Decomposition methods for quadratic programming

Lucas Létocart
with E. Bettiol, I. Bomze, A. Ceselli, F. Rinaldi, E. Traversi

LIPN - CNRS - Univ. Sorbonne Paris Nord

MIP Europe
July 3rd, 2025

Context

Generic formulation

$$
\begin{aligned}
\min\ & f(x) = x^\top \bar{Q} x + q^\top x \\
\text{s. t.}\quad & x^\top \bar{A}_i x + a_i^\top x \le b_i, \quad \forall i = 1 \ldots, m \\
& x \in X.
\end{aligned}
$$

$$
\begin{gathered}
n \in \mathbb{N}, \\
X \subseteq \mathbb{R}^n \text{or} \quad X \subseteq \mathbb{N}^n \\
\bar{Q},\ \bar{A}_i \in \mathcal{S}^n \\
q,\ a_i \in \mathbb{R}^n \\
b_i \in \mathbb{R},
\end{gathered}
$$

Simplicial Decomposition for Convex Quadratic Problems (CQPs) *Enrico Bettiol (Artelys), Francesco Rinaldi (Univ. Padova), Emiliano Traversi (ESSEC) - Comput. Optim. Appl. 75(2): 321-360 (2020)*

A generic CQP reads as follows:

Convex Quadratic Problem (CQP)

$$(CQP) \quad \max\{f(x) = x^\top Qx + L^\top x | Ax \leq b, x \in \mathbb{R}^n\}\ .$$

- $Q \in \mathbb{Q}^{n\times n}$ convex.
- $L \in \mathbb{Q}^n$.
- The problem is decomposed keeping the original objective function in the master and the original constraints in the pricing.

Dantzig-Wolfe Reformulation and Quadratic Convexification for Binary Quadratic Problems (BQPs)

Alberto Ceselli (Univ. Milano) and Emiliano Traversi (ESSEC) - Math. Prog. C 14(1): 85-120 (2022)

A generic BQP reads as follows:

Binary Quadratic Problem (BQP)

$$(BQP) \quad \max\{f(x) = x^\top Qx + L^\top x | Ax \leq b, x \in \{0,1\}\} .$$

The continuous relaxation of (BQP) can be strengthened by convexifying some constraints.

Dantzig-Wolfe Reformulation and Boolean Quadric Polytope relaxation for binary QCQPs *Enrico Bettiol (Artelys), Immanuel Bomze (Univ. Vienna), Francesco Rinaldi (Univ. Padova), Emiliano Traversi (ESSEC) - Comp. & Oper. Res. 142: 105735 (2022)*

A generic BQCQP reads as follows:

Generic formulation

$$\begin{aligned} \min\ & f(x) = x^\top \bar{Q} x + q^\top x \\ \text{s. t.}\quad & x^\top \bar{A}_i x + a_i^\top x \le b_i, \quad \forall i = 1\ldots, m \\ & x \in \{0,1\}^n. \end{aligned}$$

$$\begin{gathered} x \in \mathbb{R}^n, n \in \mathbb{N}, \\ \bar{Q},\ \bar{A}_i \in \mathcal{S}^n \\ q,\ a_i \ \in \mathbb{R}^n \\ b_i \in \mathbb{R}, \end{gathered}$$

The problem is decomposed solving the extended matrix formulation in the master and keeping the rank-1 constraint in the pricing.

Outline

Settings

The problem

$$\begin{aligned} \min \quad & f(x) = x^T Qx + c^T x \\ s.\ t. \quad & x \in X. \end{aligned}$$

where $x \in \mathbb{R}^n$, $X \subset \mathbb{R}^n$.

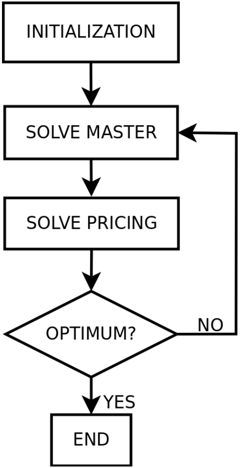
Hypotheses:

- $Q \in \mathbb{R}^{n \times n}$ is positive semidefinite and dense.
- X is a polytope.
- High number of variables and low number of constraints.

A column generation method

Simplicial Decomposition (SD)

- Master problem: original objective function, optimized over a simplex.
- Pricing problem: linear objective function, original domain.
- All the original constraints are in the pricing.
- Finite convergence.



The master problem

At a k-th iteration, k vertices $x_1, \dots x_k \in X$ are provided. $k \ll n$.

Master problem

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s. t.} \quad & x = \sum_{i=1}^{k} \omega_i x_i, \\ & \sum_{i=1}^{k} \omega_i = 1, \\ & \omega_i \geq 0, \quad \forall i = 1, \dots, k. \end{aligned}$$

The pricing problem

Pricing problem

$$\begin{aligned} \min \quad & \nabla f(x_m)^T x \\ \text{s. t.} \quad & x \in X. \end{aligned}$$

- Linearization of the original objective function in the optimal point x_m of the master.
- Same dimension as the original problem.
- Same constraints as the original problem.

Master: SD - ACDM

Adapted Conjugate Direction Method (ACDM)

Main ideas:

- Based on the conjugate Directions Method.
- Reuse the informations from previous iteration.
- Exploit the special structure of the simplices generated.

The Conjugate Direction Method

- Two directions $d_1, d_2 \in \mathbb{R}^k$ are conjugated with respect to the positive definite quadratic matrix $Q \in \mathbb{R}^{k \times k}$ if: $d_1^T Q d_2 = 0$.
- If we have a set of k conjugate directions $D = \{d_1, \ldots, d_k\}$, the minimum of $f(x) = x^T Qx + c^T x$ can be found in k steps by optimizing in sequence over the k conjugate directions.

Master: SD - ACDM

Adapted Conjugate Direction Method (ACDM)

- Reuse the information from the previous conjugate directions.
- Exploit the special structure of the simplices generated.

Main Steps

- At iteration k, we have a set of $k-1$ conjugate directions D from the previous iteration.
- The pricing provides a new point x_k (i.e., a new dimension).
- Find a new direction d_k connecting $x_{k-1}$with x_k and conjugate it w.r.t. the set D.
- Find new optimal point along this direction.

PRO: Most of the times, only one step.
CON: If the optimum is on a face, all the directions must be recalculated.

Master: SD - FGPM

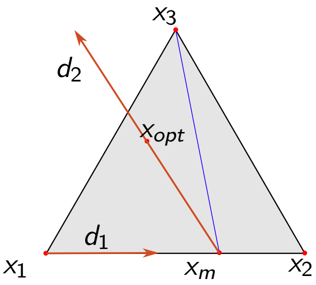
A Fast Gradient Projection Method, (FGPM)

A more general method based on the projected gradient approach.
Warmstart: start in the previous optimal point.
Iteratively, given the k-th point $\tilde{x_k}$:

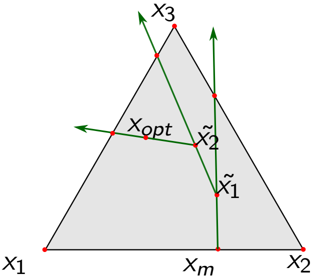
- compute the gradient $\nabla f(\tilde{x_k})$;
- project the point $y_k = \tilde{x_k} - s\nabla f(\tilde{x_k})$ onto the simplex;
- if $y_k \neq \tilde{x_k}$, find α_k with an Armijo-like rule;
- compute $\widetilde{x_{k+1}} = \tilde{x_k} + \alpha_k(p(y_k) - \tilde{x_k})$.

Master solvers

ACDM



FGPM



Pricing improvements I

Adding cuts

- Reduce the search region;
- exclude vectors that give ascent directions with respect to the previous partial optima;
- add cuts of the form

$$\nabla f(x_m^i)^T (x - x_m^i) \leq 0, \quad \exists i \in \{1, \ldots, k-1\}.$$

Early stopping

- Stop the computation before reaching the optimum, but ensure a descent direction: generate the point $\bar{x_k}$ s. t.

$$\nabla f(x_k)^T (\bar{x_k} - x_k) \leq -\varepsilon < 0.$$

Pricing improvements II

Sifting

Consider the *Sifting* options for the Cplex solver in addition to the default primal simplex.

Sifting is a column generation algorithm:

- it solves the problem with a (small) subset of columns;
- it evaluates the reduced costs of the remaining columns;
- columns that violate the optimality condition are inserted.

Problem instances

Chebichev center problem (Literature and randomly generated data)

$$\begin{aligned} \min \ & f(x) = x^\top A^\top Ax - \sum_{i=1}^{n} \|c_i\|^2 x_i \\ \text{s.t.} \ & e^\top x = 1, \\ & x \geq 0, \end{aligned}$$

Lasso problem (Literature and randomly generated data)

$$\begin{aligned} \min \ & f(x) = \|Ax - b\|_2^2 \\ \text{s.t.} \ & \|x\|_1 \leq \tau, \\ & x \in \mathbb{R}^n \end{aligned}$$

Problem instances

Portfolio optimization problem
(Markowitz's formulation)
(Literature data)

$$\begin{aligned} \min\ & f(x) = x^T \Sigma x \\ \text{s. t.}\ & r^T x \geq \mu, \\ & e^T x = 1, \\ & x \geq 0. \end{aligned}$$

General quadratic problems
(Randomly generated)

$$\begin{aligned} \min\ & f(x) = x^T Q x + c^T x \\ \text{s. t.}\ & Ax \geq b, \\ & 0 \leq x \leq 1. \end{aligned}$$

Problem instances

Quadratic shortest path problems
(Literature and randomly generated data)

$$
\begin{aligned}
\min\ & f(x) = x^\top Q x + c^\top x \\
\text{s. t.}\ & \sum_{e \in \delta^+(s)} x_s = 1, \\
& \sum_{e \in \delta^+(v)} x_v - \sum_{e \in \delta^-(v)} x_v = 0, \quad \forall v \neq s, t \\
& \sum_{e \in \delta^-(t)} x_t = 1.
\end{aligned}
$$

Multidimensional quadratic knapsack problem
(Literature and randomly generated data)

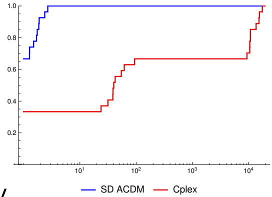
$$
\begin{aligned}
\min\ & f(x) = x^\top Q x + c^\top x \\
\text{s. t.}\ & \sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad \forall i = 1, 2, \dots, m \\
& 0 \leq x \leq 1.
\end{aligned}
$$

Instances

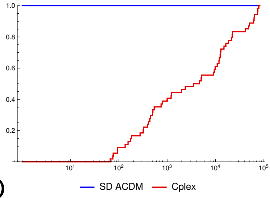
- Chebichev center problem: 27 instances, dimension 2048 to 8192.
- Lasso problem: 54 instances, dimension 2048 to 8192.
- Portfolio Optimization (PO): 40 instances, dimension 225 to 10980.
- General quadratic :
  - small m (GS): 450 instances:
    $n = 2000$ to 10000, $m = 2$ to 42.
  - large m (GL): 750 instances:
    $n = 2000$ to 10000, $m = n/32$ to $n/2$.
- Quadratic shortest path problems : grid and random shortest path instances ($1000 \leq n \leq 10000$) : 102 instances.
- Multidimensional quadratic knapsack problem : 54 instances:
  - ORLib dataset and GK dataset ($n \geq 1000$).
  - Randomly generated instances ($5000 \leq n \leq 10000$).

Hardware and software

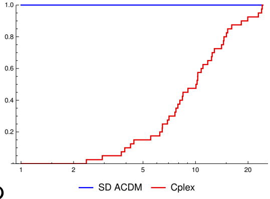
1. *IBM Ilog Cplex v.12.6.3*.
2. Intel Xeon E5-2650 v3 (2,3GHz), 64 GB of RAM using only one core.



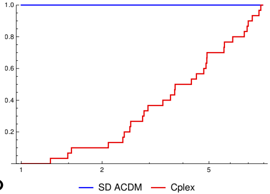
Chebyshev



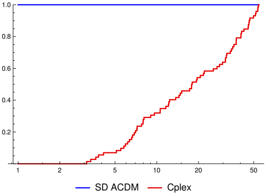
LASSO



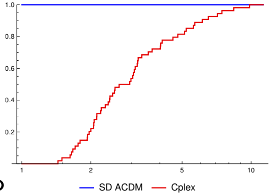
Portfolio



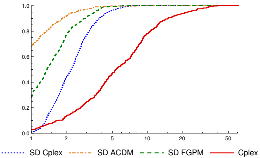
QGSPP

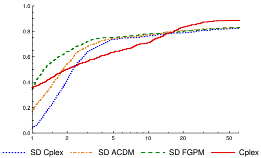


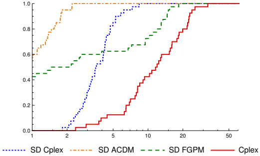
QRSPP

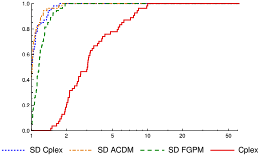


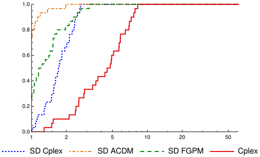
QMKP

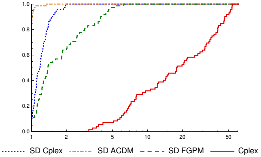
GS


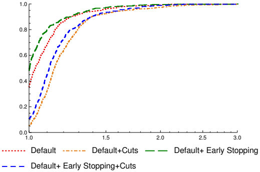
GL


POP


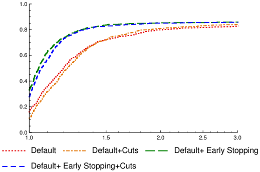
QMKP


QGSPP


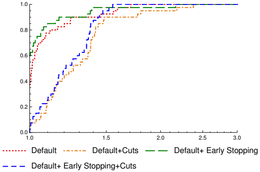
QRSPP




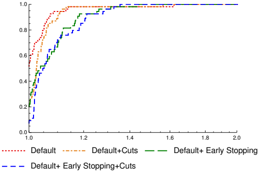
GS



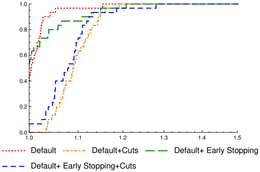
GL



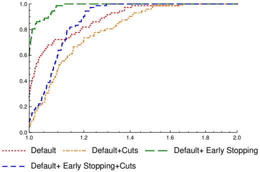
POP



QMKP



QGSPP



QRSPP

Class	N inst	NS Cplex	T Cplex	NS SD	T SD	N it
GS	450	450	11.7	450	2.4	171.4
GL	750	666	63.8	750	16.7	90.7
POP	40	40	9.6	40	0.7	116.6
QMKP	54	54	36.5	54	11.7	31.7
QGSPP	30	30	77.6	30	15.0	290.4
QRSPP	72	72	2.2	72	0.1	19.7

Table: Solved instances and average CPU time.

Settings

The problem

$$\begin{aligned} \min\ & f(x) = x^\top Q x + c^\top x \\ \text{s. t.}\quad & Ax \geq b \\ & Cx = d \\ & l \leq x \leq u \\ & x_i \in \{0,1\} \quad \forall i \in I \subseteq \{1,\dots,n\} \end{aligned}$$

with $Q \in \mathbb{R}^{n \times n}$ positive semidefinite and dense, $c, l, u \in \mathbb{R}^n$, $A \in \mathbb{R}^{m_1 \times n}$, $b \in \mathbb{R}^{m_1}$, $C \in \mathbb{R}^{m_2 \times n}$, $d \in \mathbb{R}^{m_2}$, $n, m_1, m_2 \in \mathbb{N}$.

Branch & Bound Scheme

- Valid dual bounds at each node given before the end of the SD procedure
- Branching rule: we fix to 1 the variable with the fractional part closer to one
- Tree exploration: Depth First Search
- Warmstart: reuse of columns of the father node
- Column Projection

Problem instances

Quadratic Minimum Spanning Tree (QMST) Problem (Randomly generated data)

Given an undirected graph G, the problem here is to find the subtree of G spanning all the vertices of G with minimum cost, where the cost is given by the total weight of the edges and the sum of interaction costs over all pairs of edges on the tree.

Quadratic Shortest Path (QSP) Problem (Randomly generated data)

Given a directed graph, with one source (s) and one destination (t) node, the problem is to find the path between s and t with minimum cost, where the cost is given by the total weight of the arcs and the sum of interaction costs over all pairs of arcs on the path.

Results

n	m	SDBB-basic			SDBB-advanced		
		T(s)	#nodes	#its	T(s)	#nodes	#its
10	30	0.1	776	6526	0.1	777	3932
10	45	2.3	9143	102233	2.0	9139	66368
15	35	0.4	2183	19411	0.3	2190	11119
15	70	285.9	603770	9751849	241.9	603836	6481872
15	105	2001.6	3087969	55444482	1719.3	3087306	36940699
20	64	73.9	149602	2365146	60.1	149529	1448596
25	40	0.3	1474	12959	0.2	1476	6383
36	60	60.0	180790	1879460	37.2	180177	827665
49	84	5353.3	9175201	126114913	3175.3	9144096	58401499
Average		653.0	1125163	16418334	431.3	1122283	8586314

Table: Results for QMST instances.

Results

n	m	Cplex		SDBB-basic			SDBB-advanced		
		T(s)	#nodes	T(s)	#nodes	#its	T(s)	#nodes	#its
100	180	5.7	5295	7.4	7596	43674	4.1	7644	21393
121	220	20.7	13039	27.8	23047	144249	15.4	23020	69217
144	264	62.4	29081	76.0	45546	346718	43.4	45533	177583
169	312	349.1	117044	364.8	188983	1466319	208.9	189083	737974
196	364	1152.1	281713	1177.2	494343	4094175	687.9	493437	2076179
225	420	2829.7	483860	2823.6	847296	8776642	1667.3	847058	4587965
Averages		736.6	155005	746.1	267802	2478629	437.8	267629	1278385

Table: Results for QGSP instances.

Results

type	Cplex		SDBB-basic			SDBB-advanced		
	T(s)	#nodes	T(s)	#nodes	#its	T(s)	#nodes	#its
QGSP	736.6	155005	746.1	267802	2478629	437.8	267629	1278385
QRSP	75.0	47.7	41.7	829.4	9173.7	13.2	829.5	2372.2

Table: Quadratic Shortest Path Problem.

Dantzig-Wolfe decomposition for Binary Quadratic Problems (BQPs)

A generic $0-1$ nonlinear problem reads as follows:

$0-1$ nonlinear problem (BF)

$$
\begin{aligned}
(\text{BF})\ \ \min\ & f(x) & \\
\text{s.t.}\ & g_i(x) \leq 0 & i \in I \\
& h_j(x) \leq 0 & j \in J \\
& x \in \{0,1\}^n &
\end{aligned}
$$

where $h_j(x), j \in J$ are convex.

DWR with quadratic master problem I

$$
\begin{array}{llll}
\text{(F-NLM)} & \min & f(x) & \\
& \text{s.t.} & g_i(x) \leq 0 & i \in I \ [\mu_i] \\
& & x - \sum_{p \in \mathcal{P}} x^p y^p = 0 & [\pi] \\
& & \sum_{p \in \mathcal{P}} y^p - 1 = 0 & [\pi_0] \\
& & y^p \geq 0 & \forall p \in \mathcal{P} \\
& & x \in [0,1]^n & ,
\end{array}
$$

where μ_i, π and π_0 denote the dual variables of the associated constraints and $\mathcal{P}$ being the set of extreme points of
$\Omega = \text{conv}\{x : h_j(x) \leq 0 \ \ j \in J, x \in \{0,1\}^n\}$.

Pricing problem

$$
\begin{array}{rll}
\min & -\pi^{*\top} x + \pi_0^* & \\
\text{s.t.} & h_j(x) \leq 0 & j \in J \\
& x \in \{0,1\}^n &
\end{array}
$$

- the pricing problem is (binary) linear.

DWR with quadratic pricing problem I

$$
\begin{array}{llll}
\text{(F-NLP)} & \min & \sum_{p \in \mathcal{P}} f(x^p) y_p & \\
& \text{s.t.} & \sum_{p \in \mathcal{P}} g_i(x^p) y_p \leq 0 & i \in I \ [\mu_i] \\
& & \sum_{p \in \mathcal{P}} y^p = 1 & [\pi_0] \\
& & y^p \geq 0 & \forall p \in \mathcal{P}
\end{array}
$$

where μ_i and π_0 denote the dual variables of the associated constraints and $\mathcal{P}$ being the set of extreme points of $\Omega = \text{conv}\{x : h_j(x) \leq 0 \ \ j \in J, x \in \{0,1\}^n\}$.

Pricing problem

$$
\begin{aligned}
\max \quad & \pi_0^* - \sum_{i \in I} g_i(x)\mu_i^* - f(x) & \\
\text{s.t.} \quad & h_j(x) \leq 0 & j \in J \\
& x \in \{0,1\}^n &
\end{aligned}
$$

where $f(x)$ is not required to be convex.

- the pricing problem is (binary) quadratic.

Binary Quadratic Problem

$$
\begin{aligned}
(\text{BQ}) \quad \min \ f(x) = x^\top Q x + L^\top x & \\
\text{s.t.} \quad & Gx \le g \\
& Hx \le h \\
& x \in \{0,1\}^n \, .
\end{aligned}
$$

Reformulations

Reformulation of the objective function $f(x)$

QCR/MIQCR Method

($f(x)$ can be reformulated by exploiting the property $x^2 = x$ and the constraints)

$\Downarrow$

convex problem

Reformulation of the feasible region $\{x | Hx \leq h, x \in \{0,1\}\}$

Dantzig-Wolfe Reformulation

(a subset of constraints is substituted by its convex hull)

$\Downarrow$

tighter formulation

Hierarchy of reformulations

$$
\begin{array}{ccccc}
(\text{Q-QM}) & \Leftarrow & (\text{Q}_{\delta,\rho}\text{-QM}) & \Leftarrow & (\text{Q}_{\delta,\rho,\Gamma}\text{-QM}) \\
\not\Updownarrow & \not\Nearrow & \Uparrow & & \Uparrow \\
(\text{Q-QP}) & \Leftarrow & (\text{Q}_{\delta,\rho}\text{-QP}) & \Leftrightarrow & (\text{Q}_{\delta,\rho,\Gamma}\text{-QP})
\end{array}
$$

(K) Formulation I

Notations

n : number of items
a_j : weight of item j ($j = 1, \ldots, n$)
b : capacity of the knapsack
c_{ij} : profit associated with the selection of items i and j ($i, j = 1, \ldots, n$)
k : number of items to be filled in the knapsack

Assumptions

$c_{ij} \in \mathbb{N}$ $i, j = 1, \ldots, n$, $a_j \in \mathbb{N}$ $j = 1, \ldots, n$, $b \in \mathbb{N}$
$\max_{j=1,\ldots,n} a_j \leq b < \sum_{j=1}^{n} a_j$
$k \in \{1, \ldots, k_{max}\}$

(K) Formulation II

Mathematical formulation

$$
\begin{array}{llll}
(BK) & \max & f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_i x_j & \\
& \text{s.t.} & \sum_{j=1}^{n} a_j x_j \leq b & (kn) \\
& & \sum_{j=1}^{n} x_j = k & (ca) \\
& & x_j \in \{0, 1\} & j = 1, \ldots, n
\end{array}
$$

- without constraint (2): the 0-1 quadratic knapsack problem (QKP)
- without constraint (1): the k-cluster problem

Reformulations for (K)

		constraints in the pricing	Objective function $f(x)$	$f_{\delta,\rho}(x)$	$f_{\delta,\rho,\Gamma}(x,z)$
DWR	QM	(kn)	NCM	$(K_{\delta,\rho}$-QM-$kn)$	$(K_{\delta,\rho,\Gamma}$-QM-$kn)$
		(ca)	NCM	IPP	IPP
		(kn,ca)	NCM	$(K_{\delta,\rho}$-QM-$kn,ca)$	$(K_{\delta,\rho,\Gamma}$-QM-$kn,ca)$
	QP	(kn)	(K-QP-kn)	$(K_{\delta,\rho}$-QP-$kn)$	$(K_{\delta,\rho,\Gamma}$-QP-$kn)$
		(ca)	(K-QP-ca)	$(K_{\delta,\rho}$-QP-$ca)$	$(K_{\delta,\rho,\Gamma}$-QP-$ca)$
		(kn,ca)	POP	POP	POP

Experimental environment

- Carried out on an Intel i7-2600 quad core 3.4 GHz with 8 GB of RAM, using only one core
- CSDP integrated into COIN-OR for solving SDP programs
- CPLEX 12.6.3 with default settings
- Average values over 10 instances
- $n \in \{50, 60, \ldots, 100\}$
- $k \in [1, n/4]$, $b \in [50, 30k]$, a_j, $c_{ij} \in [1, 100]$

Dens	n	$(K\delta^*, \rho^*)$ Gap	$(K\delta^*, \rho^*, \Gamma^*)$ Gap	$(K\delta^*, \rho^*\text{-QM-}kn,ca)$ Time	Gap	$(K\delta^*, \rho^*, \Gamma^*\text{-QM-}kn,ca)$ Time	Gap	$(K\delta^*, \rho^*\text{-QP-}kn)$ Time	Gap
25%	50	38.5%	30.9%	1.1	38.4%	3.5	29.1%	0.3	0.0%
	60	84.5%	72.0%	1.7	84.4%	3.5	69.2%	0.4	0.0%
	70	42.4%	34.4%	2.5	42.2%	7.7	32.9%	1.0	0.0%
	80	37.9%	30.7%	3.1	37.7%	15.1	29.6%	7.3	0.0%
	90	76.7%	63.8%	3.1	76.7%	14.0	61.2%	15.2	0.0%
	100	60.6%	53.1%	4.1	60.6%	24.9	52.3%	49.9	0.0%
50%	50	31.4%	25.3%	1.4	31.1%	3.3	23.7%	0.3	0.0%
	60	19.8%	15.1%	1.8	19.5%	6.3	14.2%	0.5	0.0%
	70	57.0%	50.9%	2.6	56.9%	11.4	48.9%	3.2	0.0%
	80	64.1%	55.4%	3.0	64.1%	10.5	53.7%	9.6	0.0%
	90	63.8%	57.0%	3.2	63.8%	22.0	55.9%	36.3	0.0%
	100	18.5%	14.2%	4.5	18.5%	29.7	14.0%	107.4	0.0%
75%	50	70.9%	64.6%	1.5	70.9%	2.9	61.9%	0.2	0.0%
	60	86.3%	79.8%	1.5	86.0%	4.1	76.8%	1.0	0.0%
	70	58.2%	50.9%	2.1	58.2%	6.6	49.3%	4.8	0.0%
	80	54.0%	48.7%	2.4	54.0%	14.8	47.7%	39.2	0.0%
	90	42.1%	37.9%	3.6	42.1%	26.4	37.1%	353.5	0.0%
	100	27.2%	21.7%	4.6	27.2%	35.1	21.5%	353.6	0.0%
100%	50	50.4%	45.8%	1.4	50.3%	3.5	44.0%	0.8	0.0%
	60	48.8%	43.6%	1.9	48.8%	5.8	42.4%	1.0	0.0%
	70	44.5%	39.6%	2.5	44.4%	11.7	38.7%	6.9	0.0%
	80	49.2%	43.3%	2.7	49.2%	12.1	42.1%	66.4	0.0%
	90	27.0%	23.9%	3.7	27.0%	24.3	23.4%	69.6	0.0%
	100	59.8%	55.1%	3.5	59.8%	24.3	54.1%	319.8	0.0%
Avg.		50.6%	44.1%	2.6	50.5%	13.5	42.6%	60.3	0.0%

Table: DD instances, continuous relaxations of QCR and DWR with quadratic Master and quadratic Pricing.

The best reformulation I

$$
\begin{aligned}
(\mathrm{K}_{\delta,\rho}\text{-QP-}kn) \quad \max \quad & \sum_{p\in\mathcal{P}_{\mathrm{kn}}} c^{p}_{\delta^*,\rho^*} y^p && \\
\text{s.t.} \quad & \sum_{p\in\mathcal{P}_{\mathrm{kn}}} \sum_{j=1}^{n} x_j^p y^p = k && [\mu] \\
& \sum_{p\in\mathcal{P}_{\mathrm{kn}}} y^p = 1 && [\pi_0] \\
& y^p \geq 0 \qquad\qquad p \in \mathcal{P}_{\mathrm{kn}} &&
\end{aligned}
$$

with $\mathcal{P}_{\mathrm{kn}}$ being the set of extreme points of $conv\{x | \sum_{j=1}^{n} a_j x_j \leq b, x \in \{0,1\}^n\}$, and μ and π_0 being the dual variables associated to the constraints in the Master Problem.

The best reformulation II

Denoting as μ^* and π_0^* the optimal dual variables of a master solution during a column generation iteration, the Pricing Problem can be written as follows:

$$\begin{aligned} \max \quad & f_{\delta^*,\rho^*}(x) + \sum_{i=1}^{n} \mu_i^* x_i + \pi_0^* \\ \text{s.t.} \quad & \sum_{j=1}^{n} a_j x_j \leq b \\ & x_j \in \{0,1\} \qquad\qquad j = 1, \ldots, n\,. \end{aligned}$$

Dantzig-Wolfe Reformulation and Boolean Quadric Polytope relaxation for (BQCQP)

Generic formulation

$$\begin{aligned} \min\ & f(x) = x^\top \bar{Q} x + q^\top x \\ \text{s. t.}\quad & x^\top \bar{A}_i x + a_i^\top x \leq b_i, \quad \forall i = 1 \dots, m \\ & x \in \{0,1\}^n. \end{aligned}$$

$x \in \mathbb{R}^n, n \in \mathbb{N},$
$\bar{Q},\ \bar{A}_i \in \mathcal{S}^n$
$q,\ a_i \in \mathbb{R}^n$
$b_i \in \mathbb{R},$

Since $x_i^2 = x_i$:
Let $Q = \bar{Q} + q,\ A_i = \bar{A}_i + a_i, \rightarrow$

Compact formulation

$$\begin{aligned} \min\ & x^\top Q x \\ \text{s. t.}\quad & x^\top A_i x \leq b_i, \quad \forall i \\ & x \in \{0,1\}^n. \end{aligned}$$

Dantzig-Wolfe Reformulation and Boolean Quadric Polytope relaxation for (BQCQP)

Generic formulation

$$\begin{aligned} \min\ & f(x) = x^\top \bar{Q} x + q^\top x \\ \text{s. t.}\quad & x^\top \bar{A}_i x + a_i^\top x \leq b_i, \quad \forall i = 1 \dots, m \\ & x \in \{0,1\}^n. \end{aligned}$$

$$\begin{gathered} x \in \mathbb{R}^n, n \in \mathbb{N}, \\ \bar{Q},\ \bar{A}_i \in \mathcal{S}^n \\ q,\ a_i\ \in \mathbb{R}^n \\ b_i \in \mathbb{R}, \end{gathered}$$

Since $x_i^2 = x_i$:

Let $Q = \bar{Q} + q,\ A_i = \bar{A}_i + a_i, \rightarrow$

Compact formulation

$$\begin{aligned} \min\ & x^\top Q x \\ \text{s. t.}\quad & x^\top A_i x \leq b_i, \quad \forall i \\ & x \in \{0,1\}^n. \end{aligned}$$

Matrix space

The problem can be written in matrix form:

$$\langle M, X \rangle := \mathrm{Tr}(M^\top X).$$

Extended formulation

$$\begin{aligned} \min\ & \langle Q, X \rangle \\ \text{s. t.}\ & \langle A_i, X \rangle \le b_i, \quad \forall i = 1\dots, m \\ & X = xx^T \\ & x \in \{0,1\}^n \end{aligned}$$

Relaxing constraint

We relax the constraint

$$X = xx^T$$

and let X be in the *convex hull* of 0-1 rk-1 matrices:

$$X = \sum_{p=1}^{2^n} x_p x_p^\top \lambda_p$$

$$\sum_{p=1}^{2^n} \lambda_p = 1$$

$$\lambda \geq 0$$

$$x_p \in \{0,1\}^n.$$

Definition: (Restricted) Boolean Quadric Polytope (BQP) of size n

$$BQP_n = Conv\left\{X \in \mathbb{R}^{n\times n} \,|\, X = xx^\top,\ x \in \{0,1\}^n\right\}$$

Relaxing constraint

We relax the constraint

$$X = xx^T$$

and let X be in the *convex hull* of 0-1 rk-1 matrices:

$$X = \sum_{p=1}^{2^n} x_p x_p^\top \lambda_p$$

$$\sum_{p=1}^{2^n} \lambda_p = 1$$

$$\lambda \geq 0$$

$$x_p \in \{0,1\}^n.$$

Definition: (Restricted) Boolean Quadric Polytope (BQP) of size n

$$BQP_n = Conv\left\{X \in \mathbb{R}^{n \times n} \,\middle|\, X = xx^\top,\ x \in \{0,1\}^n\right\}$$

Some relations

The CP and PSD cones:

We recall that the Completely Positive (CP) and the Positive Semi Definite (PSD) cones are respectively:

$$CP_n = Conv\left\{X \in \mathbb{R}^{n\times n} \,|\, X = xx^\top,\ x \in \mathbb{R}^n,\ x \geq 0\right\}$$
$$PSD_n = Conv\left\{X \in \mathbb{R}^{n\times n} \,|\, X = xx^\top,\ x \in \mathbb{R}^n\right\}$$

Then,

$$BQP_n \subset CP_n \subset PSD_n.$$

Lower bounds

Hence the lower bound (LB) obtained with our relaxation is stronger than the CP and PSD bounds:

$$LB_{BQP} \geq LB_{CP} \geq LB_{PSD}.$$

A matrix generation algorithm

Let $\mathcal{P} := \{1, \dots, 2^n\}$. Then, we have:

Formulation

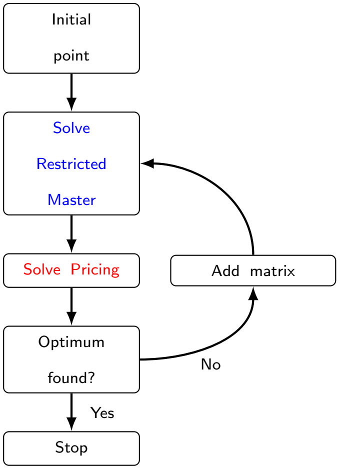
$$
\begin{aligned}
& \min \ \langle Q, X \rangle \\
(1) \quad & \text{s. t.} \ \langle A_i, X \rangle \leq b_i, \quad \forall i = 1 \dots, m \\
& X = \sum_{p \in \mathcal{P}} \bar{X}_p \lambda_p \\
& \sum_{p \in \mathcal{P}} \lambda_p = 1 \\
& \lambda_p \geq 0 \quad \forall p \in \mathcal{P} \\
& \bar{X}_p = \bar{x}_p \bar{x}_p^\top \quad \forall p \in \mathcal{P} \\
& \bar{x}_p \in \{0, 1\}^n \quad \forall p \in \mathcal{P}.
\end{aligned}
$$

A matrix generation algorithm

Let $\mathcal{P} := \{1, \ldots, 2^n\}$. Then, we have:

Formulation

$$
(1) \quad \begin{aligned}
\min\ & \langle Q, X \rangle \\
\text{s. t. } & \langle A_i, X \rangle \leq b_i, \quad \forall i = 1 \ldots, m \\
& X = \sum_{p \in \mathcal{P}} \bar{X}_p \lambda_p \\
& \sum_{p \in \mathcal{P}} \lambda_p = 1 \\
& \lambda_p \geq 0 \quad \forall p \in \mathcal{P} \\
& \bar{X}_p = \bar{x}_p \bar{x}_p^\top \quad \forall p \in \mathcal{P} \\
& \bar{x}_p \in \{0, 1\}^n \quad \forall p \in \mathcal{P}.
\end{aligned}
$$

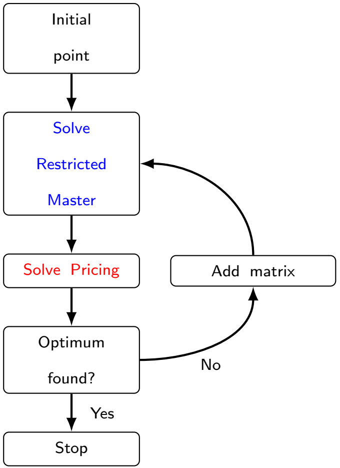


A matrix generation algorithm

Let $\mathcal{P} := \{1, \ldots, 2^n\}$. Then, we have:

Formulation

$$
\begin{aligned}
& \min \ \langle Q, X \rangle \\
(1) \quad & \text{s. t. } \langle A_i, X \rangle \leq b_i, \quad \forall i = 1 \ldots, m \\
& X = \sum_{p \in \mathcal{P}} \bar{X}_p \lambda_p \\
& \sum_{p \in \mathcal{P}} \lambda_p = 1 \\
& \lambda_p \geq 0 \quad \forall p \in \mathcal{P} \\
& \bar{X}_p = \bar{x}_p \bar{x}_p^\top \quad \forall p \in \mathcal{P} \\
& \bar{x}_p \in \{0, 1\}^n \quad \forall p \in \mathcal{P}.
\end{aligned}
$$



Master and Pricing problems

Let $\bar{\mathcal{P}} \subset \mathcal{P}$, $\quad \bar{X}_p := x_p x_p^\top,\ p \in \bar{\mathcal{P}}$

Restricted Master Problem (RMP)

$$
\begin{aligned}
\min\ & \langle Q, X \rangle \\
\text{s. t.}\quad & \langle A_i, X \rangle \leq b_i, \quad \forall i = 1 \ldots, m \\
& X = \sum_{p \in \bar{\mathcal{P}}} \bar{X}_p \lambda_p \\
& \sum_{p \in \bar{\mathcal{P}}} \lambda_p = 1 \\
& \lambda_p \geq 0 \quad \forall p \in \bar{\mathcal{P}}.
\end{aligned}
$$

Master and Pricing problems

RMP,reduced form

$$
\begin{aligned}
\min \quad & \sum_{p \in \bar{\mathcal{P}}} \langle Q, X_p \rangle \lambda_p \\
\text{s. t.} \quad & \sum_{p \in \bar{\mathcal{P}}} \langle A_i, X_p \rangle \lambda_p \le b_i, \quad \forall i = 1 \dots, m & [\pi] \\
& \sum_{p \in \bar{\mathcal{P}}} \lambda_p = 1 & [\pi_0] \\
& \lambda_p \ge 0 \quad \forall p \in \bar{\mathcal{P}}.
\end{aligned}
$$

Master and Pricing problems

Dual problem

$$
\begin{aligned}
\max\ & b^\top \pi + \pi_0 \\
\text{s. t.}\ & \sum_{i=1}^{m} \langle A_i, \bar{X}_p \rangle \pi_i + \pi_0 \leq \langle Q, \bar{X}_p \rangle, \quad \forall p \in \bar{\mathcal{P}} \\
& \pi \leq 0
\end{aligned}
$$

Pricing problem

$$
\begin{aligned}
\min\ & \langle Q, X \rangle - \sum_{i=1}^{m} \langle A_i, X \rangle \pi_i^* - \pi_0^* \\
\text{s. t.}\ & X = xx^\top \\
& x \in \{0,1\}^n
\end{aligned}
$$

Instance	BC-bound		BC-cuts		BQP	
	T (s)	Gap (%)	T (s)	Gap (%)	T (s)	Gap (%)
QPLIB-0067	0	5	22	2	0	1
QPLIB-1976	27	433	193	371	7	368
QPLIB-2017	113	441	114	441	124	240
QPLIB-2029	180	562	180	562	1865	192
QPLIB-2036	220	740	220	740	185	313
QPLIB-2055	21	41	104	35	92	32
QPLIB-2060	36	42	655	33	153	32
QPLIB-2067	72	68	149	65	242	62
QPLIB-2073	57	18	1078	10	285	10
QPLIB-2085	85	33	2642	23	1066	23
QPLIB-2087	123	71	172	71	2935	56
QPLIB-2096	82	18	2679	11	1210	11
QPLIB-2357	16	13	46	0	3223	0
QPLIB-2359	74	11	54	2	2888	0
QPLIB-2512	2	428	117	120	6	100
QPLIB-2733	10	762	1006	178	19258	154
QPLIB-2957	78	> 1000	2392	357	11261	100
QPLIB-3307	5	798	1044	198	472	100
QPLIB-3413	33	> 1000	678	210	11	100
QPLIB-3587	5	791	109	104	2	100
QPLIB-3614	4	680	123	100	2	100
QPLIB-3714	2	101	20	0	1607	0
QPLIB-3751	2	100	20	0	7709	0
QPLIB-3757	482	18	344	37	8850	0
QPLIB-3762	1	17	2	0	1037	0
QPLIB-3775	5	100	25	0	32932	0
QPLIB-3803	12	33	57	0	5620	0
QPLIB-3815	2	29	41	6	1807	2
QPLIB-6647	1009	> 1000	11271	150	232	100
QPLIB-7127	646	> 1000	1662	> 1000	2750	0
average	95.1	> 1000	881.28	132	3594.4	73

Instance	BQP		Cplex-r.n.		Cplex-bd
	T (s)	Gap (%)	T (s)	Gap (%)	T (s)
QPLIB-0067	0.2	1	0.5	1	0.3
QPLIB-1976	6.8	368	0.3	447	32.5
QPLIB-2017	124.4	240	0.4	262	20.8
QPLIB-2029	1865.5	192	0.4	207	20.4
QPLIB-2036	185.1	313	0.8	351	135.3
QPLIB-2055	92.4	32	1.2	41	24.0
QPLIB-2060	153.0	32	1.7	42	41.0
QPLIB-2067	242.2	62	1.8	69	24.7
QPLIB-2073	285.4	10	3.0	20	*36000.0
QPLIB-2085	1065.5	23	4.3	33	3403.0
QPLIB-2087	2935.3	56	5.4	68	60.4
QPLIB-2096	1209.6	11	6.1	20	*36000.0
QPLIB-2357	3222.8	0	0.2	68	528.9
QPLIB-2359	2888.2	0	0.2	43	258.3
QPLIB-2512	6.3	100	0.1	100	0.0
QPLIB-2733	19258.1	154	1.1	100	0.1
QPLIB-2957	11261.0	100	1.9	100	0.2
QPLIB-3307	472.4	100	2.0	100	0.1
QPLIB-3413	10.5	100	0.3	100	0.0
QPLIB-3587	2.4	100	0.1	100	0.0
QPLIB-3614	1.9	100	0.1	100	0.0
QPLIB-3714	1607.0	0	0.0	100	133.0
QPLIB-3751	7708.6	0	0.1	100	579.7
QPLIB-3757	8850.0	0	0.7	31	421.3
QPLIB-3762	1036.8	0	0.1	98	84.1
QPLIB-3775	32931.6	0	0.1	100	4600.2
QPLIB-3803	5620.1	0	0.2	163	814.3
QPLIB-3815	1807.3	1	0.0	48	24.7
QPLIB-6647	232.2	100	0.4	100	0.0
QPLIB-7127	2750.3	0	1.6	0	0.1
average	3594.4	73	1.2	104	> 2773.6

CP reformulation for Binary Quadratic Problems

The following problems are equivalent (Burer, 2009):

Binary Quadratic Problems

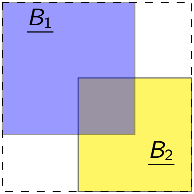
$$\begin{aligned} \min \quad & x^\top Q x + q^\top x \\ \text{s. t.} \quad & a_i^\top x = b_i, \quad \forall i = 1 \dots, m \\ & x \in \{0,1\}^n. \end{aligned}$$

Completely Positive reformulation

$$\begin{aligned} & \min \langle Q, \bar{X} \rangle + q_0^\top x \\ \text{s. t.} \quad & a_i^\top x = b_i, \quad \forall i = 1 \dots, m \\ & a_i^\top \bar{X} a_i = b_i^2, \quad \forall i = 1 \dots, m \\ & x_j = \bar{X}_{jj} \quad \forall j = 1, \dots n \\ & \begin{pmatrix} 1 & x^\top \\ x & \bar{X} \end{pmatrix} \in CP_{n+1}. \end{aligned}$$

Hence, we have an exact reformulation for binary QPs with linear equalities (no branching needed).

Notations



Let:

- $\underline{b_j} \subset \{1, \dots, n\}\ \forall j = 1, \dots, k$;
- $\bigcup_{j=1}^{k} \underline{b_j} = \{1, \dots, n\}$;
- $\underline{B_j} = \underline{b_j} \times \underline{b_j}\ \forall j = 1, \dots, k$.

A Block structure is $\underline{\mathcal{B}_k} = \bigcup_{j=1}^{k} \underline{B_j}$.

Block decomposable problems

A problem is block decomposable if all nonzero entries of Q, A_i belong to $\underline{\mathcal{B}_k}$.

$\forall X \in \mathbb{R}^{n \times n}$ we indicate with

$$X^{B_j} := \{X_{p,q} | p, q \in \underline{b_j}\} \in \mathbb{R}^{d_j \times d_j} \quad \forall j = 1, \dots, k$$

the restriction of X to a block j; d_j is the dimension of $\underline{b_j}$.
$\forall j = 1, \dots k$, for $M = Q, A_i\ \forall i = 1, \dots m$ let $M^j := M^{B_j}$, with
$M_{p,q} = 0\ \forall p, q \in \underline{b_j} \setminus (\underline{b_1}, \dots, \underline{b_{j-1}})$.

Block decomposition

Block-decomposed Master Program formulation

$$
(2) \quad \begin{aligned}
\min \quad & \sum_{j=1}^{k} \langle Q^j, Y_j \rangle \\
\text{s. t.} \quad & \sum_{j=1}^{k} \langle A_i^j, Y_j \rangle \leq b_i, \quad \forall i = 1 \ldots, m \\
& Y_j^{B_j \cap B_h} = Y_h^{B_j \cap B_h} \quad \forall 1 \leq j < h \leq k \\
& Y_j = \sum_{l=1}^{2^{d_j}} \mu_l^j (y_j^l)(y_j^l)^\top \quad \forall j = 1, \ldots, k \\
& \sum_{l=1}^{2^{d_i}} \mu_l^j = 1 \quad \forall j = 1, \ldots, k \\
& \mu_l^j \geq 0 \quad \forall l = 1, \ldots 2^{d_j}, \ \forall j = 1, \ldots, k. \\
& y_j^l \in \{0, 1\}^{d_j} \quad \forall l = 1, \ldots, 2^{d_j} \ \forall j = 1, \ldots, k.
\end{aligned}
$$

Block-decomposed restricted master and pricing

Let $\bar{\mathcal{P}}_j \subseteq \{1, \dots, 2^{d_j}\}$ $\forall j = 1, \dots, k$. Then:

Block-decomposed RMP

$$
\begin{aligned}
\min \ & \sum_{j=1}^{k} \langle Q^j, Y_j \rangle \\
\text{s. t.} \ & \sum_{j=1}^{k} \langle A_i^j, Y_j \rangle \le b_i, \quad \forall i = 1 \dots, m \qquad [\alpha] \\
& Y_j^{B_j \cap B_h} = Y_h^{B_j \cap B_h} \quad \forall 1 \le j < h \le k \quad [\beta^{j,h}] \\
& Y_j = \sum_{l \in \bar{\mathcal{P}}_j} \mu_l^j (y_j^l)(y_j^l)^\top \quad \forall j = 1, \dots, k \quad [\pi^j] \\
& \sum_{l \in \bar{\mathcal{P}}_j} \mu_l^j = 1 \quad \forall j = 1, \dots, k \qquad [\pi_0^j] \\
& \mu_l^j \ge 0 \quad \forall l \in \bar{\mathcal{P}}_j, \ \forall j = 1, \dots, k.
\end{aligned}
$$

Block-decomposed restricted master and pricing

Dual problem

$$
\begin{aligned}
\max\ & b^\top \alpha + \sum_{j=1}^{k} \pi_0^j \\
\text{s. t. } & \sum_{i=1}^{m} A_i^j \alpha_i + \sum_{h=1, h>j}^{k} C^{j,h} \beta^{j,h} - \sum_{h=1, h<j}^{k} C^{j,h} \beta^{j,h} + \pi^j = Q^j \quad \forall j = 1, \dots, k \\
& - \langle (y_j^l)(y_j^l)^\top, \pi^j \rangle + \pi_0^j \le 0, \quad \forall l \in \bar{\mathcal{P}} \\
& \alpha \le 0,
\end{aligned}
$$

where $(C^{j,h})_{p,q} = 1$ if $(p, q) \in \underline{B}_j \cap \underline{B}_h$, 0 otherwise.

Pricing problems

$$
\begin{aligned}
\min\ & \langle {\pi^j}^*, Y_j \rangle - {\pi_0^j}^* \\
\text{s. t. } & Y_j = y_j y_j^\top \\
& y_j \in \{0, 1\}^{d_j} \quad \forall j = 1, \dots, k.
\end{aligned}
$$

Equivalence problem between (1) and (2)

First inclusion "$\supseteq$"

If X, λ are feasible for (1), $\exists$ Y_j, μ_j^I feasible for (2), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

Yes, always. Hence (2) always gives a valid lower bound.

Second inclusion "$\subseteq$"

If Y_j, μ_j^I are feasible for (2), $\exists$ X, λ feasible for (1), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

It depends on the block structure.

Equivalence problem between (1) and (2)

First inclusion "$\supseteq$"

If X, λ are feasible for (1), $\exists$ Y_j, μ_j^l feasible for (2), s.t. $X^{B_j} = Y_j \; \forall j = 1, \dots, k$?

Answer

Yes, always. Hence (2) always gives a valid lower bound.

Second inclusion "$\subseteq$"

If Y_j, μ_j^l are feasible for (2), $\exists$ X, λ feasible for (1), s.t. $X^{B_j} = Y_j \; \forall j = 1, \dots, k$?

Answer

It depends on the block structure.

Equivalence problem between (1) and (2)

First inclusion "$\supseteq$"

If X, λ are feasible for (1), $\exists$ Y_j, μ_j^l feasible for (2), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

Yes, always. Hence (2) always gives a valid lower bound.

Second inclusion "$\subseteq$"

If Y_j, μ_j^l are feasible for (2), $\exists$ X, λ feasible for (1), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

It depends on the block structure.

Equivalence problem between (1) and (2)

First inclusion "$\supseteq$"

If X, λ are feasible for (1), $\exists$ Y_j, μ_j^l feasible for (2), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

Yes, always. Hence (2) always gives a valid lower bound.

Second inclusion "$\subseteq$"

If Y_j, μ_j^l are feasible for (2), $\exists$ X, λ feasible for (1), s.t. $X^{B_j} = Y_j \; \forall j = 1, \ldots, k$?

Answer

It depends on the block structure.

Counterexample

Let Q, $A_i \in \mathbb{R}^{4\times4}$ have the following block structure:

$$\underline{b_1} = \{1,2\}, \underline{b_2} = \{2,3\}, \underline{b_3} = \{3,4\}, \underline{b_4} = \{1,4\}.$$

Then:

A feasible solution for (2)

given by:

$$\frac{1}{2}\begin{pmatrix} 1 & 1 & & 0 \\ 1 & 1 & 1 & \\ & 1 & 1 & 1 \\ 0 & & 1 & 1 \end{pmatrix},$$

$$Y_j = \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \quad j = 1,2,3$$

$$Y_4 = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix},$$

it cannot be *completed* to a solution to (1).

BQP completion problem

Definitions

Let $M \in \mathbb{R}^{n \times n}$, symmetric.

- M is partial if some entries are not specified;
- a specification graph of M has n vertices and edges $\{i,j\}$ if $M_{i,j}$ is *specified*;
- M is partial BQP if $\forall$ fully specified principal submatrix N, $N \in BQP$;
- if M is *partial BQP*, it is BQP completable if $\exists N \in BQP_n$ fully specified, $N_{i,j} = M_{i,j}$ where $M_{i,j}$ is specified;
- a graph G is BQP completable if *every* partial BQP matrix with specification graph G is BQP completable.

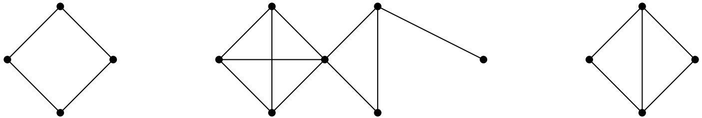
BQP completion problem $\leftrightarrow$ "$\subseteq$" inclusion:

Which graphs are BQP completable?

Theoretical results

Known results: PSD and CP completion problems

- A graph is PSD-completable iff it is chordal;
- A graph is CP-completable iff it is block-clique.



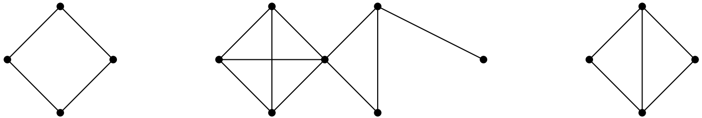
BQP completion problem

- If G is not chordal, it is not BQP-completable;
- If G is chordal, is it BQP-completable?
  - if the max size d of intersections is 2: yes (diamond graph);
  - if $d > 2$: conjecture.

Theoretical results

Known results: PSD and CP completion problems

- A graph is PSD-completable iff it is chordal;
- A graph is CP-completable iff it is block-clique.



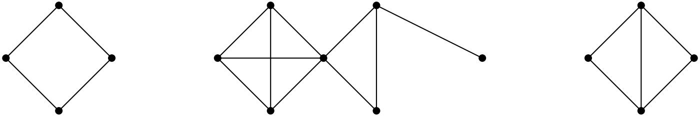
BQP completion problem

- If G is not chordal, it is not BQP-completable;
- If G is chordal, is it BQP-completable?
  - if the max size d of intersections is 2: yes (diamond graph);
  - if $d > 2$: conjecture.

Theoretical results

Known results: PSD and CP completion problems

- A graph is PSD-completable iff it is chordal;
- A graph is CP-completable iff it is block-clique.



BQP completion problem

- If G is not chordal, it is not BQP-completable;
- If G is chordal, is it BQP-completable?
  - if the max size d of intersections is 2: yes (diamond graph);
  - if $d > 2$: conjecture.

Results

Instance		BC-cuts		BQP				
				Single block		Multiple block		
Name	#	T (s)	Gap (%)	# Fails	T (s)	# Fails	T (s)	Gap (%)
ins.16	5	89	186	0	304	0	0.26	85
ins.17	5	171	201	1	7210	0	0.28	64
ins.18	5	74	232	0	1402	0	0.46	89
ins.19	5	53	280	2	15215	0	0.44	88
ins.20	5	67	175	4	28900	0	0.42	84
ins.21	5	86	348	4	28826	0	0.7	70
ins.22	5	109	396	2	14645	0	0.92	61
ins.23	3	139	330	2	32570	0	0.43	82
ins.24	5	171	> 1000	3	22156	0	1.34	75
ins.25	5	204	417	2	14542	0	1.52	55
average	5	115	394	2	16577	0	0.68	75

Table: Performance comparison on the SONET instances.

Results

Instance		BQP		Cplex-r.n.		Cplex-bd
	#	T (s)	Gap (%)	T (s)	Gap (%)	T (s)
ins.16	5	0.26	85	0.26	98	5.5
ins.17	5	0.28	64	0.22	96	21.1
ins.18	5	0.46	89	0.20	108	5.0
ins.19	5	0.44	88	0.23	99	4.8
ins.20	5	0.42	84	0.53	98	58.2
ins.21	5	0.7	70	0.74	87	22.5
ins.22	5	0.92	61	0.74	87	32.3
ins.23	3	0.43	82	0.48	88	7.8
ins.24	5	1.34	75	0.54	102	12.7
ins.25	5	1.52	55	1.09	82	56.2
average	5	0.68	75	0.50	95	23.2

Table: Comparisons with Cplex, SONET instances.

Results

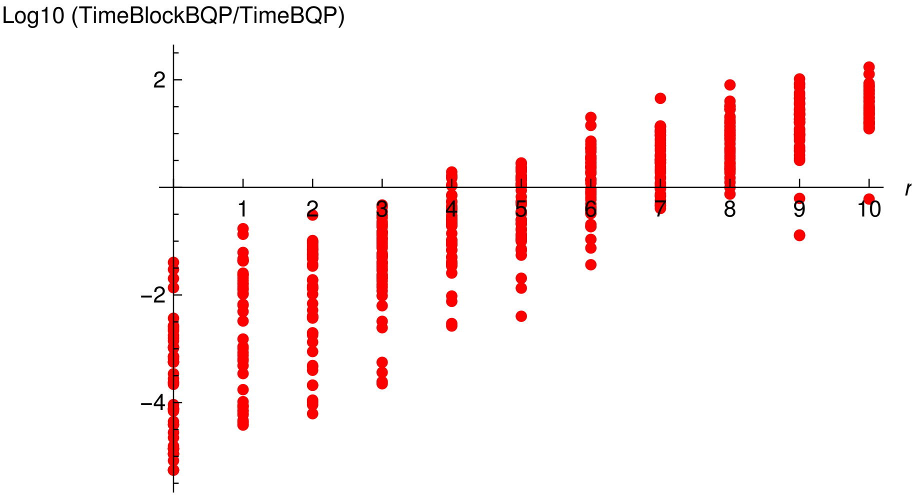


Figure: Performance of block BQP on problems with different intersection size.

Conclusions

Final considerations

- Simplicial Decomposition can be very effective for continuous and 0-1 convex quadratic problems.
- Dantzig-Wolfe reformulation and convexification approaches can be used together.
- Dantzig-Wolfe reformulation is crucial to obtain strong dual bounds (BQP bounds vs CP bounds vs SDP bounds).

Perspectives

- Derive a Branch & Price approach combining Dantzig-Wolfe reformulation and convexification.
- Combine Simplicial and Dantzig-Wolfe Decompositions.
- Derive a Branch & Price approach using the Boolean Quadric Polytope relaxation.
- Combine Decomposition Methods and Quantum Optimization.

Thank you for your attention!