k = 1, \ldots, q$, defined as above, and for all $i_{k_0} \in C_k \setminus C_{l_k}$. (If $|C_k|$ is even, $C_k \setminus C_{l_k} = \emptyset$, and the term in the variable $X_{i_{k_0} i_{k_0}}$ does not exist.)

Remark 16. Consider $\{C_1, \ldots, C_q\}$ as in Definition 15. For $k = 1, \ldots, q$, let $\gamma_k := |C_k|$ and define

$$NC_{l_k} := \frac{\gamma_k!}{2^{(\frac{\gamma_k}{2})} (\frac{\gamma_k}{2}!)},$$

if γ_k is even, or

$$NC_{l_k} := \gamma_k \times \frac{(\gamma_k - 1)!}{2^{(\frac{\gamma_k - 1}{2})} (\frac{\gamma_k - 1}{2}!)},$$

if γ_k is odd.

Then, the number of inequalities in **SKILS** is

$$\prod_{k=1}^{q} NC_{l_k}.$$

Remark 17. If $\gamma_k := |C_k|$ is even for every, or if $\tilde{\alpha}_k$ is even for every k, such that γ_k is odd, then the right rand side β of inequality (36) may be replaced with $2 \times \left\lfloor \frac{\beta}{2} \right\rfloor$, which will strengthener the inequality in case β is odd.

Corollary 18.

If inequality (34) is valid for **QKP**, then the inequalities (36), in the **SKILS**, are valid for **QKP** $_{\text{lifted}}$, whether or not the modification suggested in Remark 17 is applied.

Proof. The result is again verified, by using the same argument used in the proof of Theorem 13, i.e., considering that $X_{ij} = 1$, iff $x_i = x_j = 1$. ☐

5.3. Dominance relation among the new valid inequalities

We start this subsection investigating whether **SCILS** dominates **CILS** or vice versa.

Theorem 19. Let C be the cover in (28) and consider $\gamma := |C|$ to be even.

1. If $\beta = \gamma - 1$, then the sum of all inequalities in **SCILS** is equivalent to **CILS**. Therefore, in this case, the set of inequalities in **SCILS** dominates **CILS**.
2. If $\beta < \gamma - 1$, there is no dominance relation between **SCILS** and **CILS**.

Proof. Let sum(**SCILS**) denote the inequality obtained by adding all inequalities in **SCILS**, and let $rhs(sum(\textbf{SCILS}))$ denote its right hand side (rhs). We have that $rhs(sum(\textbf{SCILS}))$ is equal to the number of inequalities in **SCILS** multiplied by the rhs of each inequality, i.e.:

$$rhs(sum(\textbf{SCILS})) = \frac{\gamma!}{2^{(\frac{\gamma}{2})}(\frac{\gamma}{2}!)} \times \left\lfloor \frac{\beta}{2} \right\rfloor .$$

The coefficient of each variable X_{ij} in sum(**SCILS**) ($coef_{ij}$) is given by the number of inequalities in the set **SCILS** in which X_{ij} appears, i.e.:

$$coef_{ij} = \frac{(\gamma-2)!}{2^{(\frac{(\gamma-2)}{2})}(\frac{(\gamma-2)}{2}!)}$$

Dividing $rhs(sum(\textbf{SCILS}))$ by $coef_{ij}$, we obtain

$$rhs(sum(\textbf{SCILS}))/coef_{ij} = (\gamma - 1) \times \left\lfloor \frac{\beta}{2} \right\rfloor . \tag{37}$$

On the other side, the rhs of **CILS** is:

$$rhs(\textbf{CILS}) = \binom{\beta}{2} = \frac{\beta(\beta-1)}{2}. \tag{38}$$

1. Replacing β with $\gamma - 1$, and $\left\lfloor \frac{\beta}{2} \right\rfloor$ with $\frac{\beta-1}{2}$ (since β is odd), we obtain the result.
2. Consider, for example, $C = \{1, 2, 3, 4, 5, 6\}$ and $\beta = 3$ ($\beta < \gamma - 1$ and odd). In this case, the **CILS** becomes:

$$X_{12} + X_{13} + X_{14} + X_{15} + X_{16} + X_{23} + X_{24} + X_{25} + X_{26} + X_{34} + X_{35} + X_{36} + X_{45} + X_{46} + X_{56} \leq 3.$$

And a particular inequality in **SCILS** is

$$X_{12} + X_{34} + X_{56} \leq 1. \tag{39}$$

The solution $X_{1j} = 1$, for $j = 2, \ldots, 6$, and all other variables equal to zero, satisfies all inequalities in **SCILS**, because only one of the positive variables appears in each inequality in the set. However, the solution does not satisfy **CILS**. On the other side, the solution $X_{12} = X_{34} = X_{56} = 1$, and all other variables equal to zero, satisfies **CILS**, but does not satisfy (39).

Now, consider $C = \{1, 2, 3, 4, 5, 6\}$ and $\beta = 4$ ($\beta < \gamma - 1$ and even). In this case, the **CILS** becomes:

$$\begin{aligned} X_{12} + X_{13} + X_{14} + X_{15} + X_{16} + X_{23} + X_{24} \\ + X_{25} + X_{26} + X_{34} + X_{35} + X_{36} + X_{45} + X_{46} + X_{56} \leq 6. \end{aligned}$$

And a particular inequality in **SCILS** is

$$X_{12} + X_{34} + X_{56} \leq 2. \tag{40}$$

The solution $X_{1j} = 1$, for $j = 2, \ldots, 6$, $X_{2j} = 1$, for $j = 3, \ldots, 6$, and all other variables equal to zero, satisfies all inequalities in **SCILS**, because at most two of the positive variables appear in each inequality in the set. However, the solution does not satisfy **CILS**. On the other side, the solution $X_{12} = X_{34} = X_{56} = 1$, and all other variables equal to zero, satisfies **CILS**, but does not satisfy (40).

□

Theorem 20. Let C be the cover in (28) and consider $\gamma := |C|$ to be odd. Then there is no dominance relation between **SCILS** and **CILS**.

Proof. Consider, for example, $C = \{1, 2, 3, 4, 5\}$ and $\beta = 3$ (β odd). In this case, the **CILS** becomes:

$$X_{12} + X_{13} + X_{14} + X_{15} + X_{23} + X_{24} + X_{25} + X_{34} + X_{35} + X_{45} \leq 3.$$

And a particular inequality in **SCILS** is

$$X_{23} + X_{45} \leq 1. \tag{41}$$

The solution $X_{1j} = 1$, for $j = 1, \ldots, 5$, and all other variables equal to zero, satisfies all inequalities in **SCILS**, because only one of the positive variables appears in each inequality in the set. However, the solution does not satisfy **CILS**. On the other side, the solution $X_{23} = X_{45} = 1$, and all other variables equal to zero, satisfies **CILS**, but does not satisfy (41).

Now, consider $C = \{1, 2, 3, 4, 5\}$ and $\beta = 4$ (β even). In this case, the **CILS** becomes:

$$X_{12} + X_{13} + X_{14} + X_{15} + X_{23} + X_{24} + X_{25} + X_{34} + X_{35} + X_{45} \leq 6.$$

And a particular inequality in **SCILS** is

$$X_{11} + X_{23} + X_{45} \leq 2. \tag{42}$$

The solution $X_{1j} = 1$, for $j = 1, \ldots, 5$, $X_{2j} = 1$, for $j = 2, \ldots, 5$, and all other variables equal to zero, satisfies all inequalities in **SCILS**, because at most two of the positive variables appear in each inequality in the set. However, the solution does not satisfy **CILS**. On the other side, the solution $X_{11} = X_{23} = X_{45} = 1$, and all other variables equal to zero, satisfies **CILS**, but does not satisfy (42). □

Now, we investigate if **SCILS** is just a particular case of **SKILS**, when $\alpha_j \in \{0, 1\}$, for all $j \in N$ in (34).

Theorem 21. In case the modification suggested in Remark 17 is applied, then if $|C|$ is even in (28), **SCILS** becomes just a particular case of **SKILS**. In case $|C|$ is odd, however, the inequalities in **SCILS** are stronger.

Proof. If $|C|$ is even, the result is easily verified. If $|C|$ is odd, the inequalities in **SCILS** become

$$2 \sum_{(i,j) \in C_s} X_{ij} \leq \beta - 1,$$

if β is odd, and

$$2X_{i_0 i_0} + 2 \sum_{(i,j) \in C_s} X_{ij} \leq \beta,$$

if β is even, and the inequalities in **SKILS** become

$$X_{i_0 i_0} + 2 \sum_{(i,j) \in C_s} X_{ij} \leq \beta,$$

for all β. In all cases, C_s is a partition of $C \setminus \{i_0\}$, where $i_0 \in C$.

Either with β even or odd, it becomes clear that **SCILS** is stronger than **SKILS**. □

6. Lower bounds from solutions of the relaxations for QKP $_{\text{lifted}}$

In order to evaluate the quality of the upper bounds obtained with **CRel**, we compare them with lower bounds for the **QKP**, given by feasible solutions constructed by a heuristic procedure.

Let $(\bar{x}, \bar{X})$ be a solution of **CRel**. We initially apply principal component analysis (PCA) [19] to construct an approximation to the solution of the **QKP** and then apply a special rounding procedure to obtain a feasible solution from it. PCA selects the largest eigenvalue and the corresponding eigenvector of $\bar{X}$, denoted by $\bar{\lambda}$ and $\bar{v}$, respectively. Then $\bar{\lambda}\bar{v}\bar{v}^T$ is a rank-one approximation of $\bar{X}$. We set $\bar{x} = \bar{\lambda}^{\frac{1}{2}}\bar{v}$ to be an approximation of the solution x of the **QKP**. Finally, we round $\bar{x}$ to a binary solution that satisfies the knapsack capacity constraint, using the simple approach described in Algorithm 3.

Algorithm 3: Heuristic procedure

Input: the solution $\bar{X}$ from **CRel**, the weight vector w, the capacity c.
Let $\bar{\lambda}$ and $\bar{v}$ be, respectively, the largest eigenvalue and the corresponding eigenvector of $\bar{X}$.
Set $\bar{x} = \bar{\lambda}^{\frac{1}{2}}\bar{v}$.
Round $\bar{x}$ to $\hat{x} \in \{0,1\}^n$.
While $w^T\hat{x} > c$

Set $i = \text{argmin}_{j\in N}\{\bar{x}_j | \bar{x}_j > 0\}$.
Set $\bar{x}_i = 0$, $\hat{x}_i = 0$.

End
Output: a feasible solution $\hat{x}$ of the **QKP**.

7. Numerical Experiments

We summarize our algorithmic framework in Algorithm 4, where at each iteration we update the perturbation Q_p of the parametric relaxation and, at every m iterations, we add to the relaxation, the valid inequalities considered in this paper, namely, **SCI**, defined in (27), **CILS**, defined in (29), and **SCILS**, defined in (31).

The numerical experiments performed had the following main purposes,

- verify the effectiveness of the **IPM** described in Section 3 in decreasing the upper bound while optimizing the perturbation Q_p,
- verify the impact of the valid inequalities, **SCI**, **CILS**, and **SCILS**, when iteratively added to cut the current solution of the relaxation of the **QKP**,
- compute the upper and lower bounds obtained with the proposed algorithmic approach described in Algorithm 4, and compare them, with the optimal solutions of the instances.

We coded Algorithm 4 in MATLAB, version R2015a, and ran the code on a desktop with an AMD FX- 6300 processor, 16GB RAM, running under Ubuntu 16.04. We used the primal-dual **IPM** method implemented in Mosek, version 8, to solve relaxation **CQP**$_{Q_p}$, and, to solve the separation problems **MILP**$_1$ and **MILP**$_2$, we use Gurobi, version 8.

The input data used in the first iteration of the **IPM** described in Algorithm 2 ($k = 0$) are: $B_0 = I$, $\mu^0 = 1$. We depart from a matrix Q_p^0, such that $Q - Q_p^0$ is negative definite. By solving **CQP**$_{Q_p}$, with $Q_p := Q_p^0$, we obtain $x(Q_p^0)$, $X(Q_p^0)$, as its optimal solution, and set $\nabla p^*_{\text{CQP}}(Q_p^0) := X(Q_p^0) - x(Q_p^0)x(Q_p^0)^T$. Finally, the positive definiteness of Z^0 and Λ^0 are assured by setting: $Z^0 := Q_p^0 - Q$ and $\Lambda^0 := \nabla p^*_{\text{CQP}}(Q_p^0) + (2|\lambda_{\min}(\nabla p^*_{\text{CQP}}(Q_p^0)| + .1)I$.

Algorithm 4: Our algorithmic framework

Input: $Q \in \mathbb{S}^n$, $max.n_{cuts}$.
$k := 0$, $B_0 := I$, $\mu^0 := 1$.
Let $\lambda_i(Q), v_i$ be the i^{th} largest eigenvalue of Q and corresponding eigenvector.
$Q_n := \sum_{i=1}^{n}(-|\lambda_i(Q)|-1)v_i v_i'$, $Q_p^0 := Q - Q_n$.
Solve $\mathbf{CQP}_{Q_p}$ (in (8)), with $Q_p := Q_p^0$, and obtain $x(Q_p^0)$, $X(Q_p^0)$.
$\nabla p^*_{\text{CQP}}(Q_p^0) := X(Q_p^0) - x(Q_p^0)x(Q_p^0)^T$.
$Z^0 := Q_p^0 - Q$.
$\Lambda^0 := \nabla p^*_{\text{CQP}}(Q_p^0) + (2|\lambda_{\min}(\nabla p^*_{\text{CQP}}(Q_p^0)|+.1)I$.
While (stopping criterium is violated)

> Run Algorithm 2, where Q_p^{k+1} is obtained and relaxation $\mathbf{CQP}_{Q_p}$, with $Q_p := Q_p^{k+1}$ is solved. Let $(x(Q_p^{k+1}), X(Q_p^{k+1}))$ be its optimal solution.
> $upper.bound^{k+1} := p^*_{\text{CQP}}(Q_p^{k+1})$.
> Run Algorithm 3, where $\hat{x}$ is obtained.
> $lower.bound^{k+1} := \hat{x}^T Q \hat{x}$.
> If k mod $m == 0$
>
> > Solve problem (26) and obtain cuts **SCI** in (27).
> > Add the $\max\{n, max.n_{cuts}\}$ cuts **SCI** with the largest violations at $(x(Q_p^{k+1}), X(Q_p^{k+1}))$, to $\mathbf{CQP}_{Q_p}$.
> > $n_{cuts} := 0$.
> > While ($n_{cuts} < max.n_{cuts}$ & $\mathbf{MILP}_1$ feasible)
> >
> > > Solve $\mathbf{MILP}_1$ and add the **CI** and **CILS** obtained to $\mathbf{CQP}_{Q_p}$.
> > > Add the "no-good" cut (30) to $\mathbf{MILP}_1$.
> > > $n_{cuts} := n_{cuts} + 1$.
> >
> > End
> > $n_{cuts} := 0$.
> > While ($n_{cuts} < max.n_{cuts}$ & $\mathbf{MILP}_2$ feasible)
> >
> > > Solve $\mathbf{MILP}_2$ and add the **CI** and **SCILS** obtained to $\mathbf{CQP}_{Q_p}$.
> > > Add the "no-good" cut (33) to $\mathbf{MILP}_2$.
> > > $n_{cuts} := n_{cuts} + 1$.
> >
> > End
>
> End
> $k := k + 1$.

End
Output: Upper bound $upper.bound^k$, lower bound $lower.bound^k$, and feasible solution $\hat{x}$ to the **QKP**.

Our randomly generated test instances were also used by J. O. Cunha in [8], who provided us with the instances data and with their optimal solutions. Each weight w_j, for $j \in N$, was randomly selected in the interval $[1, 50]$, and the capacity of the knapsack c was randomly selected in $[50, \sum_{j=1}^{n} w_j]$. The procedure used by Cunha to generate the instances was based on previous works [4, 6, 7, 12, 22].

In Tables 1–3, we identify the method applied to compute the lower bound on the first column. On the remaining columns we present, for each method, average results for

- relative optimality gap (OptGap (%):= ((upper bound - opt)/opt) $\times$ 100, where opt is the optimal solution value),
- computational time to compute the bound (Time (sec)),
- relative duality gap (DuGap) := (upper bound - lower bound)/(|upper bound|) + (|lower bound|), where the lower bound is computed as described in Sect. 6,
- number of iterations (Iter),
- the number of cuts added to the relaxation (Cuts),
- computational time to obtain cuts **CILS** and **SCILS** ($\text{Time}_{\text{MILP}}$ (sec)).

In Tables 1–2 we present statistics for 10 instances with $n = 10$. Results in Table 1 have the purpose of showing the impact of the cuts presented. For that, we first add them iteratively to the following linear relaxation

$$(43) \qquad (\tilde{\mathbf{LPR}}) \quad \begin{array}{lll} \max & \text{trace}(QX) & \\ \text{s.t.} & \sum_{j=1}^{n} w_j x_j \le c, & \\ & 0 \le X_{ij} \le 1, & \forall i, j \in N \\ & 0 \le x_i \le 1, & \forall i \in N \\ & X \in \mathbb{S}^n. & \end{array}$$

In the first row of Table 1, the results correspond to the solution of the linear relaxation $\tilde{\mathbf{LPR}}$ with no cuts. In $\mathbf{SCI}_1$, we add only the most violated cut from the n cuts in **SCI** to $\tilde{\mathbf{LPR}}$ at each iteration, and in the **SCI** we add all n cuts. In **CILS** and **SCILS**, we solve **MILP** problems to find the most violated cut of each type. The last row of the table (All) corresponds to results obtained when we add all n cuts in **SCI**, and one cut of each type, **CILS** and **SCILS**. In these initial tests, we run up to 50 iterations, and in most cases, stop the algorithm when no more cuts are found to be added to the relaxation. We note that we use a time limit of 3 seconds to solve the separation problems. However, when $n = 10$, this time is sufficient to solve all problems to optimality.

Figure 1 depicts the optimality gaps from Table 1. There is a trade-off between the quality of the cuts and the computational time needed to find them. Considering a unique cut of each type, we note that **SCILS** is the strongest cut

Method	**OptGap** (%)	**Time** (sec)	**DuGap**	**Iter**	**Cuts**	$\textbf{Time}_{\textbf{MILP}}$ (sec)
$\tilde{\text{LPR}}$	38.082	0.35	0.620	1.0		
SCI_1	36.703	32.38	0.343	1.1	28.4	
SCI	10.036	39.98	0.058	3.0	364.1	
CILS	19.719	9.00	0.293	2.7	82.2	6.91
SCILS	9.121	266.81	0.250	50.0	794.3	198.12
ALL	3.315	315.82	0.016	28.3	646.6	264.91

Table 1: Impact of the cuts added to $\tilde{\text{LPR}}$ (10 instances, $n = 10$).

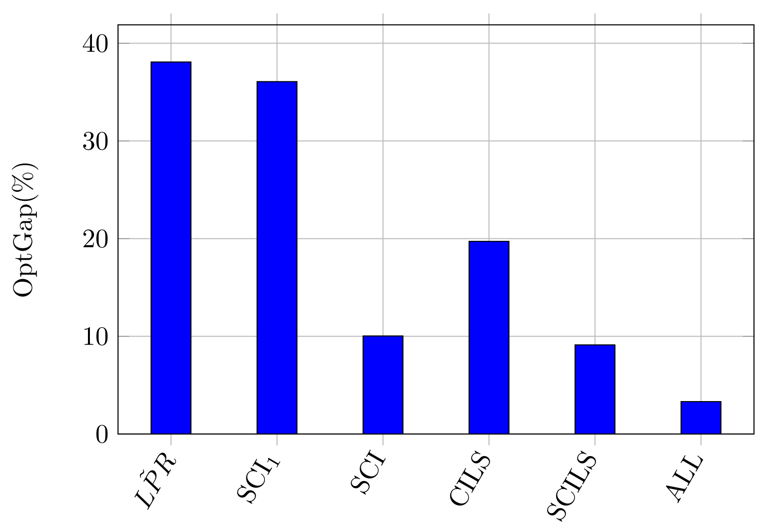

Figure 1: Average optimality gaps from Table 1

(OptGap = 9.121%), but the computational time to obtain it, if compared to **CILS** and **SCI**, is bigger. Nevertheless, a decrease in the times could be achieved with a heuristic solution for the separation problems. We point out that using all cuts together we find a better upper bound than using each type of cut in separate (OptGap = 3.315%).

We now present results from our main tests, considering the parametric quadratic relaxation, the **IPM** and the cuts. To improve the results, we also consider in our initial relaxation the valid inequalities obtained by multiplying the capacity constraint by each nonnegative variable x_i, and also valid inequalities derived from the fact that $x_i \in \{0, 1\}$. We then start the algorithms solving the following relaxation.

$$
\text{(44)} \qquad (\textbf{QPR}) \qquad
\begin{array}{lll}
\max & x^T(Q - Q_p^0)x + \operatorname{trace}(Q_p^0 X) & \\
\text{s.t.} & \sum_{j=1}^n w_j x_j \le c, & \\
& \sum_{j=1}^n w_j X_{ij} \le c X_{ii}, & \forall i \in N \\
& X_{ii} = x_i, & \forall i \in N \\
& 0 \le X_{ij} \le 1, & \forall i, j \in N \\
& 0 \le x_i \le 1, & \forall i \in N \\
& X \in \mathbb{S}^n. &
\end{array}
$$

In order to evaluate the influence of the initial decomposition of Q on the behavior of the **IPM**, we considered two initial decompositions. In both cases, we compute the eigendecomposition of Q, getting $Q = \sum_{i=1}^n \lambda_i v_i v_i'$.

- For the first decomposition, we set $Q_n := \sum_{i=1}^n (-|\lambda_i| - 1) v_i v_i'$, and $Q_p^0 := Q - Q_n/2$. We refer to this initial matrix Q_p^0 as Q_p^a in the following tables and figures.
- For the second, we set $Q_n := \sum_{i=1}^n (\min\{\lambda_i, -10^{-6}\}) v_i v_i'$, and $Q_p^0 := Q - Q_n/2$. We refer to this initial matrix as Q_p^b.

We implemented the **IPM** updating the Hessian matrix B using the BFGS procedure described in Section 3 and also considering the simpler approximation $B = I$ in all iterations.

In Table 2 we show the average results obtained for these two procedures, and for the two initial decompositions of Q described above. In the first two rows of the table, the results are obtained from the solution of the initial quadratic relaxation, with the initial decomposition of Q and no cuts. In the next four rows of the table, the results are obtained with the application of the **IPM**, with no cuts added to the relaxation. In the last four rows, the results are obtained with the inclusion of cuts in the relaxation. The cuts are added at every $m = 10$ iterations of the **IPM** and the numbers of cuts added at each iteration are n **SCI**, 5 **CILS** and 5 **SCILS**. Note that when solving each **MILP** problem, besides the cut **CILS** or **SCILS**, we also obtain a cover inequality **CI**. We check if this **CI** was already added to the relaxation, and if not, we add it as well.

For these tests we set the maximum number of iterations to 150, and the maximum computational time to 900 seconds. We also stop the algorithms if DuGap is sufficiently small. Table 2 shows that the best bounds are obtained

Method	**OptGap** (%)	**Time** (sec)	**DuGap**	**Iter**	**Cuts**	**Time$_{\mathbf{MILP}}$** (sec)
Q_p^a,QPR	21.640	0.79	0.138	1.0		
Q_p^b,QPR	12.276	0.83	0.076	1.0		
Q_p^a,I	7.242	26.80	0.042	150.0		
Q_p^b,I	7.633	25.81	0.055	150.0		
Q_p^a,BFGS	7.091	27.41	0.041	150.0		
Q_p^b,BFGS	7.094	25.94	0.041	150.0		
Q_p^a,I,Cuts	0.863	87.09	0.009	144.5	104	42.13
Q_p^b,I,Cuts	1.516	77.60	0.012	132.8	97.6	38.37
Q_p^a,BFGS,Cuts	0.639	46.13	0.008	77.7	275.4	23.57
Q_p^b,BFGS,Cuts	0.640	56.32	0.008	98.7	80.7	29.24

Table 2: Average results for 10 instances ($n = 10$).

when we use the **IPM** with the BFGS update of the Hessian, and adding the cuts. Concerning the starting point, Q_p^a leads better bounds in general, but the computational time is slightly bigger than for Q_p^b. Figure 2 depicts the optimality gaps from Table 2.

In Table 3 we show how the results evolve when n increases. For these final tests, we set the maximum number of iterations to 500, the maximum computational time to 2700 seconds, and also stop the algorithms if DuGap is sufficiently small. We note again that the **IPM** , with or without cuts, decreases

Method	**OptGap** (%)	**Time** (sec)	**DuGap**	**Iter**	**Cuts**	**Time$_{\mathbf{MILP}}$** (sec)
Q_p^a,QPR	21.136	0.87	0.113	1.0		
Q_p^b,QPR	9.460	0.86	0.056	1.0		
Q_p^a,BFGS	1.345	2732.70	0.015	424.0		
Q_p^b,BFGS	1.345	2713.26	0.015	430.0		
Q_p^a,BFGS,Cuts	0.078	2216.81	0.001	168.6	241.2	858.51
Q_p^b,BFGS,Cuts	0.076	1882.68	0.001	145.0	199.4	731.55

Table 3: Average results for 5 instances ($n = 50$).

the initial upper bound given by the solution of the quadratic relaxation. The influence of the initial decomposition of Q on the bounds obtained by the **IPM** is not relevant, but the convergence is still faster with the initial decomposition Q_p^b. It is interesting to note that, although the solution of the **MILP** problems is computationally expensive, the time spent solving them is compensated by the faster convergence of the algorithm and to better bounds.

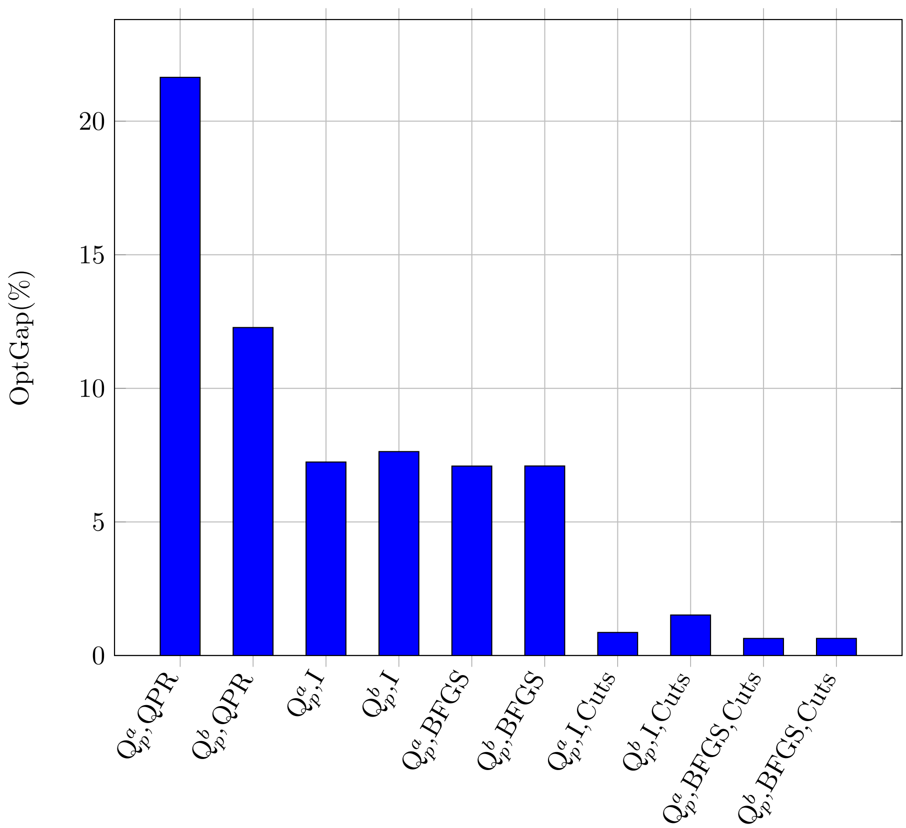

Figure 2: Average optimality gaps from Table 2

8. Conclusion

In this paper we present a cutting plane algorithm (CPA) to iteratively improve the upper bound for the quadratic knapsack problem, **QKP** . The initial relaxation for the problem is given by a parametric convex quadratic problem, where the Hessian Q of the objective function of the **QKP** is perturbed by a matrix parameter Q_p, such that $Q - Q_p \preceq 0$. Seeking for the best possible bound, the concave term $x^T(Q - Q_p)x$, is then kept in the objective function of the relaxation and the remaining part, given by $x^T Q_p x$ is linearized through the standard approach that lifts the problem to space of symmetric matrices defined by $X := xx^T$.

We present an interior point algorithm, **IPM** , which update the perturbation Q_p at each iteration of the CPA aiming at reducing the upper bound given by the relaxation. We also present new classes of cuts that are added at each iteration of the CPA, defined on the lifted variable X, and derived from cover inequalities and the binary constraints.

We show that both the **IPM** and the cuts generated are effective in improving the upper bound for the **QKP** and note that these procedures could be applied to more general binary indefinite quadratic problems as well. The separation problems described to generate the cuts could also be solved heuristically, in order to accelerate the process.

Finally, we show that if the positive semidefinite constraint $X - xx^T \succeq 0$ was introduced in the relaxation of the **QKP** , or any other indefinite quadratic problem (maximizing the objective function), then the decomposition of objective function, that leads to a convex quadratic relaxation, where a perturbed concave part of the objective is kept, and the remaining part is linearized, is not effective. In this case the best bound is always attained when the whole objective function is linearized, i.e., when the perturbation Q_p is equal to Q. This observation also relates to the well known DC (difference of convex) decomposition of indefinite quadratics that have been used in the literature to generate bounds for indefinite quadratic problems. Once more, in case the positive semidefinite constraint is added to the relaxation, the DC decomposition is not effective anymore, and the alternative linear **SDP** relaxation leads to the best possible bound.

Acknowledgements

We thank J. O. Cunha for providing us with the test instances used in our experiments. M. Fampa was partially supported by CNPq (National Council for Scientific and Technological Development) Grant 303898/2016-0, D. C. Lubke was supported by Research Grants from CAPES (Brazilian Federal Agency for Support and Evaluation of Graduate Education) Grant 88881.131629/2016-01, CNPq Grant 142143/2015-4 and partially supported by PSR - Power Systems Research, Rio de Janeiro, Brazil.

References

References

[1] A. Atamtürk. Cover and pack inequalities for (mixed) integer programming. *Annals of Operations Research*, 139(1):21–38, Oct 2005.

[2] E. Balas. Facets of the knapsack polytope. *Mathematical Programming*, 8(1):146–164, Dec 1975.

[3] E. Balas and E. Zemel. Facets of the knapsack polytope from minimal covers. *SIAM Journal on Applied Mathematics*, 34(1):119–148, 1978.

[4] A. Billionnet and F. Calmels. Linear programming for the 0-1 quadratic knapsack problem. *European Journal of Operational Research*, 92(2):310–325, 1996.

[5] A. Billionnet, A. Faye, and É. Soutif. A new upper bound for the 0-1 quadratic knapsack problem. *European Journal of Operational Research*, 112(3):664–672, 1999.

[6] A. Caprara, D. Pisinger, and P. Toth. Exact solution of the quadratic knapsack problem. *INFORMS Journal on Computing*, 11(2):125–137, 1999.

[7] P. Chaillou, P. Hansen, and Y. Mahieu. Best network flow bounds for the quadratic knapsack problem. In Bruno Simeone, editor, *Combinatorial Optimization: Lectures given at the 3rd Session of the Centro Internazionale Matematico Estivo (C.I.M.E.) held at Como, Italy, August 25–September 2, 1986*, pages 225–235. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989.

[8] J.O. Cunha, L. Simonetti, and A. Lucena. Lagrangian heuristics for the quadratic knapsack problem. *Computational Optimization and Applications*, 63(1):97–120, Jan 2016.

[9] J.M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

[10] M. Fampa, J. Lee, and W. Melo. On global optimization with indefinite quadratics. *EURO Journal on Computational Optimization*, 5(3):309–337, 09 2017.

[11] A.V. Fiacco. *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, volume 165 of *Mathematics in Science and Engineering*. Academic Press, 1983.

[12] G. Gallo, P. L. Hammer, and B. Simeone. Quadratic knapsack problems. In M. W. Padberg, editor, *Combinatorial Optimization*, pages 132–149. Springer Berlin Heidelberg, Berlin, Heidelberg, 1980.

[13] Z. Gu, G.L. Nemhauser, and M.W.P. Savelsbergh. Lifted cover inequalities for 0-1 integer programs: Computation. *INFORMS Journal on Computing*, 10(4):427–437, 1998.

[14] P.L. Hammer, E.L. Johnson, and U. N. Peled. Facet of regular 0–1 polytopes. *Mathematical Programming*, 8(1):179–206, Dec 1975.

[15] C. Helmberg, F. Rendl, and R. Weismantel. Quadratic knapsack relaxations using cutting planes and semidefinite programming. In *Integer Programming and Combinatorial Optimization (Vancouver, BC, 1996)*, volume 1084 of *Lecture Notes in Comput. Sci.*, pages 190–203. Springer, Berlin, 1996.

[16] C. Helmberg, F. Rendl, and R. Weismantel. A semidefinite programming approach to the quadratic knapsack problem. *J. Comb. Optim.*, 4(2):197–215, 2000.

[17] W.W. Hogan. The continuity of the perturbation function of a convex program. *Operat. Research*, 21:351–352, 1973.

[18] R. Horst and N.V.l Thoai. Dc programming: Overview. *Journal of Optimization Theory and Applications*, 103:1–43, 1999.

[19] I. Jolliffe. *Principal component analysis.* 2nd ed. Springer, Berlin, 2010.

[20] A.S. Lewis and M.L. Overton. Nonsmooth optimization via quasi-newton methods. *Mathematical Programming*, 141(1):135–163, Oct 2013.

[21] G.P. McCormick. Computability of global solutions to factorable nonconvex programs: part i convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

[22] P. Michelon and L. Veillieux. Lagrangean methods for the 0-1 quadratic knapsack problem. *European Journal of Operational Research*, 92(2):326 – 341, 1996.

[23] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization.* John Wiley & Sons, New York, NY, 1988.

[24] D. Pisinger. The quadratic knapsack problem, a survey. *Discrete Applied Mathematics*, 155(5):623 – 648, 2007.

[25] B.A. Turlach and S.J. Wright. Quadratic programming. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(2):153–159.

[26] L.A. Wolsey. Faces for a linear inequality in 0–1 variables. *Mathematical Programming*, 8(1):165–178, Dec 1975.

[27] L.A. Wolsey. *Integer programming.* Wiley-Interscience, New York, NY, USA, 1998.

[28] E. Zemel. Easily computable facets of the knapsack polytope. *Mathematics of Operations Research*, 14(4):760–764, 1989.