ESCAPE 26, June 12-15, 2016, Portoroz, Slovenia

Testing a Non-Diagonal Convex Reformulation Technique for 0-1 Quadratic Programs

Otto Nissfolk, Ray Pörn and Tapio Westerlund

PROCESS DESIGN AND SYSTEMS ENGINEERING LABORATORY
ÅBO AKADEMI UNIVERSITY, TURKU, FINLAND

ESCAPE 26, JUNE 13th 2016

OSE OPTIMIZATION AND SYSTEMS ENGINEERING

Åbo Akademi University

Structure of the presentation

Work by PhD student Otto Nissfolk

(supervisors PhD Ray Pörn and Professor Tapio Westerlund)

- Background
- Convexification of (0-1) QPs and their relaxation, with some examples
- Quadratic and semidefinite programming
- The Quadratic Convex Reformulation (QCR) Method
- The Nondiagonal QCR (NDQCR)
- Some numerical experiments

Background, (0-1) Quadratic Program

A standard 0-1 QP has the form:

$$\begin{aligned} \min \quad & x^T Q x + q^T x \\ s.t. \quad & Ax = a \\ & Bx \leq b \\ & x \in \{0,1\}^n \end{aligned}$$

Q, A, B are matrices and q, a, b are vectors of appropriate dimensions.
x is a vector with binary decision variables

These problems are generally nonconvex. (Integer relaxed) convex problem if Q is PSD.

Applications include, for example:

- Max-cut of a graph (unconstrained)
- Knapsack problems (inequality constrained)
- Graph bipartitioning
- Task allocation
- Quadratic assignment problems
- Coulomb glass problems
- Gray-scale pattern problems, (for example the taixxc instances in QAPLIB)

Background, convexity

The following are equivalent ($Q = Q^T$):

- The quadratic function $f(x) = x^T Q x$ is convex on R^n.
- The matrix Q is positive semidefinite (psd, $Q \succcurlyeq 0$).
- All eigenvalues of Q are non-negative ($\lambda_i \geq 0$).

A sufficient condition for convexity: A **diagonally dominant** matris is psd.

Definition: A matrix Q is diagonally dominant if

$$Q_{ii} \geq \sum_{i \neq j} |Q_{ij}| \quad \forall i$$

Also given by Gerschgorin's circle theorem.

Convexification of 0-1 QPs

Basic approach: If Q is indefinite, then sufficient large quadratic terms can be added to the diagonal and the same amount subtracted from the linear terms.

(Such a convexification is an exact (0-1) reformulation of the original (0-1) QP problem, but a convex relaxation of the (integer relaxed) QP problem.
The tightness of the convex relaxation affect the solution efficiency, when solving the (0-1) QP-problem.)

Example 1

$$f(x) = x^T \begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix} x = x_1^2 + 6x_1x_2 + 2x_2^2$$

Recalling that: $x_i \in \{0,1\} \;\Leftrightarrow x_i^2 = x_i$

$$f(x) = x^T \begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix} x = x^T \begin{bmatrix} 3 & 3 \\ 3 & 5 \end{bmatrix} x - \begin{bmatrix} 2 \\ 3 \end{bmatrix}^T x = 3x_1^2 + 6x_1x_2 + 5x_2^2 - 2x_1 - 3x_2$$

Indefinite

Positive semidefinite

Convexification of 0-1 QPs

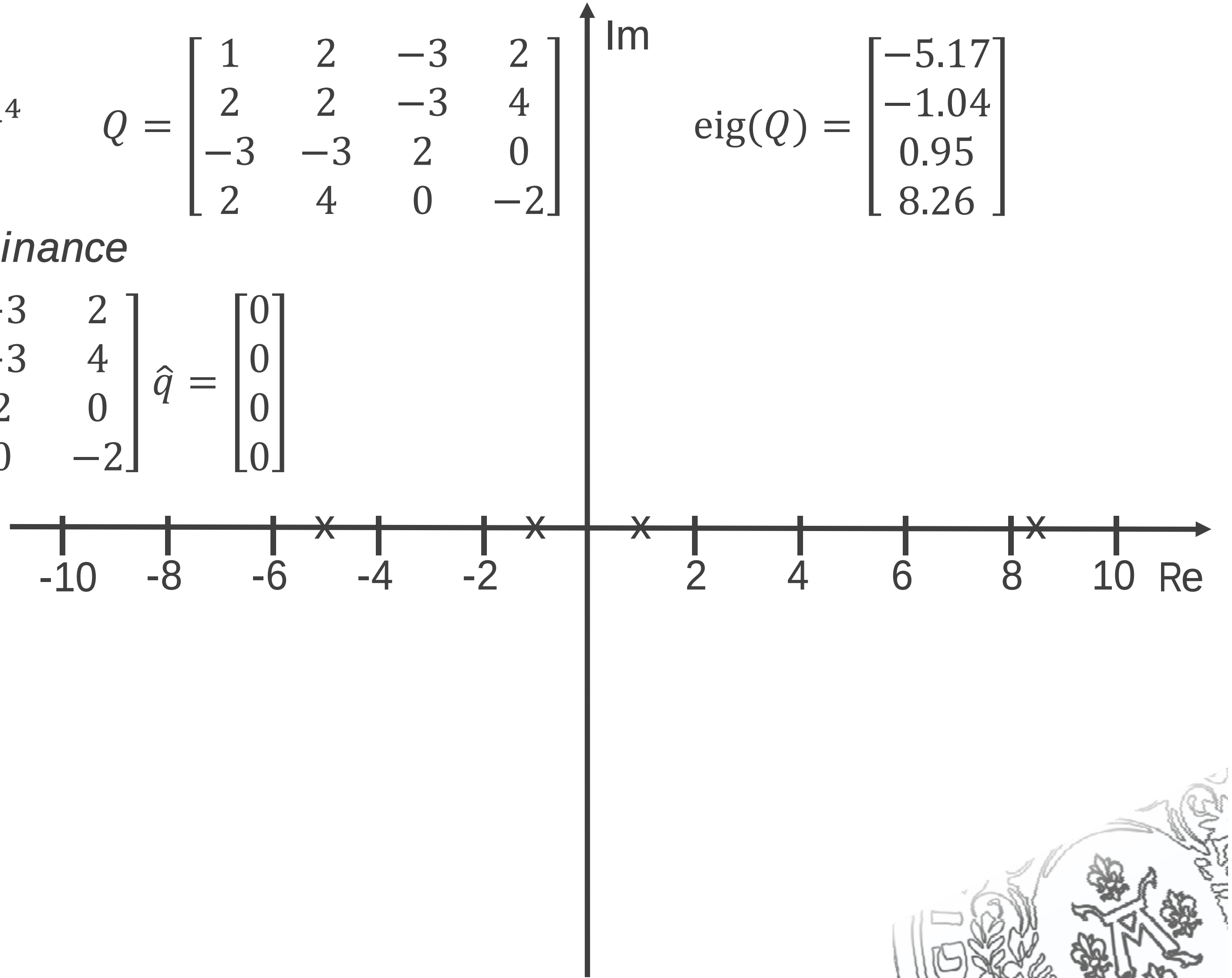
Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix}$$

$$\operatorname{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Convexification of 0-1 QPs

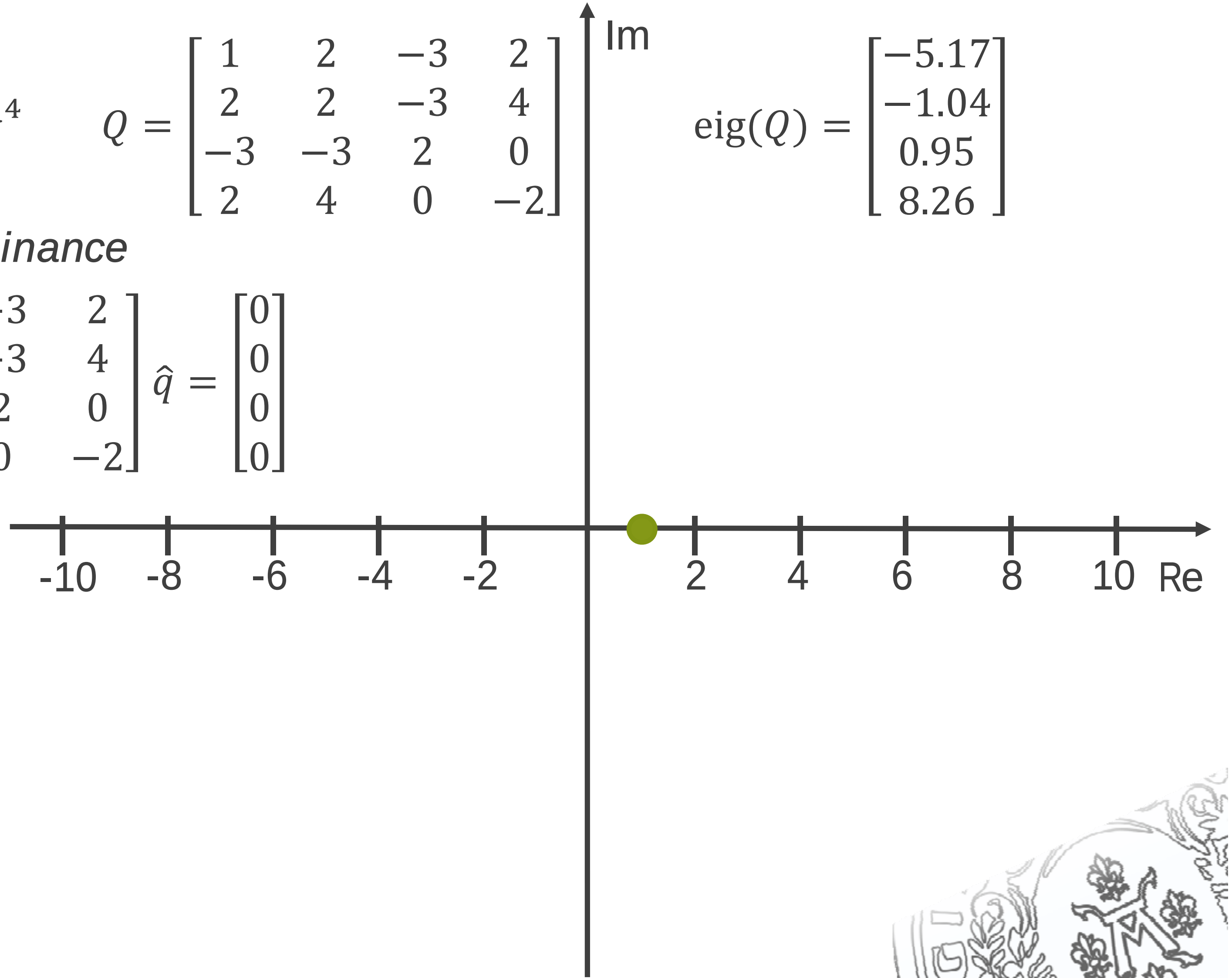
Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix}$$

$$\operatorname{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



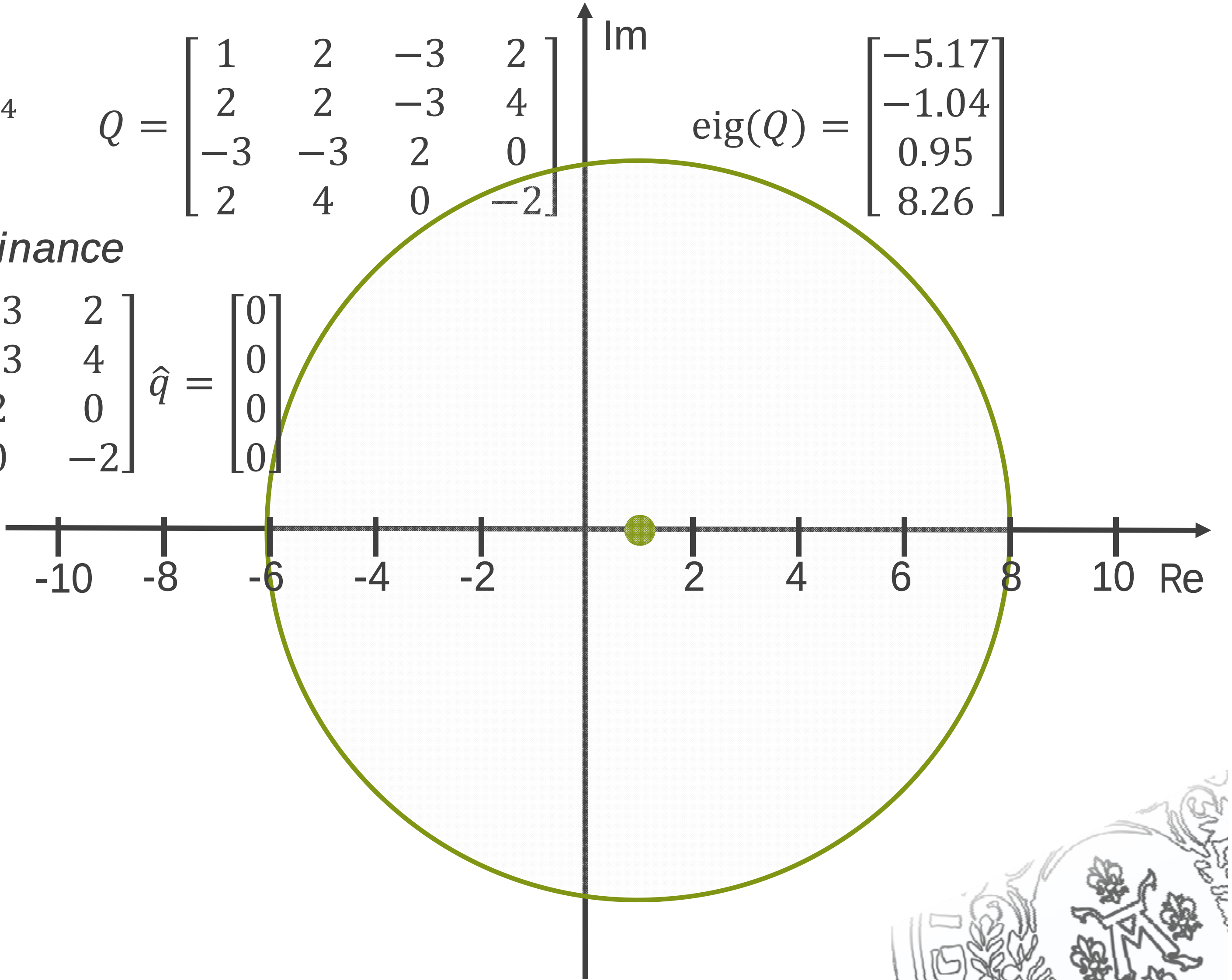
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Convexification of 0-1 QPs

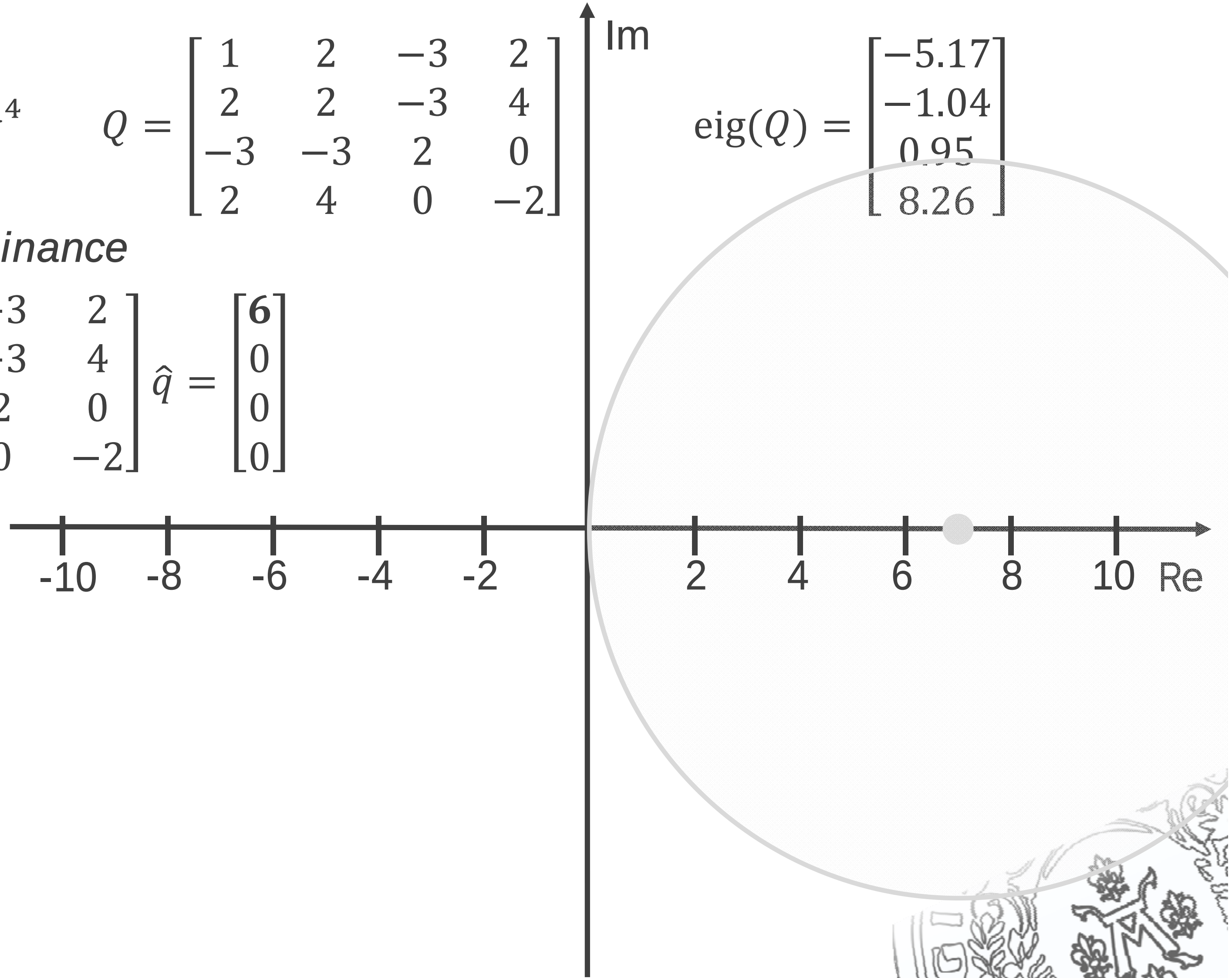
Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix}$$

$$\operatorname{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} \mathbf{6} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



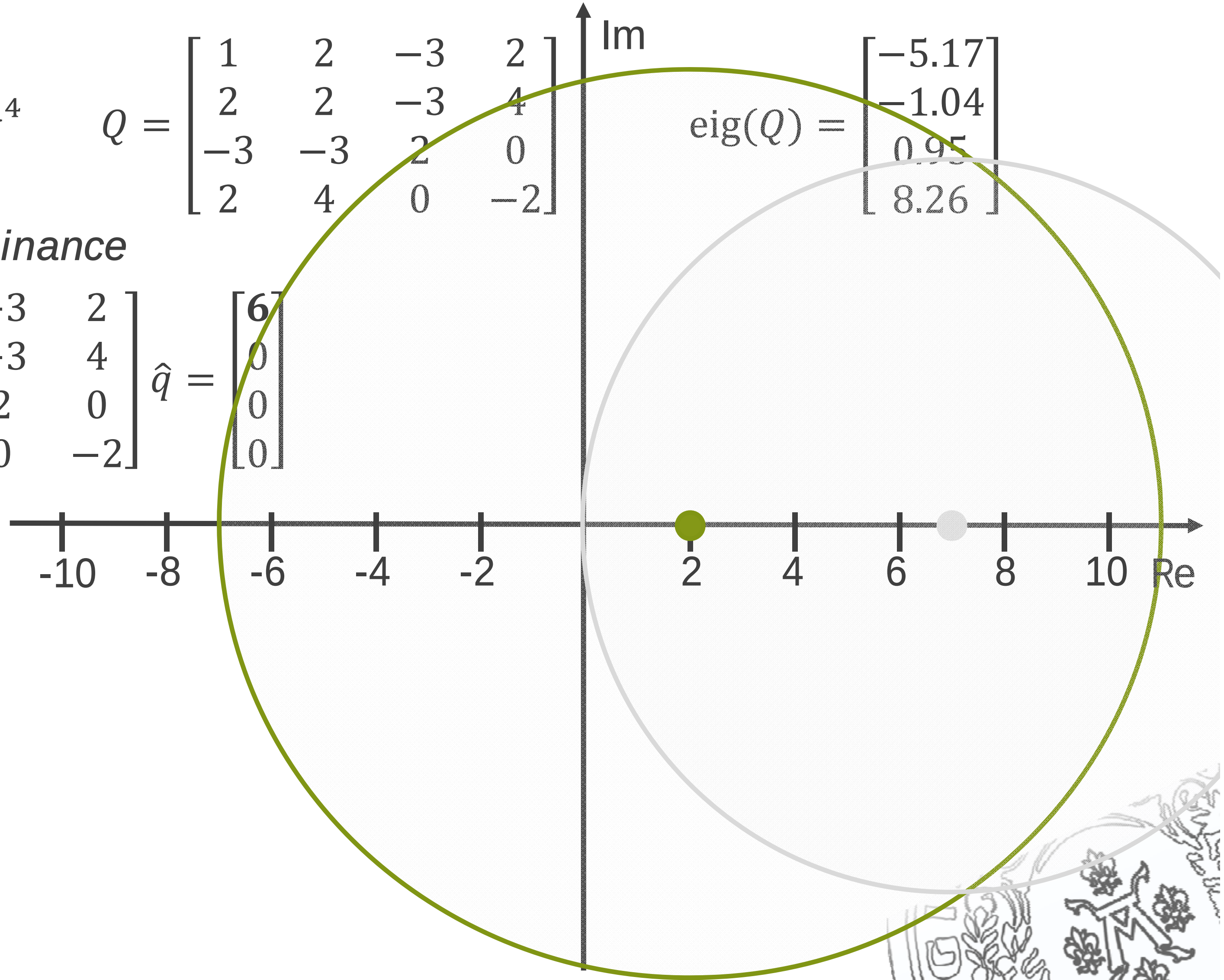
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \operatorname{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} \mathbf{6} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Convexification of 0-1 QPs

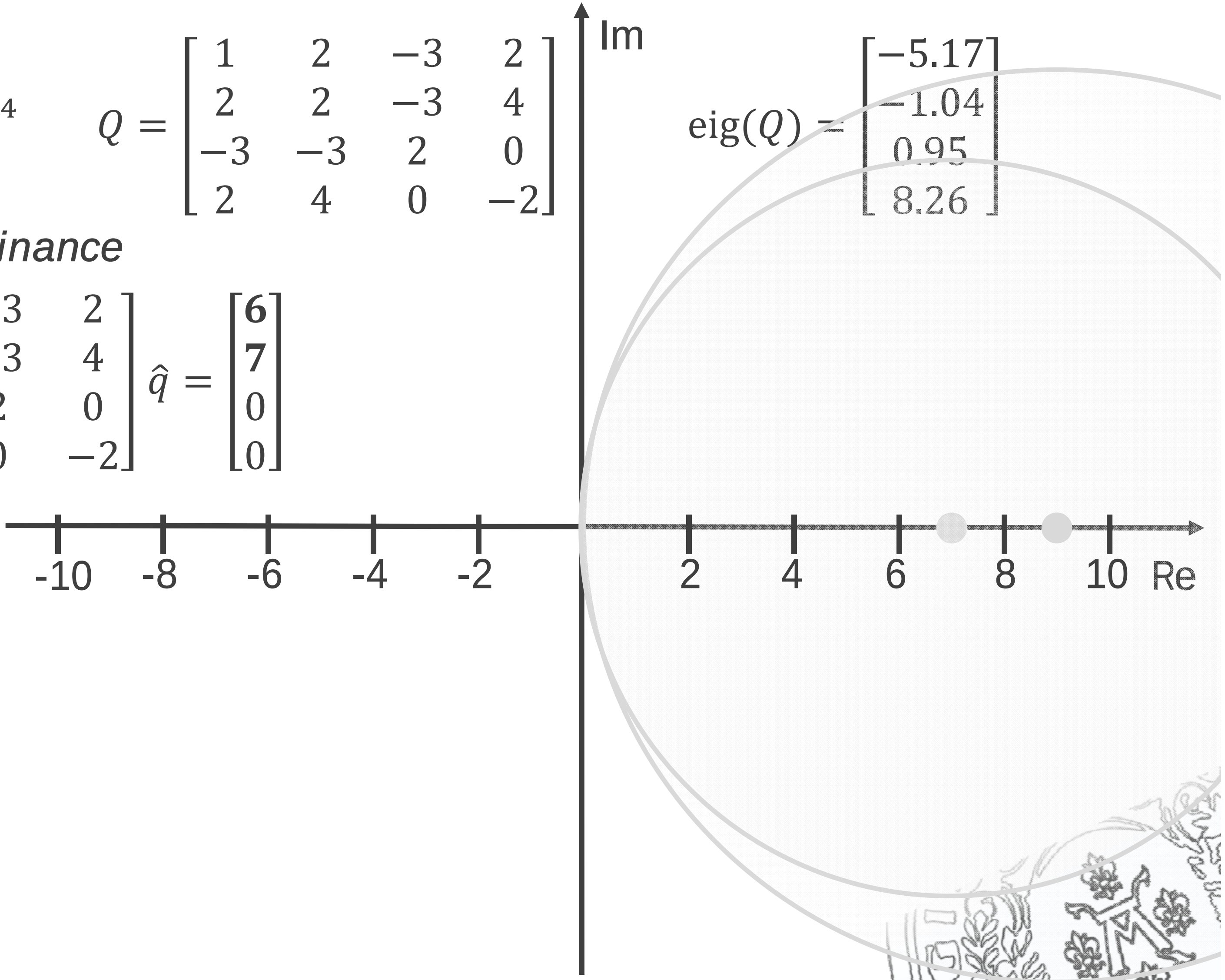
Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix}$$

$$\mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ 0 \\ 0 \end{bmatrix}$$



Convexification of 0-1 QPs

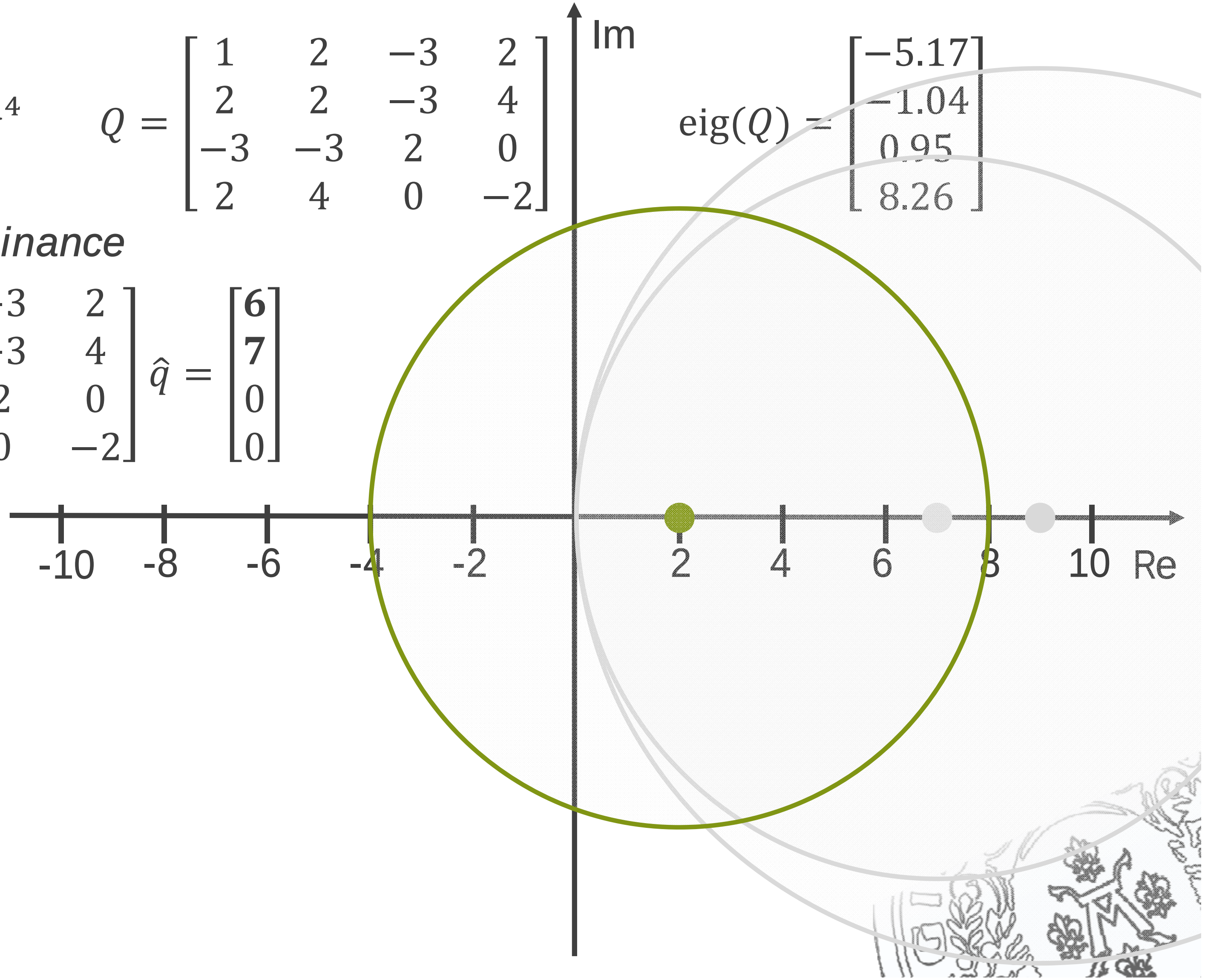
Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix}$$

$$\mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ 0 \\ 0 \end{bmatrix}$$



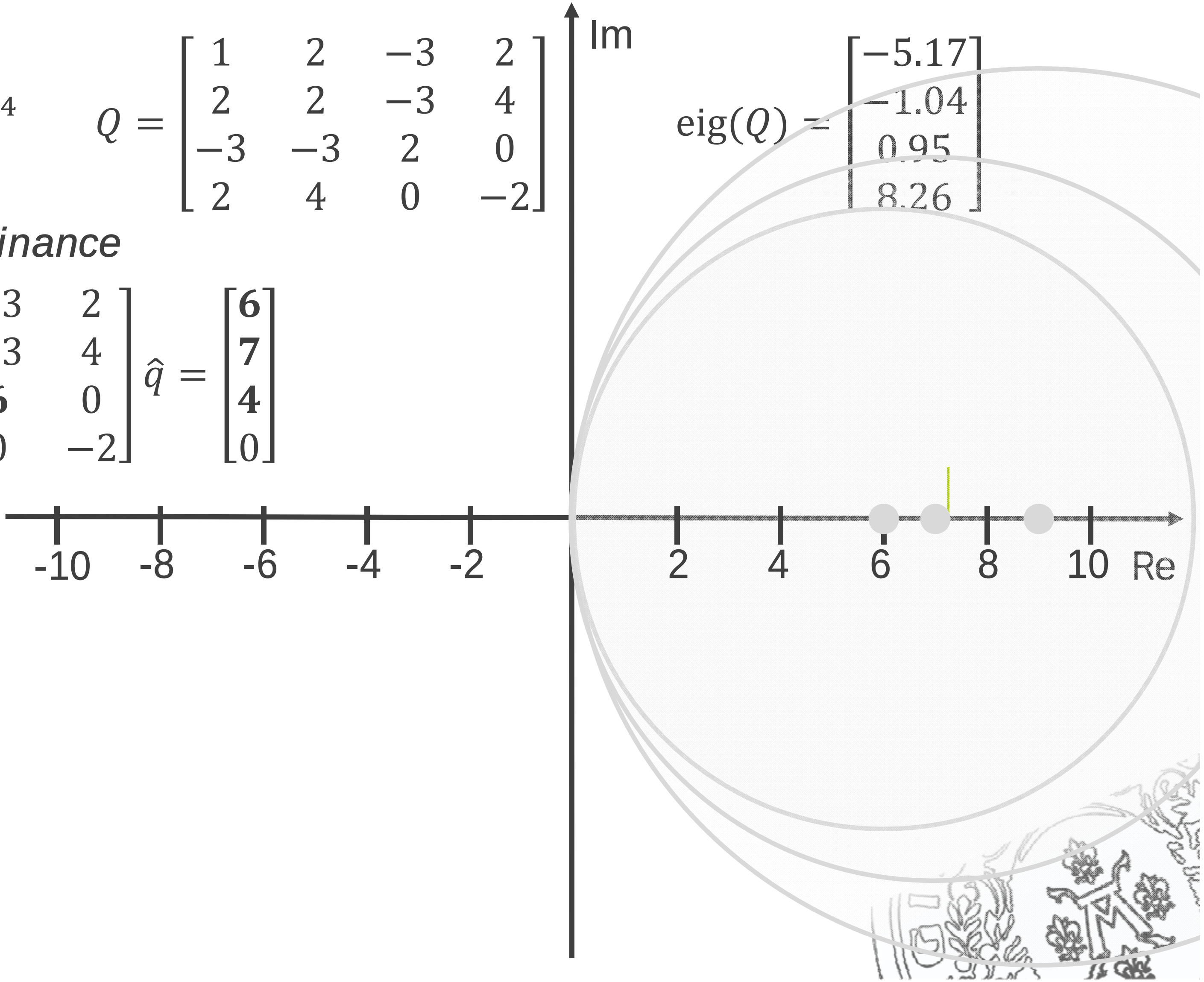
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & \mathbf{6} & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ \mathbf{4} \\ 0 \end{bmatrix}$$



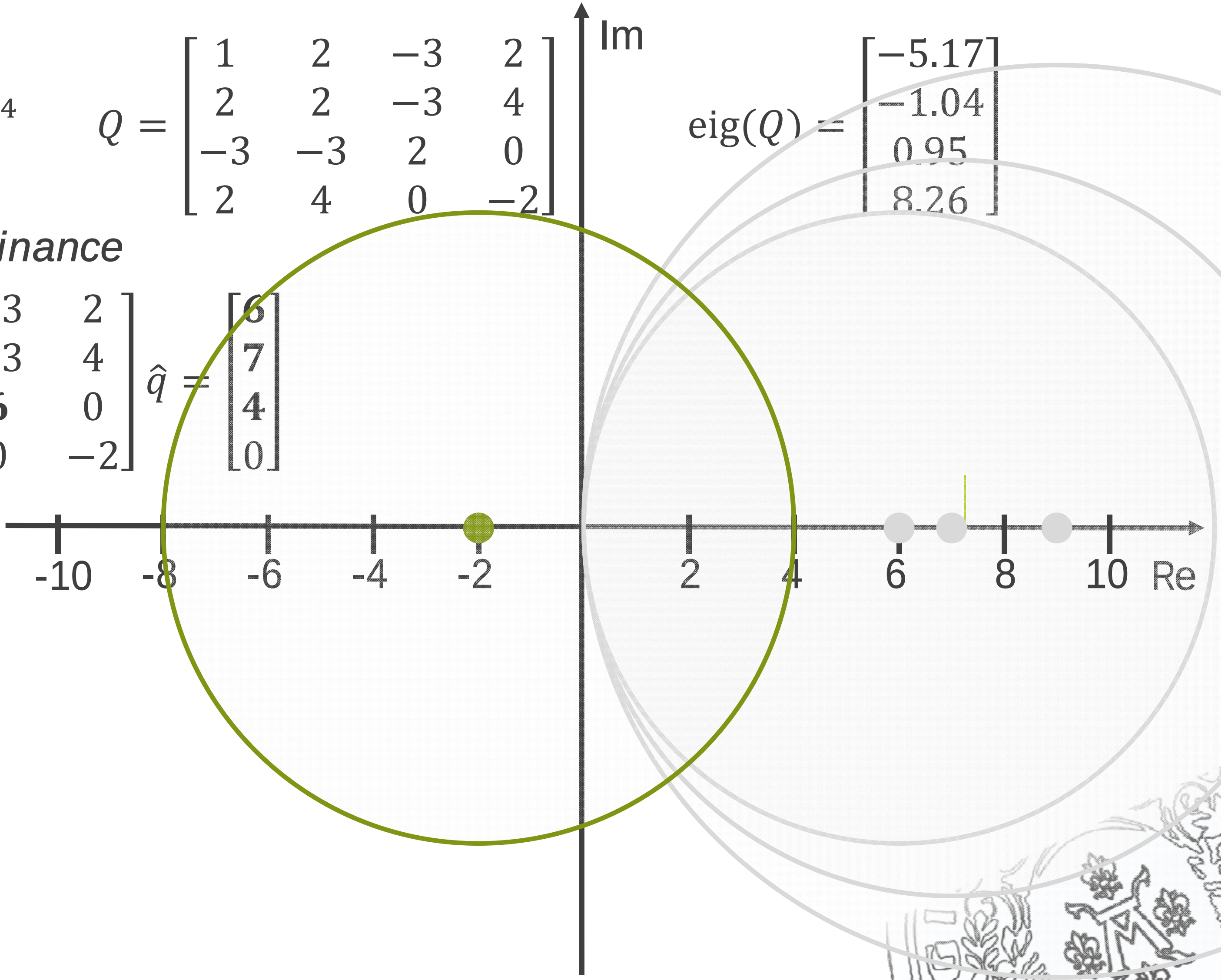
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & \mathbf{6} & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ \mathbf{4} \\ 0 \end{bmatrix}$$



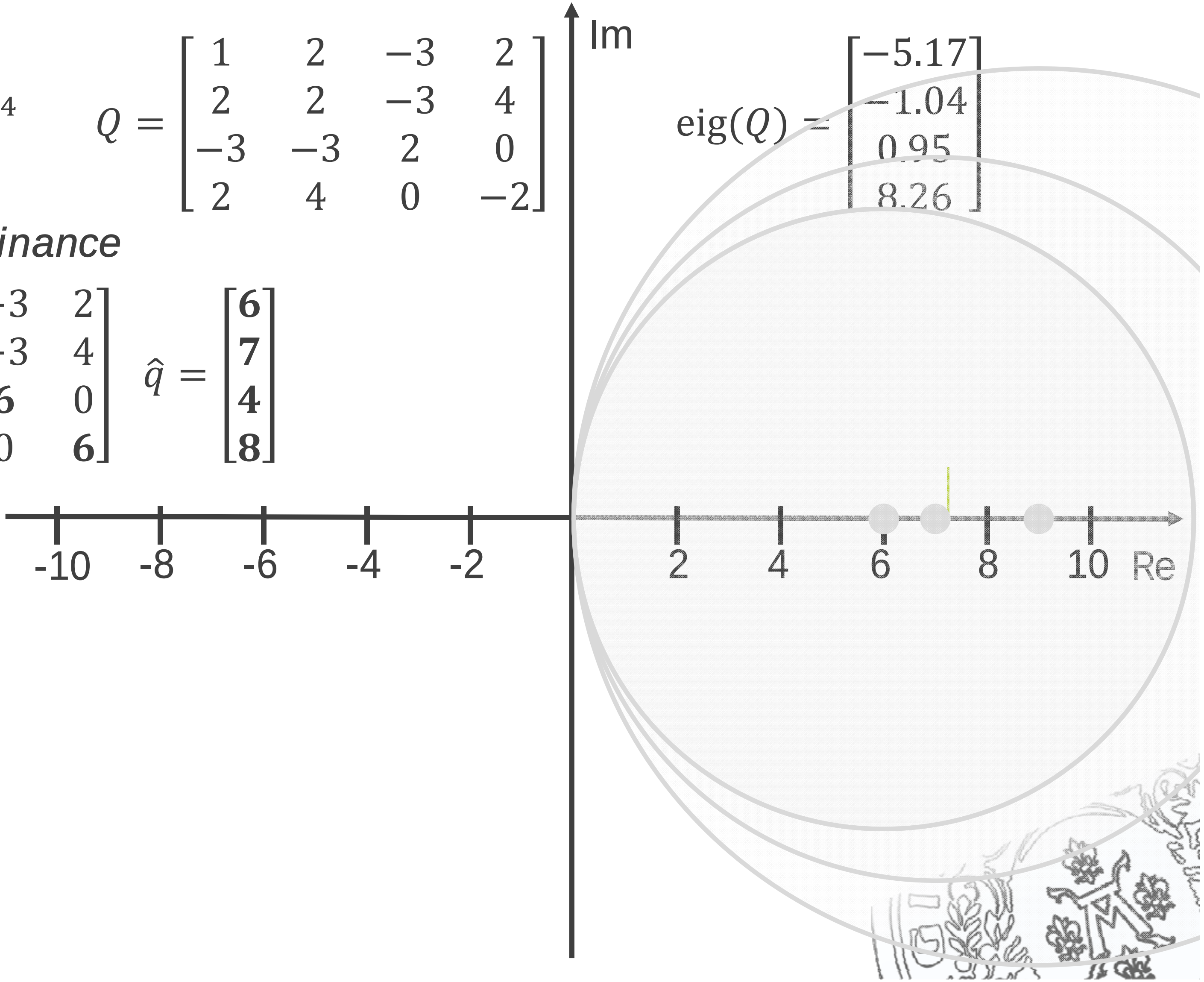
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & \mathbf{6} & 0 \\ 2 & 4 & 0 & \mathbf{6} \end{bmatrix} \qquad \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ \mathbf{4} \\ \mathbf{8} \end{bmatrix}$$



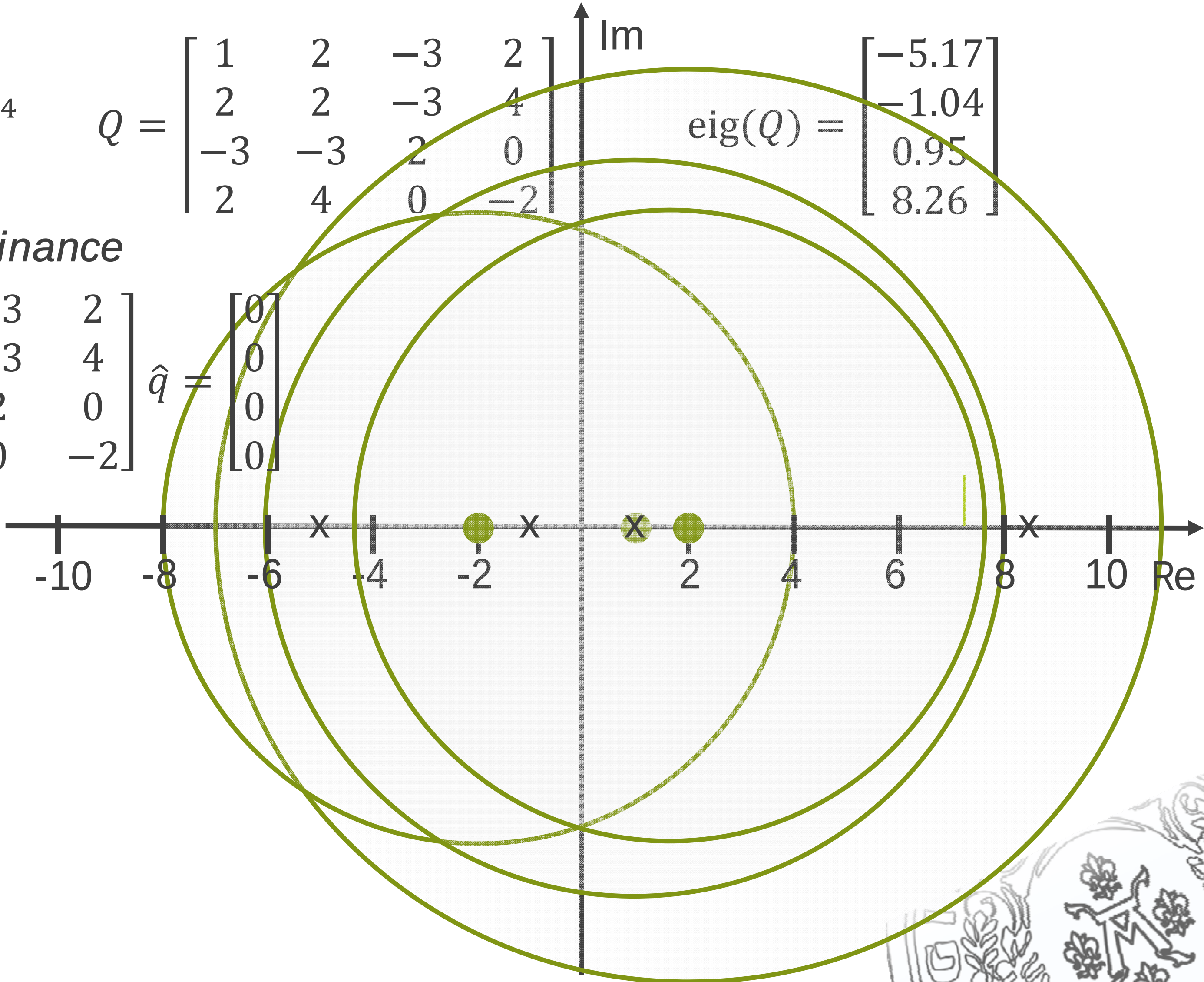
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



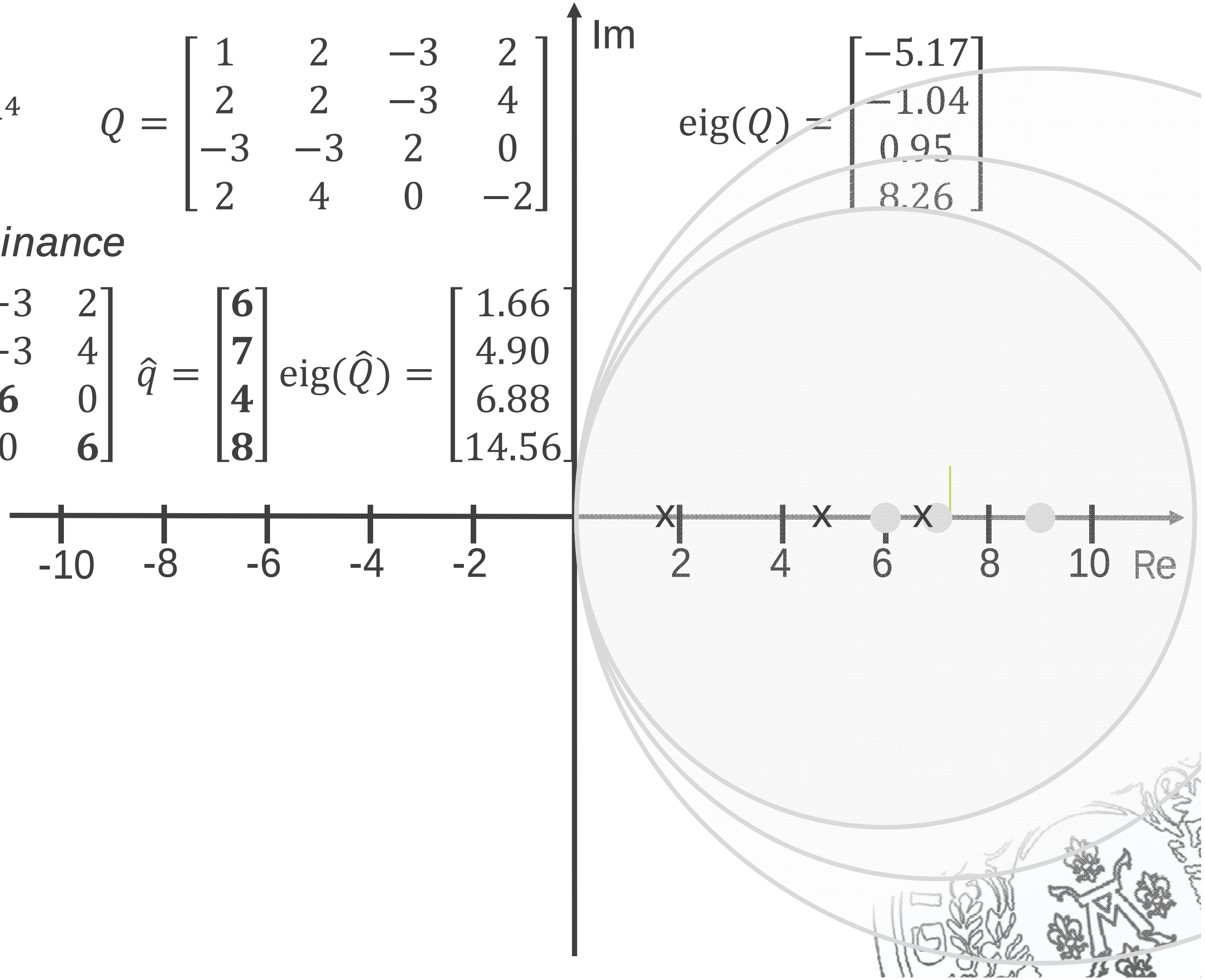
Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} \mathbf{7} & 2 & -3 & 2 \\ 2 & \mathbf{9} & -3 & 4 \\ -3 & -3 & \mathbf{6} & 0 \\ 2 & 4 & 0 & \mathbf{6} \end{bmatrix} \quad \hat{q} = \begin{bmatrix} \mathbf{6} \\ \mathbf{7} \\ \mathbf{4} \\ \mathbf{8} \end{bmatrix} \quad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 1.66 \\ 4.90 \\ 6.88 \\ 14.56 \end{bmatrix}$$



Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 7 & 2 & -3 & 2 \\ 2 & 9 & -3 & 4 \\ -3 & -3 & 6 & 0 \\ 2 & 4 & 0 & 6 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} 6 \\ 7 \\ 4 \\ 8 \end{bmatrix} \qquad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 1.66 \\ 4.90 \\ 6.88 \\ 14.56 \end{bmatrix}$$

$$\begin{array}{ll} \min & x^T \hat{Q} x - \hat{q}^T x \\ s.t. & x \in [0,1]^4 \end{array}$$

$\mathbf{Optimal\ value} = \mathbf{-5.93}$
(continuous relaxation)

Convexification of 0-1 QPs

Example 2: a) Diagonal dominance, b) Minimum eigenvalue, c) Best diagonal

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array} \qquad Q = \begin{bmatrix} 1 & 2 & -3 & 2 \\ 2 & 2 & -3 & 4 \\ -3 & -3 & 2 & 0 \\ 2 & 4 & 0 & -2 \end{bmatrix} \qquad \mathrm{eig}(Q) = \begin{bmatrix} -5.17 \\ -1.04 \\ 0.95 \\ 8.26 \end{bmatrix}$$

a) Diagonal dominance

$$\hat{Q} = \begin{bmatrix} 7 & 2 & -3 & 2 \\ 2 & 9 & -3 & 4 \\ -3 & -3 & 6 & 0 \\ 2 & 4 & 0 & 6 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} 6 \\ 7 \\ 4 \\ 8 \end{bmatrix} \quad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 1.66 \\ 4.90 \\ 6.88 \\ 14.56 \end{bmatrix} \qquad \begin{array}{ll} \min & x^T \hat{Q} x - \hat{q}^T x \\ s.t. & x \in [0,1]^4 \end{array}$$

$\mathbf{Optimal\ value} = -\mathbf{5.93}$
(continuous relaxation)

b) Minimum eigenvalue

$$\hat{Q} = \begin{bmatrix} 6.17 & 2 & -3 & 2 \\ 2 & 7.17 & -3 & 4 \\ -3 & -3 & 7.17 & 0 \\ 2 & 4 & 0 & 3.17 \end{bmatrix} \quad \hat{q} = \begin{bmatrix} 5.17 \\ 5.17 \\ 5.17 \\ 5.17 \end{bmatrix} \quad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 0 \\ 4.13 \\ 6.12 \\ 13.43 \end{bmatrix}$$

$$\begin{array}{ll} \min & x^T \hat{Q} x - \hat{q}^T x \\ s.t. & x \in [0,1]^4 \end{array}$$

$\mathbf{Optimal\ value} = -\mathbf{5.34}$
(continuous relaxation)

Convexification of 0-1 QPs

c) The best diagonal. The QCR method allows computation of the diagonal for which the highest optimal value of the relaxation is obtained.

$$\hat{Q} = \begin{bmatrix} 2.93 & 2 & -3 & 2 \\ 2 & 4.28 & -3 & 4 \\ -3 & -3 & 6.83 & 0 \\ 2 & 4 & 0 & 6.20 \end{bmatrix} \qquad \hat{q} = \begin{bmatrix} 1.93 \\ 2.28 \\ 4.83 \\ 8.20 \end{bmatrix} \qquad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 0 \\ 1.31 \\ 6.71 \\ 12.21 \end{bmatrix}$$

$$\begin{array}{ll} \min & x^T \hat{Q} x - \hat{q}^T x \\ s.t. & x \in [0,1]^4 \end{array}$$

$\mathbf{Optimal\ value} = \mathbf{-4.08}$
(continuous relaxation)

The convexifications **a-c** are exact convex reformulations of the (0-1) QP problem, but convex relaxations of the corresponding integer relaxed QP problem, resulting in the

LBs: $\mathbf{-5.93 \leq -5.34 \leq -4.08 \leq} \boxed{}$

Minimum value of the (0-1) QP

In order to solve the (0-1) QP efficiently it is desirable to obtain a tight convex reformulation.

Convexification of 0-1 QPs

c) The best diagonal. The QCR method allows computation of the diagonal that gives the highest optimal value of the relaxation.

$$\hat{Q} = \begin{bmatrix} 2.93 & 2 & -3 & 2 \\ 2 & 4.28 & -3 & 4 \\ -3 & -3 & 6.83 & 0 \\ 2 & 4 & 0 & 6.20 \end{bmatrix} \qquad \hat{q} = \begin{bmatrix} 1.93 \\ 2.28 \\ 4.83 \\ 8.20 \end{bmatrix} \qquad \mathrm{eig}(\hat{Q}) = \begin{bmatrix} 0 \\ 1.31 \\ 6.71 \\ 12.21 \end{bmatrix}$$

$$\begin{array}{ll} \min & x^T \hat{Q} x - \hat{q}^T x \\ s.t. & x \in [0,1]^4 \end{array}$$

Optimal value = −4.08
(continuous relaxation)

$$\begin{array}{ll} \min & x^T Q x \\ s.t. & x \in \{0,1\}^4 \end{array}$$

Optimal value = −3
(0-1) QP

The convexifications **a-c** are exact convex reformulations of the (0-1) QP problem, but convex relaxations of the corresponding integer relaxed QP problem, resulting in the

LBs: $\mathbf{-5.93 \leq -5.34 \leq -4.08 \leq}$ $\boxed{\mathbf{-3}}$

In order to solve the (0-1) QP efficiently it is desirable to obtain a tight convex reformulation.

Quadratic and semidefinite programming
Semidefinite relaxation of 0-1 QPs

$$\begin{array}{ll} \min & x^T Q x + q^T x \\ s.t. & Ax = a \\ & Bx \leq b \\ & x \in \{0,1\}^n \end{array}$$

The original problem can be relaxated into a semidefinite program by using a positive semidefinite matrix including the variables

$$X = xx^T \quad \mapsto \quad X - xx^T \succcurlyeq 0 \quad \Leftrightarrow \quad \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succcurlyeq 0$$

A quadratic expression in x is linear in X:

$$x^T Q x = tr\left(QX^T\right) = Q \bullet X = \sum_i \sum_j Q_{ij} X_{ij}$$

Binary condition: $x_i \in \{0,1\} \quad \Leftrightarrow \quad x_i^2 - x_i = 0 \Leftrightarrow \quad X_{ii} = x_i$

Semidefinite relaxation:

$$\begin{array}{ll} \min & Q \bullet X + q^T x \\ s.t. & Ax = a \\ & Bx \leq b \\ & \mathrm{diag}(X) = x \\ & \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succcurlyeq 0 \end{array}$$

Deriving a dual problem

Lagrangian relaxation of a 0-1 QP:

$$f(x,\lambda,\mu,\delta) = x^T Q x + q^T x + \lambda^T (Ax - a) + \mu^T (Bx - b) + \sum_{i=1}^{n} \delta_i (x_i^2 - x_i)$$

$$= x^T \underbrace{(Q + \mathrm{Diag}(\delta))}_{\bar{Q}} x + \underbrace{(q + A^T\lambda + B^T\mu - \delta)}_{\bar{q}}{}^T x \underbrace{-\lambda^T a - \mu^T b}_{\bar{c}}$$

Lagrangian dual problem:

$$\sup_{\delta,\lambda,\mu} \; \inf_{x \in R^n} \; x^T \bar{Q} x + \bar{q}^T x + \bar{c}$$

Inf (infinum) = GLB (the greates lower bound)
Sup (supremum) = LUB (the least upper bound)

which <u>equals</u> a semidefinite program

$$\begin{aligned} \max \quad & t \\ s.t. \quad & \begin{bmatrix} -t + \bar{c} & \frac{1}{2}\bar{q}^T \\ \frac{1}{2}\bar{q} & \bar{Q} \end{bmatrix} \succcurlyeq 0 \\ & \delta \in R^n, \lambda \in R^m, \mu \in R_+^k \end{aligned}$$

Thus, we have both a primal and a dual SDP relaxation. The latter <u>equals</u> the dual of the Lagrangian relaxation of the 0-1 QP problem, from which we can obtain the multipliers.

The primal and dual relaxations of the 0-1 QP

$$
\begin{array}{ll}
\min & Q \bullet X + q^T x \\
s.t. & Ax = a \\
 & Bx \leq b \\
 & \mathrm{diag}(X) = x \\
 & \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succcurlyeq 0
\end{array}
$$

$$
\begin{array}{ll}
\max & t \\
s.t. & \begin{bmatrix} -t + \bar{c} & \frac{1}{2}\bar{q}^T \\ \frac{1}{2}\bar{q} & \bar{Q} \end{bmatrix} \succcurlyeq 0 \\
 & \lambda \in R^m, \mu \in R_+^k, \delta \in R^n
\end{array}
$$

The solution of the dual relaxation give us the optimal values on the multipliers : $\lambda^*, \mu^*, \delta^*$.

The multipliers, δ^* , can then be used to construct the "best" diagonal perturbation of the matrix Q (recalling that the multipliers δ^* are from $\delta_i(x_i^2 - x_i)$ and affect the diagonal of Q in the Lagrangian) according to

$$Q^* = Q + \mathrm{Diag}(\delta^*).$$

The SDPT3 solver in CVX (Grant and Boyd) can, for example, be used to solve the SDP problems.

Strengthening the relaxation

Inclusion of other constraints can, however, still improve the bounding quality. There are many ways to include or construct quadratic constraints.

1) *Add new redundant quadratic constraints (for example, RLT (reformulation-linearization technique) constraints)*

$$x_i x_j \geq 0, \qquad x_i x_j \geq x_i + x_j - 1, \qquad x_i x_j \leq x_i, \qquad x_i x_j \leq x_j$$

2) *Combine and multiply existing linear constraints (some examples)*

$$p^T x = s \implies x_i p^T x = x_i s \quad \forall i$$
$$p^T x = s \implies (1 - x_i) p^T x = (1 - x_i) s \quad \forall i$$

$$\begin{cases} p^T x = s \\ r^T x = t \end{cases} \implies p^T x r^T x = st \implies x^T \left(p r^T \right) x = st$$

$$Ax = a \implies \|Ax - a\|^2 = 0 \implies x^T A^T A x = a^T a$$

Our strengthening

Original 0-1 QP

$$
\begin{array}{ll}
\min & x^T Q x + q^T x \\
s.t. & Ax = a \\
& Bx \le b \\
& x \in \{0,1\}^n
\end{array}
$$

Strengthened SDP relaxation

$$
\begin{array}{ll}
\min & Q \bullet X + q^T x \\
s.t. & Ax = a \\
& Bx \le b \\
& \operatorname{diag}(X) = x \\
& A^T A \bullet X = a^T a \\
& X_{ij} \ge 0, \quad X_{ij} \ge x_i + x_j - 1 \quad \forall\, i \ne j \\
& X_{ij} \le x_i, \quad X_{ij} \le x_j \quad \forall\, i \ne j \\
& \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succeq 0 \\
& x \in R^n,\ X \in S^n
\end{array}
$$

- Multipliers from all quadratic constraints are used to convexify the objective function so that the lower bound becomes as high as possible.

- Multipliers: $\delta \in R^n, \alpha \in R, \quad S, T, U, V \ge 0$

Convexified 0-1 QP problem

$$
\begin{array}{ll}
\min & x^T Q x + q^T x \\
s.t. & Ax = a \\
 & Bx \le b \\
 & x \in \{0,1\}^n
\end{array}
$$

$$
\begin{aligned}
\bar{Q}^* &= Q + \mathrm{Diag}(\delta^*) + \alpha^* A^T A - S^* - T^* + U^* + V^* \\
\bar{q}^* &= q + A^T \lambda^* + B^T \mu^* - \delta^* \\
\bar{c}^* &= -\lambda^{*T} a - \mu^{*T} - \alpha^* a^T a
\end{aligned}
$$

(MIQP)

$$
\begin{array}{ll}
\min & x^T \overline{Q}^* x + \overline{q}^{*T} x + \overline{c}^* + 2\sum_{i=1}^{n}\sum_{j=i+1}^{n}(S^*_{ij} + T^*_{ij}) y_{ij} - 2\sum_{i=1}^{n}\sum_{j=i+1}^{n}(U^*_{ij} + V^*_{ij}) z_{ij} \\
s.t. & Ax = a \\
 & Bx \le b \\
 & y_{ij} \ge 0, \quad y_{ij} \ge x_i + x_j - 1 \\
 & z_{ij} \le x_i, \quad z_{ij} \le x_j \\
 & x \in \{0,1\}^n \\
 & y_{ij}, \quad z_{ij} \in R_+ \ (i < j)
\end{array}
$$

The NDQCR method

Non-diagonal quadratic convex reformulation technique (NDQCR)
Given a general QP01 problem.

1. Strengthen the problem by including a set of RLT inequalities and squared norm constraints.
2. Solve the semidefinite relaxation (SDPr) and its dual (SDPd).
3. Collect the multiplier values and form problem MIQP.
4. Solve problem MIQP using any suitable solver.

The SDPT3 solver in CVX (Grant and Boyd) can, for example, be used to solve the SDP problems, to get the multipliers needed to form the MIQP problem.

The optimal solution of the (0-1) QP problem can then be obtain by solving the MIQP problem, for example, using CPLEX. The root node (when solving the MIQP) will end up with exactly the same objective value as was obtained when solving the dual SDP problem.

NDQCR versus QCR

Example 3:

$$\begin{aligned} \min \quad & x^T Q x + q^T x \\ & Ax = a \\ & x \in \{0,1\}^5 \end{aligned}$$

Optimal value = -80
of the (0-1) QP problem

$$Q = \begin{bmatrix} 0 & -24 & 2 & 18 & -12 \\ -24 & 0 & -3.5 & 18 & -42 \\ 2 & -3.5 & 0 & 20 & 2 \\ 18 & 18 & 20 & 0 & -44 \\ -12 & -42 & 2 & -44 & 0 \end{bmatrix}, \quad q = \begin{bmatrix} -9 \\ -7 \\ 2 \\ 23 \\ 12 \end{bmatrix}, \quad A^T = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \quad a = 2$$

i) α and δ perturbations (QCR method)

ii) α, δ and S perturbations

iii) α, δ and T perturbations

iv) α, δ and U perturbations

v) α, δ and V perturbations

Strategy	i)	ii)	iii)	iv)	v)
$v(*)$	-88.02	-80	-82.23	-82.20	-83.84

Lower bounds:
I.e. optimal values of the different dual SDPs.
Also equal to the root node values when solving the corresponding MIQPs.

Observe that, in this case, the optimal (0-1) QP solution will already be found in the root node when, using the ii) strategy, to form the MIQP. (The S multipliers in ii) are nondiagonal multipliers in the Q matrix obtained from including RLT constraints of the form $X_{ij} \geq 0$, while the α and δ multipliers affecting the diagonal of Q are from the squared norm constraint $x^T A^T A x = a^T a$ and the binary conditions $x_i^2 = x_i$.

NDQCR versus QCR

Best reformulation – strategy (ii)

Multipliers

$$S^* = \begin{bmatrix} 0 & 1.99 & 1.40 & 56.96 & 12.66 \\ 1.99 & 0 & 0 & 32.40 & 0 \\ 1.40 & 0 & 0 & 22.38 & 0 \\ 56.96 & 32.40 & 22.38 & 0 & 6.36 \\ 12.66 & 0 & 0 & 6.36 & 0 \end{bmatrix}, \quad \delta^* = \begin{bmatrix} -15.89 \\ 4.78 \\ 1.00 \\ -18.07 \\ -25.22 \end{bmatrix}, \quad \alpha^* = 113.32$$

Matrices

$$\overline{Q}^* = Q + \mathrm{Diag}\,(\delta^*) + \alpha^* A^T A - S^* = \begin{bmatrix} 97.44 & 87.33 & 0.60 & 74.36 & 88.66 \\ 87.33 & 118.10 & -3.50 & 98.92 & 71.32 \\ 0.60 & -3.50 & 1.00 & -2.38 & 2.00 \\ 74.36 & 98.92 & -2.38 & 95.26 & 62.96 \\ 88.66 & 71.32 & 2.00 & 62.96 & 88.10 \end{bmatrix}$$

$$\overline{q}^* = q - \delta^* = \begin{bmatrix} 6.89 \\ -11.78 \\ 1.00 \\ 41.07 \\ 37.22 \end{bmatrix}$$

$$\overline{c}^* = -\alpha^* a^T a = -453.29$$

Convexified QP

$$\begin{aligned} \min \quad & x^T\,\overline{Q}^* x + \overline{q}^{*T} x + \overline{c}^* + 2 \sum_{(i,j)\in I} S^*_{ij} y_{ij} \\ \text{s.t.} \quad & x_1 + x_2 + x_4 + x_5 = 2 \\ & y_{ij} \geq 0, \;\; y_{ij} \geq x_i + x_j - 1 \quad \forall\, (i,j) \in I \\ & x \in \{0,1\}^5 \end{aligned}$$

Coulomb glass problem

**Given *n* sites in the plane. *k* of these sites are filled with electrons.
Find the configuration that has minimal energy.**

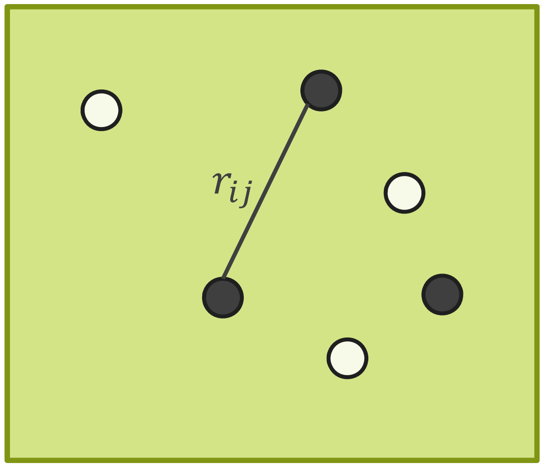
Energy = Coulomb interaction + site specific energy

Variables: $x \in \{0,1\}^n$

$$x_i = \begin{cases} 0, & \text{if site } i \text{ is empty} \\ 1\ , & \text{if site } i \text{ is filled} \end{cases}$$

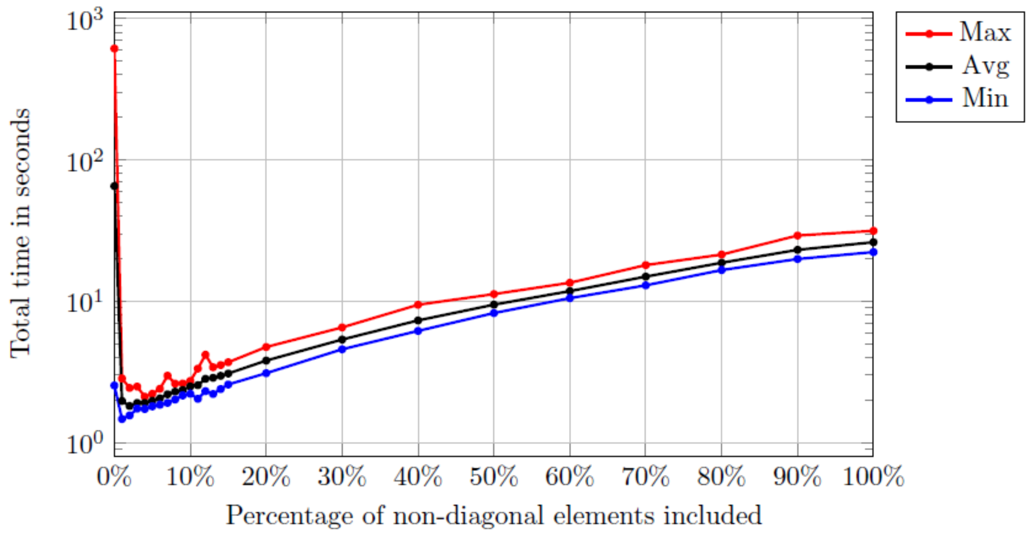
$$\boxed{\begin{array}{ll} \min & x^T Q x + q^T x \\ s.t. & e^T x = k \\ & x \in \{0,1\}^n \end{array}}$$

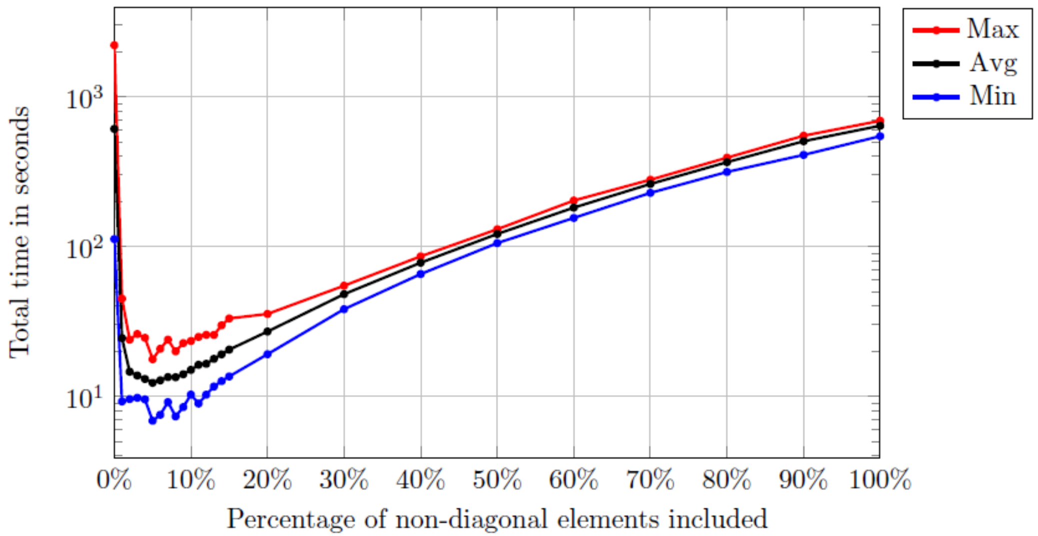
$$Q_{ij} = \begin{cases} 0, & \text{if } i = j \\ \dfrac{1}{2r_{ij}}, & \text{if } i \neq j \end{cases}$$



The Coulomb glass is a physical model for lightly doped semiconductors at very low temperature (a few K) where the conduction electrons are localized to certain impurity sites and the electrones strongly interact with each other.

NDQCR on Coulomb glass problems (n=50, n=100)





- $k = \frac{n}{2}$ in all experiments
- Constraints $X_{ij} \geq 0$ and $X_{ij} \geq x_i + x_j - 1$ are included for indeces corresponding to the p% largest elements of Q.
- Even a small fraction of non-diagonal elements has a large impact on the total solution time.
- 2% - 10% non-diagonal elements result in fastest solution times.

Boolean least squares

The problem is to identify a binary signal $x \in \{0,1\}^n$ from a collection of noisy measurements.

$$
\begin{aligned}
\min \quad & \|Dx - d\|^2 \\
\text{s.t.} \quad & x \in \{0,1\}^n
\end{aligned}
$$

$$
\begin{aligned}
\min \quad & D^T D \bullet X - 2d^T Dx + d^T d \\
\text{s.t.} \quad & \operatorname{diag}(X) = x \\
& \begin{bmatrix} 1 & x^T \\ x & X \end{bmatrix} \succcurlyeq 0 \\
& X_{ij} \geq 0, \ X_{ij} \geq x_i + x_j - 1 \\
& X_{ij} \leq x_i, \ X_{ij} \leq x_j
\end{aligned}
$$

Size (n)	MIQP Gap	Time	SDP Gap	Time	Total time
40	0.00 %	1.2	21.47 %	0.4	1.6
60	0.00 %	16.3	27.33 %	0.5	16.8
80	0.00 %	175.0	31.50 %	0.6	175.6
100	3.29 %	1849.9	37.59 %	0.8	1850.7

Table 1: Average results for BLS with QCR

Size (n)	MIQP Gap	Time	SDP Gap	Time	Total time
40	0.00 %	0.6	2.78 %	14.1	14.8
60	0.00 %	3.8	5.87 %	41.0	44.8
80	0.00 %	17.4	7.72 %	107.8	125.2
100	0.00 %	236.5	11.13 %	261.8	498.4

Table 3: Average results for BLS with NDQCR

Tai*xx*c instances from QAPLIB

These instances are special type QAP problems, where the flow matrix *F* has only binary elements and is of *rank* 1. These problems can, therefore, be rewritten into simplier (0-1) QP problems with only n binary variables and *Q*=*D* with n^2 elements (instedad of n^2 binaries and n^4 elements in the extended *Q* matrix, when using the tensor formulation). The flow matrix is first written in the form:

$$F = bb^T \text{ where } b \in \{0,1\}^n.$$

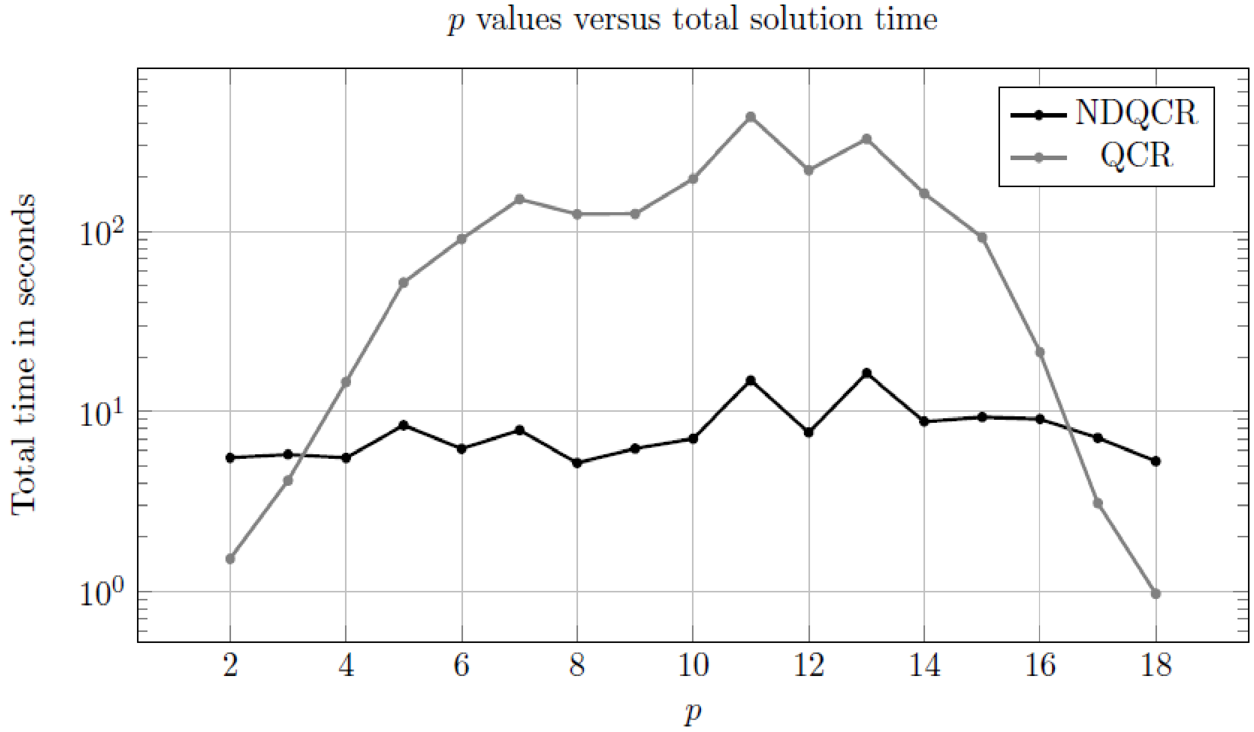
The objective function of the QAP can therafter be rewritten and simplified using the binary *rank*-1 property, as follows:

$$\begin{aligned} \operatorname{trace}(DXFX^T) &= \operatorname{trace}(DXbb^TX^T) = \operatorname{trace}(DXb(Xb)^T) \\ &= \operatorname{trace}(Dyy^T) = \operatorname{trace}(y^TDy) = y^TDy \end{aligned}$$

and we end up in a (0-1) QP of the following form:

$$\boxed{\begin{array}{ll} \min & y^TDy \\ \text{s.t.} & e^Ty = p \\ & y \in \{0,1\}^n \end{array}}$$

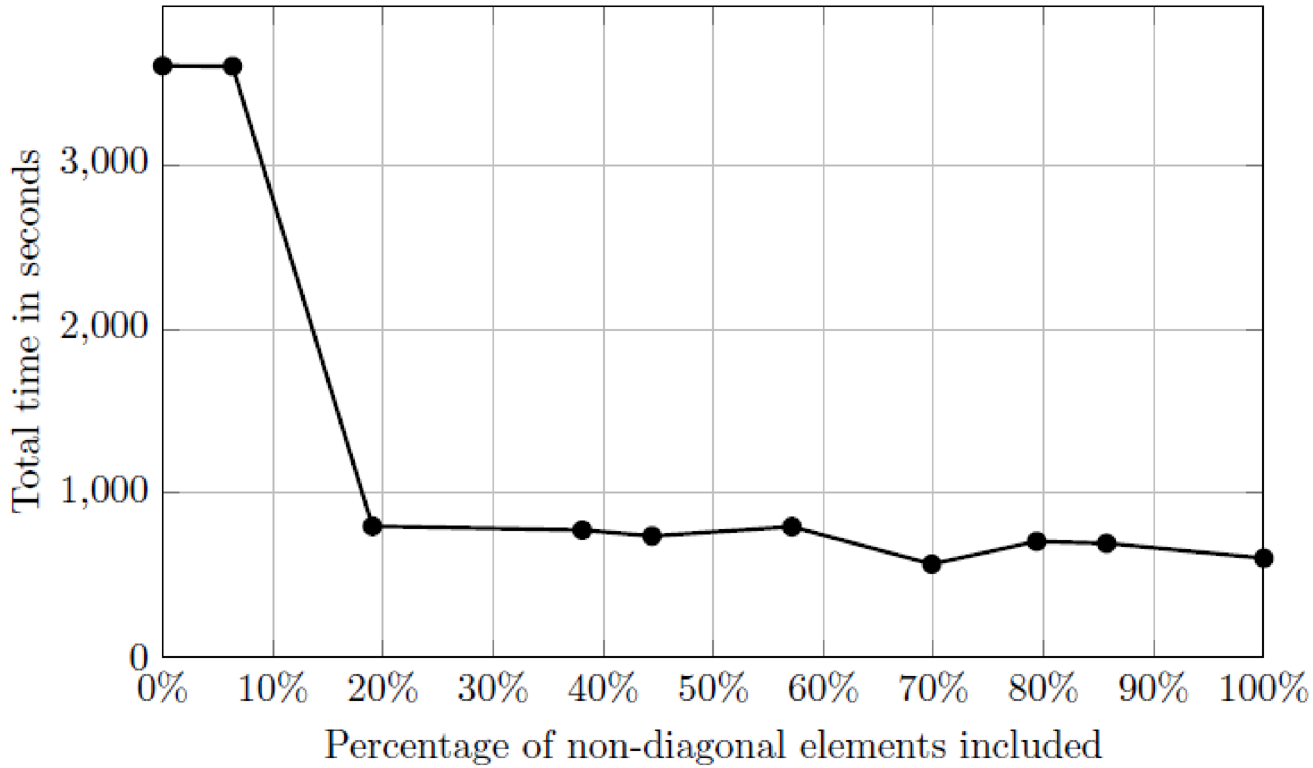
NDQCR (Y≥0) versus QCR on tai36c



$$\begin{array}{ll} \min & y^T D y \\ \text{s.t.} & e^T y = p \\ & y \in \{0,1\}^n \end{array}$$

$p=11$	QCR	NDQCR
CPU time	433 s	15 s
SDP gap	24 %	4 %

NDQCR on problem tai64c



Some results when solving large QAP problems from QAPLIB with the NDQCR technique

The Tai64c problem

Lower bound = 1 814 485 in 49sec when using NDQCR
Optimal solution = 1 855 928 in 529sec when using NDQCR *)

The Tai256c problem (Largest problem in QAPLIB)

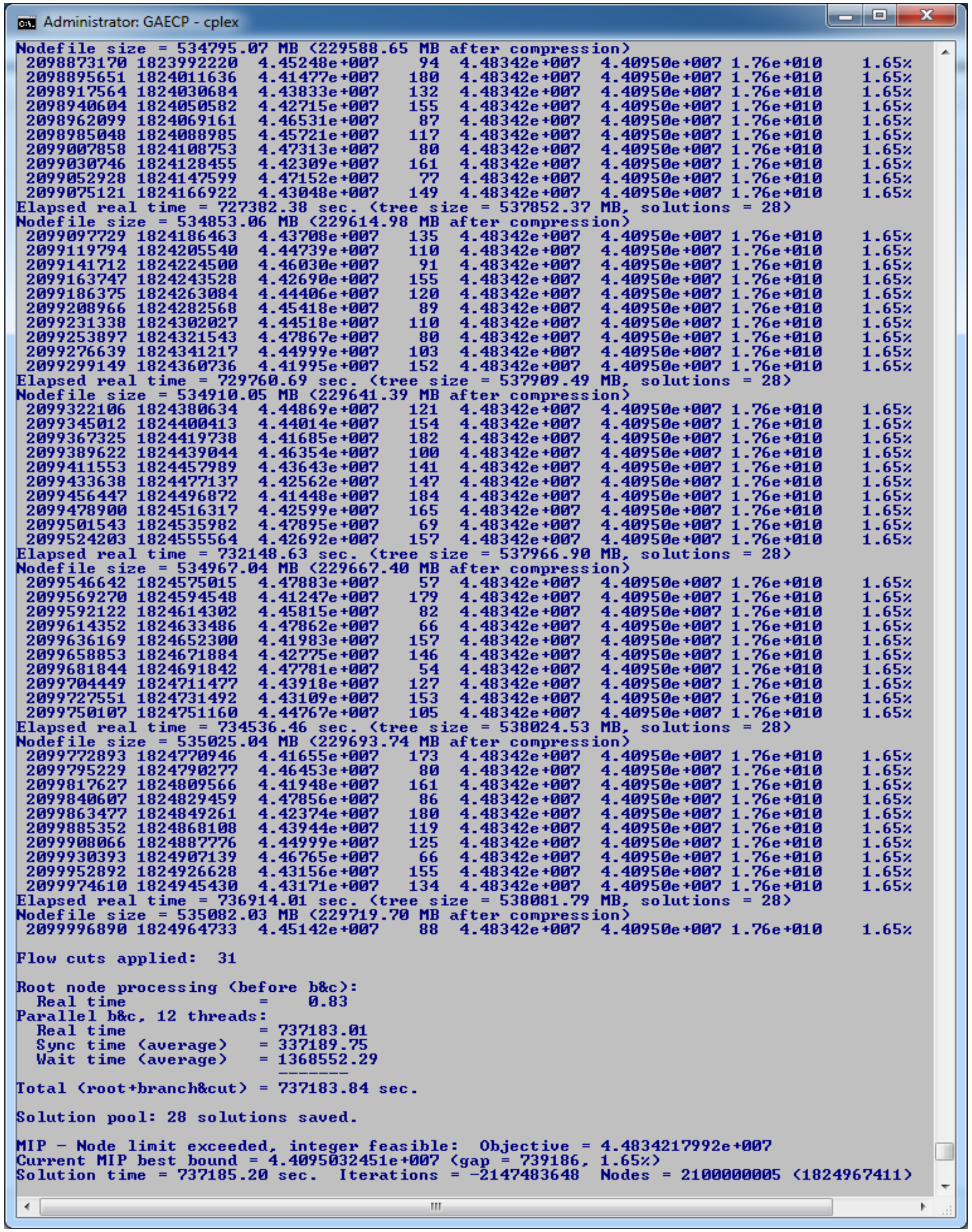
	NDQCR	Best known solutions in QAPLIB
Lower bound =	43 849 789 1a)	43 849 646 2)
	44 095 032 1b)	
Best solution =	(44 834 218) 1b)	44 759 294 3)
	GAP=1,65% (or 1.48%)	GAP=2.03%

*) Optimal solution verified by Drezner in (2005) in 2h 11min 45 sec.

1a) LB Obtained with the NDQCR in 2h 46 min 46 sec when solving the dual SDP relaxation.
1b) Best LB and UB obtained with the NDQCR technique after 8d 12h 46min 25sec when solving the MIQP problem, with CPLEX and it terminated because the node limit exceeded.

2) Obtained by Peng J., Mittelmann H. and Li X. (2008), in 4h 23min 7 sec using matrix splitting, (S-SVD)

3) Obtained by using Ant colony technique.



```
Administrator: GAECP - cplex
Nodefile size = 534795.07 MB (229588.65 MB after compression)
 2098873170 1823992220  4.45248e+007    94  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2098895651 1824011636  4.41477e+007   180  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2098917564 1824030684  4.43833e+007   132  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2098940604 1824050582  4.42715e+007   155  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2098962099 1824069161  4.46531e+007    87  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2098985048 1824088985  4.45721e+007   117  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099007858 1824108753  4.47313e+007    80  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099030746 1824128455  4.42309e+007   161  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099052928 1824147599  4.47152e+007    77  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099075121 1824166922  4.43048e+007   149  4.48342e+007  4.40950e+007 1.76e+010     1.65%
Elapsed real time = 727382.38 sec. (tree size = 537852.37 MB, solutions = 28)
Nodefile size = 534853.06 MB (229614.98 MB after compression)
 2099097729 1824186463  4.43708e+007   135  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099119794 1824205540  4.44739e+007   110  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099141712 1824224500  4.46030e+007    91  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099163747 1824243528  4.42690e+007   155  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099186375 1824263084  4.44406e+007   120  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099208966 1824282568  4.45418e+007    89  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099231338 1824302027  4.44518e+007   110  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099253897 1824321543  4.47867e+007    80  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099276639 1824341217  4.44999e+007   103  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099299149 1824360736  4.41995e+007   152  4.48342e+007  4.40950e+007 1.76e+010     1.65%
Elapsed real time = 729760.69 sec. (tree size = 537909.49 MB, solutions = 28)
Nodefile size = 534910.05 MB (229641.39 MB after compression)
 2099322106 1824380634  4.44869e+007   121  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099345012 1824400413  4.44014e+007   154  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099367325 1824419738  4.41685e+007   182  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099389622 1824439044  4.46354e+007   100  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099411553 1824457989  4.43643e+007   141  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099433638 1824477137  4.42562e+007   147  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099456447 1824496872  4.41448e+007   184  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099478900 1824516317  4.42599e+007   165  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099501543 1824535982  4.47895e+007    69  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099524203 1824555564  4.42692e+007   157  4.48342e+007  4.40950e+007 1.76e+010     1.65%
Elapsed real time = 732148.63 sec. (tree size = 537966.90 MB, solutions = 28)
Nodefile size = 534967.04 MB (229667.40 MB after compression)
 2099546642 1824575015  4.47883e+007    57  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099569270 1824594548  4.41247e+007   179  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099592122 1824614302  4.45815e+007    82  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099614352 1824633486  4.47862e+007    66  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099636169 1824652300  4.41983e+007   157  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099658853 1824671884  4.42775e+007   146  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099681844 1824691842  4.47781e+007    54  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099704449 1824711477  4.43918e+007   127  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099727551 1824731492  4.43109e+007   153  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099750107 1824751160  4.44767e+007   105  4.48342e+007  4.40950e+007 1.76e+010     1.65%
Elapsed real time = 734536.46 sec. (tree size = 538024.53 MB, solutions = 28)
Nodefile size = 535025.04 MB (229693.74 MB after compression)
 2099772893 1824770946  4.41655e+007   173  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099795229 1824790277  4.46453e+007    80  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099817627 1824809566  4.41948e+007   161  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099840607 1824829459  4.47856e+007    86  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099863477 1824849261  4.42374e+007   180  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099885352 1824868108  4.43944e+007   119  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099908066 1824887776  4.44999e+007   125  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099930393 1824907139  4.46765e+007    66  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099952892 1824926628  4.43156e+007   155  4.48342e+007  4.40950e+007 1.76e+010     1.65%
 2099974610 1824945430  4.43171e+007   134  4.48342e+007  4.40950e+007 1.76e+010     1.65%
Elapsed real time = 736914.01 sec. (tree size = 538081.79 MB, solutions = 28)
Nodefile size = 535082.03 MB (229719.70 MB after compression)
 2099996890 1824964733  4.45142e+007    88  4.48342e+007  4.40950e+007 1.76e+010     1.65%

Flow cuts applied:  31

Root node processing (before b&c):
  Real time             =    0.83
Parallel b&c, 12 threads:
  Real time             = 737183.01
  Sync time (average)   = 337189.75
  Wait time (average)   = 1368552.29
                          -------
Total (root+branch&cut) = 737183.84 sec.

Solution pool: 28 solutions saved.

MIP - Node limit exceeded, integer feasible:  Objective = 4.4834217992e+007
Current MIP best bound = 4.4095032451e+007 (gap = 739186, 1.65%)
Solution time = 737185.20 sec.  Iterations = -2147483648  Nodes = 2100000005 (1824967411)
```

Solution result when solving the MIQP problem, obtained for the tai256c problem, using the NDQCR technique with:

CPLEX 12.6.0.0

Computer:
Intel(R) CoreTM) i7 CPU X980 @ 3,33GHz 3.33GHz
RAM 8,00GB

64-bit Operating System Windows7

Conclusions

- A technique for non-diagonal quadratic convex reformulation (NDQCR) was presented.
- The non-diagonal elements were obtained by adding squared norm constraints and a set of redundant RLT inequalities in the Largrangian relaxation of the (0-1) QP.
- The NDQCR technique gives tight bounding reformulations and fast solution for small to medium sized ($n < 200$), (0-1) QP problems.
- Full application still difficult for (very) large (0-1) QP problems ($n > 300$) (>90 000 elements in the Q matrix!)
- However, inclusion of just a few RLT inequalities may still have a large impact on the solution time and bounding quality when solving large (0-1) QP problems.

Future work: Discover other set of inequalities to include.

Some references on quadratic reformulation

P.L. Hammer and A.A. Rubin. Some remarks on quadratic programming with 0-1 variables. *RAIRO*, 3:67–79, 1970.

Billionnet, A., Elloumi, S., Plateau, M.-C.: Improving the performance of standard solvers for quadratic 0-1 programs by a tight convex reformulation: the QCR method. *Discrete Applied Mathematics*. **157**(6) : 1185-1197 (2009)

Billionnet, A., Elloumi, S., Lambert, A.: Extending the QCR method to the case of general mixed integer program. *Mathematical Programming*. Available online DOI: 10.1007/s10107-010-0381-7 (2010)

O. Nissfolk, R. Pörn, T. Westerlund, F. Jansson, A mixed integer quadratic reformulation of the quadratic assignment problem with rank-1 matrix, in: I. A. Karimi, R. Srinivasan (Eds.), 11th International Symposium on Process Systems Engineering, Vol. 31 of Computer Aided Chemical Engineering, Elsevier, 2012, pp. 360 – 364. doi:10.1016/B978-0-444-59507-2.50064-0.

L. Galli, A. Letchford, A compact variant of the QCR method for quadratically constrained quadratic 0-1 programs, Optimization Letters (2013) 1–12.

Peng J., Mittelmann H., and Li X. (2008). New relaxations for quadratic assignment problems based on matrix splitting. *Manuscript*, October 17th 2008, 1-20.

Peng J., Mittelmann H., and Li X. (2010). A New relaxation framework for quadratic assignment problems based on matrix splitting. *Mathematical Programming Computation*, 2, 59-77.

Ji, S., Zheng, S., Sun,S. (2012). An improved convex 0-1 quadratic program reformulation for quadratic knapsack problems. *Pacific Journal of Optimization*, **8**, 75-87.

THANK YOU FOR YOUR ATTENTION