Two-stage Stochastic Standard Quadratic Optimization

Markus Gabl[1] Immanuel Bomze[2] Francesca Maggioni [3]
Georg Pflug [2]

[1]Karlsruhe Institute of Technology, Karlsruhe,

[2]University of Vienna, Vienna, Austria

[3]Università degli Studi di Bergamo, Italy

May 21, 2022

FWF
Der Wissenschaftsfonds.

VGSCO
Vienna Graduate School on Computational Optimization

Overview

Contents

The Standard Quadratic Problem

Definition

The (deterministic) **Standard Quadratic Problem** (DStQP) is to optimize a quadratic form over the Standard Simplex (Probability Simplex)

$$(\mathrm{DStQP}) \qquad z^*_{\mathrm{det}} := \min \left\{ q(\mathsf{z}) := \mathsf{z}^\mathsf{T} \mathsf{Q} \mathsf{z} : \mathsf{z} \in \Delta^n \right\}$$

where $\Delta^n = \left\{ \mathsf{z} \in \mathbb{R}^n_+ : \bar{\mathsf{e}}^\mathsf{T} \mathsf{z} = 1 \right\}$ and $\bar{\mathsf{e}}$ is the vector of all ones.

The DStqP arises in many different contexts such as

- the Maximum-Clique- and the Maximum-Weight-Clique-Problem,
- Evolutionary Game Theory,
- Portfolio Selection and
- Dominant-Set Clustering

The Standard Quadratic Problem

Definition

The (deterministic) **Standard Quadratic Problem** (DStQP) is to optimize a quadratic form over the Standard Simplex (Probability Simplex)

$$(\mathrm{DStQP}) \qquad z^*_{\mathrm{det}} := \min \left\{ q(\mathsf{z}) := \mathsf{z}^{\mathsf{T}} \mathsf{Q} \mathsf{z} : \mathsf{z} \in \Delta^n \right\}$$

where $\Delta^n = \left\{ \mathsf{z} \in \mathbb{R}^n_+ : \bar{\mathsf{e}}^{\mathsf{T}} \mathsf{z} = 1 \right\}$ and $\bar{\mathsf{e}}$ is the vector of all ones.

- DStqP defines a class of NP-Hard optimization problems.
- Many polynomial time approximation schemes.
- Well performing local algorithms (e.g. based on immunization-infection dynamics).
- Many nice analytic bounds.

Contents

Introducing Uncertainty

Uncertain blocks in the objective matrix

- Henceforth, we assume that the quadratic for has uncertain entries:

$$\tilde{Q} := \begin{bmatrix} A & \tilde{B}^T \\ \tilde{B} & \tilde{C} \end{bmatrix}.$$

- Only the matrix $A \in \mathbb{R}^{n_1 \times n_1}$ is known exactly.
- The matrices $\tilde{B} \in \mathbb{R}^{n_2 \times n_1}$, $\tilde{C} \in \mathbb{R}^{n_2 \times n_2}$ are only known to follow a certain, known probability distribution $[\tilde{B}, \tilde{C}] =: \boldsymbol{\xi} \sim \mathcal{P}$.

Two-Stage Decision Process

- We split the decision vector in two parts $z = (x, y(\boldsymbol{\xi}))$.
- x is the here and now decision, to be made immediately
- $y(\boldsymbol{\xi})$ is the second stage decision, that may adapt to the uncertain outcome.

Problem Formulation

Definitions

Decomposing $z := (x, y(\xi))$,we arrive at

$$q\left(x, y(\xi)\right) = x^T A x + 2x^T \tilde{B}^T y(\xi) + y(\xi)^T \tilde{C} y(\xi)$$

The two stage Stochastic version of the StQP is then given by

$$\min_{x \in T^{n_1}} \left\{ x^T A x + \mathbb{E}_{\boldsymbol{\xi}} \left[\min_{y(\boldsymbol{\xi}) \geq 0} \left\{ 2x^T \tilde{B}^T y(\xi) + y(\xi)^T \tilde{C} y(\xi) \colon e^T x + e^T y(\xi) = 1 \right\} \right] \right\} \tag{1}$$

$\Rightarrow$ We want to find a first stage decision, that optimizes the expected performance of our second stage decision.

The Scenario Problem

- We can approximate the true probability distribution with an discretization.
- We get a discrete probability measure $p_1, \ldots, p_S$,
- associated realizations of the random data $\mathsf{B}_1, \ldots, \mathsf{B}_S$ and $\mathsf{C}_1, \ldots, \mathsf{C}_S$

The problem then reduces to the so called *scenario problem*:

$$\begin{aligned} \min_{\mathsf{x},\mathsf{y}_1,\ldots,\mathsf{y}_S} \; & \mathsf{x}^\mathsf{T}\mathsf{A}\mathsf{x} + \sum_{i=1}^{S} p_s \left(2\mathsf{y}_s^\mathsf{T}\mathsf{B}_s\mathsf{x} + \mathsf{y}_s^\mathsf{T}\mathsf{C}_s\mathsf{y}_s\right) \\ \text{s.t.} : \mathsf{e}^\mathsf{T}\mathsf{x} + \mathsf{e}^\mathsf{T}\mathsf{y}_s &= 1, \qquad s = 1, \ldots, S, \\ \mathsf{x} &\geq 0, \\ \mathsf{y}_s &\geq 0, \qquad s = 1, \ldots, S, \end{aligned}$$

Non-convex QP with linear constraints!

Contents

Wait-and-See Approach

Suppose the following:

- ξ has a finite support $\Xi = \{\xi_1, \ldots, \xi_S\}$,
- Finite number of possible scenarios $\xi_s = (\mathsf{B}_s, \mathsf{C}_s)$.
- Positive probabilities p_s, $s = 1, \ldots, S$.

Deterministic optimization problem under scenario s:

$$z_s^* := \min \left\{ q_s(\mathsf{z}) := \mathsf{z}^\mathsf{T} \mathsf{Q}_s \mathsf{z} : \mathsf{z} \in \Delta^n \right\},$$

Wait and See Solution (WS)

$$z^{1^*} := \sum_{s=1}^{S} p_s z_s^* \leq z_{\text{stoch}}^*$$

Contents

A Chain of Lower Bounds

Refinement of the Probability Measure

Consider the following **successively refining partition of the sample space**:

$$\Xi = \{\xi_1, \dots, \xi_S\}$$
$$\dots$$
$$(\Xi_1^{(j)}, \Xi_2^{(j)}, \dots, \Xi_{m_j}^{(j)})$$
$$\dots$$
$$(\Xi_1^{(2)}, \Xi_2^{(2)}, \dots, \Xi_{m_2}^{(2)})$$
$$(\{\xi_1\}, \{\xi_2\}, \dots, \{\xi_S\}),$$

- Each row is a collection of subsets of the probability space Ξ.
- The whole space $\Xi = \cup_i \Xi_i^{(j)}$ for all j.
- $\Xi_i^{(j)}$ is the union of sets from the next more refined collection

A Hierarchy of Lower Bounds

A Chain of Lower Bounds

Denoting with

$$z^{j*} = \sum_{i=1}^{m_j} \pi_i^{(j)} z^*(\Xi_i^{(j)}),$$

where $\pi_i^{(j)} = \sum_{\xi_s \in \Xi_i^{(j)}} p_s$, we get to a chain of lower bounds expressed as follows

$$z^{1*} \le z^{2*} \le \cdots \le z^{j*} \le \cdots \le z^*_{\text{stoch}}. \tag{2}$$

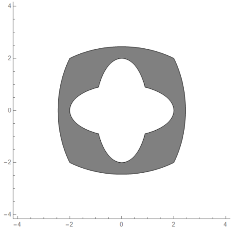
- The higher the index j, the fewer problems have to be solved,
- but with an increasing number of scenarios.

Contents

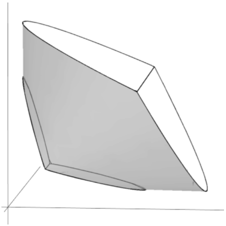
Convex Reformulations of QCQPs in a Nutshell

$$\min_{x\in\mathbb{R}^2} q_{11}x_1^2 + 2q_{12}x_1x_2 + q_{22}x_2^2$$
$$\text{s.t.}:\ 2x_1^2 + x_2^2 \leq 12,$$
$$x_1^2 + 2x_2^2 \leq 12,$$
$$4x_1^2 + x_2^2 \geq 4,$$
$$x_1^2 + 4x_2^2 \geq 4,$$

$$\min_{X\in\mathcal{S}_+^2} q_{11}X_{11} + 2q_{12}X_{21} + q_{22}X_{22}$$
$$\text{s.t.}:\ 2X_{11} + X_{22} \leq 12,$$
$$X_{11} + 2X_{22} \leq 12,$$
$$4X_{11} + X_{22} \geq 4,$$
$$X_{11} + 4X_{22} \geq 4,$$

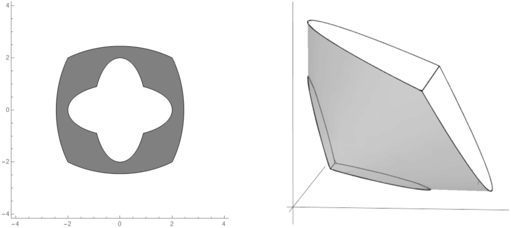


$\mathcal{F}$



$\mathcal{G}(\mathcal{F})$

Convex Reformulations of QCQPs in a Nutshell



$\mathcal{F}$ $\qquad\qquad$ $\mathcal{G}(\mathcal{F})$

- We can lift the space of variables by replacing $xx^T \to X$
- Then all quadratic expression become $x^T A x = \mathrm{tr}(Axx^T) \to \mathrm{tr}(AX) = A \bullet X$, hence linear.
- Since not all psd-matrices are of the form xx^T the lifted problem is a relaxation.
- If the extreme points of the lifted feasible set matrices of the form xx^T, we get a feasible solution for the original problem.
- Thus, in these cases the relaxation is tight!

Contents

Lower Bounds Based on Copositive Optimization

Exact CPP-Reformulation

Burer (2009) proofs the tightness of the CPP-relaxation

$$\min_{x, y_1, \dots, y_S} A \bullet X + \sum_{i=1}^{S} p_s \left(2B_s \bullet Z_s +^{\mathsf{T}} C_s \bullet Y_s \right)$$

$$\text{s.t.} : e^{\mathsf{T}} x + e^{\mathsf{T}} y_s = 1, \qquad s = 1, \dots, S,$$

$$E \bullet X + E \bullet Y_s + 2E \bullet Z_s = 1, \qquad s = 1, \dots, S,$$

$$\begin{pmatrix} 1 & x^{\mathsf{T}} & y_1^{\mathsf{T}} & \dots & y_S^{\mathsf{T}} \\ x & X & Z_1^{\mathsf{T}} & \dots & Z_S^{\mathsf{T}} \\ y_1 & Z_1 & Y_1 & \dots & Y_{1,S}^{\mathsf{T}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_S & Z_S & Y_{1,S} & \dots & Y_S \end{pmatrix} \in \mathcal{CPP}\left(\mathbb{R}_+^{n_1 + S n_2 + 1} \right) := \left\{ BB^{\mathsf{T}} : B \geq 0 \right\} \tag{CPP1}$$

[a] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. Mathematical Programming, 120(2) 2009

Lower Bounds Based on Copositive Optimization

DNN Relaxation

We can further relax the problem in order to obtain a **tractable lower bound**

$$\min_{\mathsf{x},\mathsf{y}_1,\dots,\mathsf{y}_S} \mathsf{A} \bullet \mathsf{X} + \sum_{i=1}^{S} p_s \left(2\mathsf{B}_s \bullet \mathsf{Z}_s +^{\mathsf{T}} \mathsf{C}_s \bullet \mathsf{Y}_s\right)$$

$$\text{s.t.}: \mathsf{e}^{\mathsf{T}}\mathsf{x} + \mathsf{e}^{\mathsf{T}}\mathsf{y}_s = 1, \qquad s = 1, \dots, S,$$

$$\mathsf{E} \bullet \mathsf{X} + \mathsf{E} \bullet \mathsf{Y}_s + 2\mathsf{E} \bullet \mathsf{Z}_s = 1, \qquad s = 1, \dots, S,$$

$$\begin{pmatrix} 1 & \mathsf{x}^{\mathsf{T}} & \mathsf{y}_1^{\mathsf{T}} & \dots & \mathsf{y}_S^{\mathsf{T}} \\ \mathsf{x} & \mathsf{X} & \mathsf{Z}_1^{\mathsf{T}} & \dots & \mathsf{Z}_S^{\mathsf{T}} \\ \mathsf{y}_1 & \mathsf{Z}_1 & \mathsf{Y}_1 & \dots & \mathsf{Y}_{1,S}^{\mathsf{T}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathsf{y}_S & \mathsf{Z}_S & \mathsf{Y}_{1,S} & \dots & \mathsf{Y}_S \end{pmatrix} \in \mathcal{DNN}^{n_1+Sn_2+1} := \mathcal{S}_+^{n_1+Sn_2+1} \cap \mathcal{N}^{n_1+Sn_2+1} \tag{DNN1}$$

Lower Bounds Based on Copositive Optimization

Cheaper CPP-Relaxation

A **cheaper lower bound** is given by the problem

$$
\begin{aligned}
\min_{\mathsf{x},\mathsf{y}_1,\dots,\mathsf{y}_S} \ & \mathsf{A} \bullet \mathsf{X} + \sum_{i=1}^{S} p_s \left(2\mathsf{B}_s \bullet \mathsf{Z}_s + \mathsf{C}_s \bullet \mathsf{Y}_s\right) \\
\text{s.t.}: \ & \mathsf{e}^{\mathsf{T}}\mathsf{x} + \mathsf{e}^{\mathsf{T}}\mathsf{y}_s = 1, && s = 1,\dots,S, \\
\mathsf{E} \bullet \mathsf{X} + \mathsf{E} \bullet \mathsf{Y}_s + 2\mathsf{E} \bullet \mathsf{Z}_s &= 1, && s = 1,\dots,S, \\
& \begin{pmatrix} 1 & \mathsf{x}^{\mathsf{T}} & \mathsf{y}_s^{\mathsf{T}} \\ \mathsf{x} & \mathsf{X} & \mathsf{Z}_s^{\mathsf{T}} \\ \mathsf{y} & \mathsf{Z}_s & \mathsf{Y}_s \end{pmatrix} \in \mathcal{CPP}\left(\mathbb{R}_+^{n_1+n_2+1}\right) && s = 1,\dots,S,
\end{aligned}
\tag{CPP2}
$$

Lower Bounds Based on Copositive Optimization

Cheaper DNN Relaxation

We can relax the problem in the same way as before:

$$
\begin{aligned}
\min_{x,y_1,\dots,y_S} \; & A \bullet X + \sum_{i=1}^{S} p_s \left(2B_s \bullet Z_s +^{\mathsf{T}} C_s \bullet Y_s \right) \\
\text{s.t.}: \; & e^{\mathsf{T}} x + e^{\mathsf{T}} y_s = 1, && s = 1, \dots, S, \\
& E \bullet X + E \bullet Y_s + 2E \bullet Z_s = 1, && s = 1, \dots, S, \\
& \begin{pmatrix} 1 & x^{\mathsf{T}} & y_s^{\mathsf{T}} \\ x & X & Z_s^{\mathsf{T}} \\ y & Z_s & Y_s \end{pmatrix} \in \mathcal{S}_+^{n_1+n_2+1} \cap \mathcal{N}^{n_1+n_2+1} && s = 1, \dots, S,
\end{aligned}
\tag{DNN2}
$$

Contents

The Franke-Wolfe Algorithm in a Nutshell

Algorithm 0: A fast way to solve $\min_{x \in P} f(x)$, for a polytope P.

Result: v^*

set $k = 1$

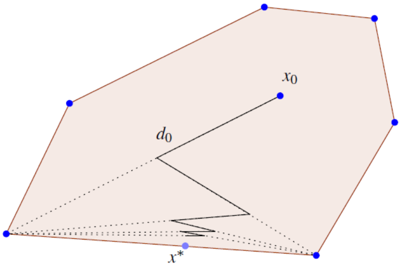
set $x_1 \in P$

repeat

  solve $\min_{y_+ \in P} \nabla f(x_k)^T y_+$

  move to best vertex i.e. : $x_{k+1} = x_k + \alpha(y_+ - x_k)$

until *Some criterium is met*



The Pairwise Franke-Wolfe Algorithm in a Nutshell

Algorithm 1: An even faster way to solve $\min_{x \in P} f(x)$, for a polytope P.

Result: v^*

set $k = 1$

set $x_1 \in P$

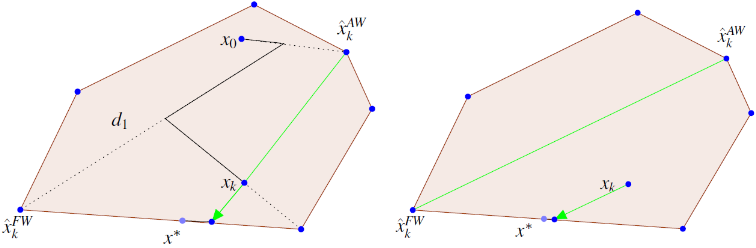
choose $S_k \subseteq P$

repeat

solve $\min_{y_+ \in P} \nabla f(x_k)^\top y_+$ and $\max_{y_- \in S_k} \nabla f(x_k)^\top y_-$; move towards best and away from worst vertex i.e. : $x_{k+1} = x_k + \alpha(y_+ - y_-)$

Update S_k

until *Some criterium is met*



Adapting AS-FW for the scenario problem

- The gradient is easily calculated from the problem data since $\nabla \mathrm{x}^{\mathsf{T}}\mathrm{Q}\mathrm{x} = 2\mathrm{Q}\mathrm{x}$.
- The vertices of the feasible set of $\left\{(\mathrm{x}, \mathrm{y}_1, \dots, \mathrm{y}_s) \in \mathbb{R}_+^{n_1+Sn_2} \colon \mathrm{e}^{\mathsf{T}}\mathrm{x} + \mathrm{e}^{\mathsf{T}}\mathrm{y}_s = 1,\ s = 1, \dots, S\right\}$ are simply $V(P) = \{\mathrm{e}_i : i \in [1\colon n_1]\} \cup \left\{\sum_{s=1}^{S} \mathrm{e}_{j_s+n_1+(s-1)n_2} : \mathrm{j} \in [1\colon n_2]^S\right\}$
- Thus, solving the linear optimization problem amounts to finding $S+1$ largest/smallest values.
- S_k is easily updated by dropping coordinates that became zero.
- We can choose different starting points and pick the best solution.

Contents

Comparing the copositive bounds

Results

- Test with different sizes of instances.
- 20 randomly generated instances per size category.
- Average %-gap between the upper bound and the two lower bounds.

(n_1, n_2, S)	Time		Gap		max Gap		Vs.	
	DNN1	DNN2	DNN1	DNN2	DNN1	DNN2	avgG	maxG
(5,5,10)	1.647	0.191	1.075%	1.373%	13.26%	14.167%	0.019%	0.016%
(5,10,10)	28.05	0.237	0.003%	0.039%	0.056%	0.379%	0.001%	0.001%
(5,20,10)	1132.45	0.754	0.428%	1.281%	5.136%	18.633%	0.001%	0.001%
(10,5,10)	2.942	0.438	1.365%	1.665%	17.201%	18.306%	0.014%	0.011%
(20,5,10)	5.515	0.977	9.432%	9.305%	35.772%	35.824%	0.002%	0.001%
(5,5,20)	37.588	0.546	0.807%	1.394%	13.756%	14.317%	0.029%	0.023%
(5,5,40)	1504.456	0.998	1.09%	1.579%	21.536%	21.625%	0.006%	0.004%

Comparing different Upperbounds

(n_1, n_2, S)	Time					Gap				
	DNN	`Gurobi`	`fmincon`	PFW	PFWM	DNN	`Gurobi`	`fmincon`	PFW	PFWM
(10,5,10)	0,274	291,987	0,701	**0,031**	5,419	16,33%	0,32%	3,01%	1,92%	**0,25%**
(20,5,10)	0,492	300,406	0,994	**0,061**	7,170	50,41%	2,13%	10,57%	2,10%	**0,23%**
(5,10,10)	0,362	300,483	2,622	**0,039**	8,463	15,24%	0,09%	2,31%	1,32%	**0,08%**
(5,20,10)	0,614	300,743	2,872	**0,082**	16,843	13,76%	0,15%	2,86%	1,88%	**0,04%**
(5,5,10)	0,362	296,317	0,764	**0,024**	4,207	19,06%	**0,32%**	0,94%	1,26%	**0,32%**
(5,5,20)	0,108	300,527	1,698	**0,050**	13,319	19,58%	**0,15%**	1,29%	2,54%	**0,15%**

Table: Results of cold-starting the upper-bound procedures

- Multistarting PFW yields small gaps in reasonable time.

Warmstarting primal procedures

(n_1, n_2, S)	**Gap**				Time			
	DNN+FW+`Gurobi`	DNN+`fmincon`	DNN+`Gurobi`	DNN+FW	DNN+FW+`Gurobi`	DNN+`fmincon`	DNN+`Gurobi`	DNN+FW
(10,5,10)	**0,26%**	0,63%	0,31%	0,27%	292,431	1,139	291,379	**0,534**
(20,5,10)	**0,54%**	1,43%	1,87%	0,82%	301,150	1,797	301,163	**0,796**
(5,10,10)	**0,08%**	0,72%	0,09%	0,10%	301,038	1,819	301,048	**0,591**
(5,20,10)	**0,04%**	0,47%	0,17%	**0,04%**	301,598	6,654	301,595	**0,908**
(5,5,10)	**0,32%**	0,33%	**0,32%**	0,34%	301,769	1,125	301,419	**0,548**
(5,5,20)	**0,15%**	0,25%	**0,15%**	0,21%	301,089	1,982	301,006	**0,505**

Table: Warmstarting the upper bound procedures: gaps and runtimes

- Using the upper bounds to warm start primal algorithms can be advantageous.

Experiments on dissecting the probability measure

Recall the **successively refining partition of the sample space**:

$$\Xi = \{\xi_1, \dots, \xi_S\}$$

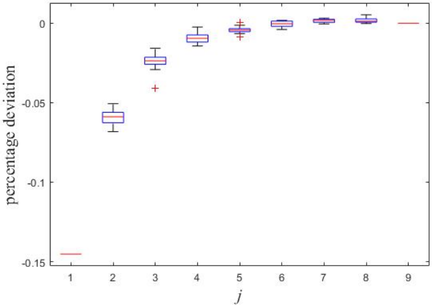
$$\dots$$

$$(\Xi_1^{(j)}, \Xi_2^{(j)}, \dots, \Xi_{m_j}^{(j)})$$

$$\dots$$

$$(\Xi_1^{(2)}, \Xi_2^{(2)}, \dots, \Xi_{m_2}^{(2)})$$

$$(\{\xi_1\}, \{\xi_2\}, \dots, \{\xi_S\}),$$



Application to Mean-Varaince Portfolio Optimization

- We consider the historical mean return and variance of returns of ten assets based on ten time series.
- For assets $i \in [1:5]$ a longer time series of 48 days is available while for assets $i \in [6:10]$ only a short time series of 12 days is available.
- Hence, we consider the portion of the covariance matrix that involve the latter assets to be uncertain and we consider the model:

$$\min_{x \in T^5} \left\{ \mu_x^\mathsf{T} x + x^\mathsf{T} \Sigma_{xx} x + \mathbb{E}\left[\min_{y \in P_x} \mu_y^\mathsf{T} y + 2x^\mathsf{T} \Sigma_{xy} y + y^\mathsf{T} \Sigma_{yy} y \right] \right\} \tag{3}$$

Application to Mean-Varaince Portfolio Optimization

- We consider the historical mean return and variance of returns of ten assets based on ten time series.
- For assets $i \in [1:5]$ a longer time series of 48 days is available while for assets $i \in [6:10]$ only a short time series of 12 days is available.
- Hence, we consider the portion of the covariance matrix that involve the latter assets to be uncertain and we consider the model:

$$\min_{\mathrm{x} \in T^5} \left\{ \mu_x^{\mathsf{T}} \mathrm{x} + \mathrm{x}^{\mathsf{T}} \Sigma_{xx} \mathrm{x} + \mathbb{E}\left[\min_{\mathrm{y} \in P_x} \mu_y^{\mathsf{T}} \mathrm{y} + 2\mathrm{x}^{\mathsf{T}} \Sigma_{xy} \mathrm{y} + \mathrm{y}^{\mathsf{T}} \Sigma_{yy} \mathrm{y} \right] \right\} \tag{4}$$

Despite the presence of a linear term, the above optimization problem can be rewritten as

$$\min_{x \in T^5} \left\{ \mathrm{x}^\mathsf{T} \bar{\bar{\Sigma}}_{xx} \mathrm{x} + \mathbb{E}\left[\min_{\mathrm{y} \in P_x} 2\mathrm{x}^\mathsf{T} \bar{\bar{\Sigma}}_{xy} \mathrm{y} + \mathrm{y}^\mathsf{T} \bar{\bar{\Sigma}}_{yy} \mathrm{y} \right] \right\}$$

where

$$\begin{aligned}
\bar{\bar{\Sigma}}_{xx} &:= \Sigma_{xx} + \tfrac{1}{2} \left(\mu_x \mathrm{e}^\mathsf{T} + \mathrm{e} \mu_x^\mathsf{T} \right) , \\
\bar{\bar{\Sigma}}_{xy} &:= \Sigma_{xy} + \tfrac{1}{2} \left(\mu_x \mathrm{e}^\mathsf{T} + \mathrm{e} \mu_y^\mathsf{T} \right) , \\
\bar{\bar{\Sigma}}_{yy} &:= \Sigma_{yy} + \tfrac{1}{2} \left(\mu_y \mathrm{e}^\mathsf{T} + \mathrm{e} \mu_y^\mathsf{T} \right) .
\end{aligned}$$

- We approximated the problem via the scenario approach
- the covariance matrix were generated from a elementwise normal distribution, i.e. $(\Sigma)_{ij} \sim N((\hat{\Sigma})_{ij}, \sigma)$ with $\hat{\Sigma}$ the empirical covariance matrix and $\sigma = 1$
- Scenarios with $S \in \{100, 500, 1000, 5000, 10000\}$ were generated.

Numerical experiments on portfolio optimization

S	**Obj**	**Gap**	**Time**
100	163.59	0,00%	0.47
500	163.60	0,00%	1.07
1000	163.60	0,00%	2.27
5000	163.57	0,00%	10.84
10000	163.58	0,00%	22.49

Table: Results of the portfolio optimization experiment

- In our experiments, all instances were solved using merely the reduced conic relaxation
- Hence, there are real-world applications where gaps can be closed confidently using the basic strategies.

[1] I. M. Bomze, M. Dür, E. De Klerk, C. Roos, A. Quist, and T. Terlaky. On copositive programming and standard quadratic optimization problems. *Journal of Global Optimization*, 24(2):163–185, 2002.

[2] I. M. Bomze, F. Rinaldi, and D. Zeffiro. Frank–Wolfe and friends: a journey into projection-free first-order optimization methods. *4OR – A Quarterly Journal of Operations Research*, , in press, 2021.

[3] S. Burer. On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120(2):479–495, 2009.

[4] F. Maggioni and G. C. Pflug. Bounds and approximations for multistage stochastic programs. *SIAM Journal on Optimization*, 26(1):831–855, 2016.

Thank you