

Notation

\mathbb{R} - Set of real numbers

S^n - Set of symmetric $n \times n$ matrices

S_+^n - Set of symmetric positive semidefinite $n \times n$ matrices

S_{++}^n - Set of symmetric positive definite $n \times n$ matrices

$M \succ 0$ - M is symmetric and positive definite

$M \succeq 0$ - M is symmetric and positive semidefinite

$C \bullet X$ - Matrix inner product = $Tr(C^T X)$

PSD - Positive Semidefinite

QCQP - Quadratically Constrained Quadratic Program

SDP - Semidefinite program

SOCP - Second Order Cone Program

WLOG - Without Loss Of Generality

Chapter 1

Introduction

In this thesis we will study quadratically constrained quadratic programs (QCQP)

Definition 1.1 (QCQP). We say minimization problem is Quadratically Constrained Quadratic Program (or shortly just QCQP) if it has the form of

$$\begin{aligned} & \text{minimize} && x^T A_0 x + a_0^T x \\ & \text{subject to} && x^T A_k x + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \end{aligned} \quad (1.1)$$

Where $x \in \mathbb{R}^n$ is variable, and symmetric $n \times n$ matrices $A_0, A_1, \dots, A_m \in S^n$, vectors $a_0, \dots, a_m \in \mathbb{R}^n$ and scalars $\alpha_1, \dots, \alpha_m \in \mathbb{R}$ are given.

Matrices A_0, A_1, \dots, A_m are not necessarily positive semidefinite. Therefore the objective function as well as constraints may be nonconvex. Generally this problem is hard to solve. For example, 0-1 constraint $x_i \in \{0, 1\}$ can be reformulated as $x^T e_i e_i^T x - e_i^T x = 0$. Thus QCQP includes 0-1 programming, which describes various NP-hard problems, such as knapsack, stable set, max cut etc. On the other hand, this suggests that QCQP has many applications and is worth solving.

To this end, often used approach is relaxing QCQP to obtain problems which can be solved in polynomial time, namely semidefinite programming (SDP) or second order cone programming (SOCP). Let us define these two classes and see why are they easier to solve.

1.1 Convex optimization classes

In this section we will first talk generally about convex programming, introduce primal and dual problem and their relationship. In the following subsections we will introduce some important classes of convex optimization problems. Namely

linear programming (LP), convex quadratic programming (QP), second order cone programming (SOCP) and finally semidefinite programming (SDP). We will formulate standard forms of these problems and their duals, explain relationships between mentioned classes and briefly state some important properties. For reference and more information about this topic please see [TODO REFS].

Let us first define problem of convex optimization and its lagrange dual.

Definition 1.2 (Convex Program). We say minimization problem is Convex Program (or shortly just CP) if it has the form of

$$\begin{aligned} & \text{minimize} && f_0(x) \\ & \text{subject to} && Ax = b, \\ & && f_k(x) \leq 0, \quad k = 1, \dots, m \end{aligned} \tag{CP}$$

Where $x \in \mathbb{R}^n$ is a variable, convex functions f_0, f_1, \dots, f_m from \mathbb{R}^n to $\mathbb{R}, \mathbb{R}^{n_1}, \dots, \mathbb{R}^{n_m}$ respectively, $l \times n$ matrix A , and vector $b \in \mathbb{R}^l$ are given problem data.

The Dual form of standard Convex Programming problem (or shortly CD) is

$$\begin{aligned} & \text{maximize} && g(\lambda, \nu) \\ & \text{subject to} && \lambda \in \mathbb{R}_+^m \end{aligned} \tag{CD}$$

Where $\lambda \in \mathbb{R}^m, \nu \in \mathbb{R}^l$ are variables, and function $g : \mathbb{R}^{m+l} \rightarrow \mathbb{R}$ is lagrange dual function

$$g(\lambda, \nu) = \min_x f_0(x) + \sum_{k=1}^m \lambda_k f_k(x) + \nu^T (Ax - b) \tag{1.2}$$

Following theorem captures relation between these two problems.

Theorem 1.1. Let $f_0, f_1, \dots, f_m, A, b$ be given as above defining (CP) and (CD). Then

- (a) (*weak duality*) For any feasible \bar{x} in (CP) and any feasible $(\bar{\lambda}, \bar{\nu})$ in (CD), it holds that $f_0(\bar{x}) \geq g(\bar{\lambda}, \bar{\nu})$. Moreover, if $f_0(\bar{x}) = g(\bar{\lambda}, \bar{\nu})$ then $(\bar{x}), (\bar{\lambda}, \bar{\nu})$ are optimal solutions to (CP), (CD) respectively.
- (b) (*strong duality*) If (CP) has a strictly feasible solution (Slater's condition holds) i.e. there exists $x_0 \in \mathbb{R}^n$ such that $Ax_0 = b$ and $f_k(x_0) < 0$ for all $k = 1, \dots, m$, then the strong duality holds, that is both problems have optimal solutions, and their values coincide.

1.1.1 Linear programming

When both objective and constraint functions are linear (affine), the problem is called linear program or shortly LP. In this section we will introduce standard

and dual form of LP, and few important properties. For reference and more information about this topic please see [5, 6,] .

Definition 1.3. We say minimization problem is Linear program (or shortly LP) if it has form of

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b, \\ & && x \in \mathbb{R}_+^n \end{aligned} \tag{LP}$$

Where $x \in \mathbb{R}^n$ is variable, the real $m \times n$ matrix A and vectors $b \in \mathbb{R}^m, c \in \mathbb{R}^n$ are given.

The dual form of standard Linear Programming problem (or shortly LD) is

$$\begin{aligned} & \text{maximize} && b^T y \\ & \text{subject to} && A^T y + s = c, \\ & && s \in \mathbb{R}_+^n \end{aligned} \tag{LD}$$

Where $y \in \mathbb{R}^m$, $s \in \mathbb{R}^n$ are variables; and problem data A, b, c are given from the primal LP above.

Remark. Problems of linear programming can be introduced in various forms (including \geq, \leq inequalities, free variables, possibly some linear fractions in objective) but they all can be transformed to

Various properties holds for this (LP)-(LD) pair of problems, let us mention the most important ones.

Theorem 1.2. *Let A, b, c be given as above, defining (LP) and (LD). Then*

- (a) *(weak duality) For any feasible \bar{x} in (LP) and any feasible \bar{y} in (LD), it holds that $c^T \bar{x} \geq \bar{y}^T b$. Moreover, if $c^T \bar{x} = \bar{y}^T b$, then $(\bar{x}), (\bar{y})$ are optimal solutions to (LP), (LD) respectively.*
- (b) *(strong duality I) If (LP) has an optimal solution, then so does (LD) and their values coincide.*
- (c) *(strong duality II) IF (LP) and (LD) both have feasible solutions, then they both have optimal solutions and their optimum values are the same.*

Since this is well known fact, we omit the proof. Notice that part (c) states, that there is zero duality gap in linear programming with no condition on feasible solutions (only their existence is required).

1.1.2 Convex Quadratic Programming

Linear programming can be easily generalized to by introducing convex quadratic objective function and additional convex quadratic constraints into Quadratic Program (QP) and Quadratically Constrained Quadratic Program (QCQP) respectively. In this section we will introduce standard and dual form of QP and QCQP, and few important properties. For reference and more information about this topic see [5,]

Definition 1.4. We say minimization problem is convex Linearly Constrained Quadratic Program (or shortly convex LCQP) if it has form of

$$\begin{aligned} & \text{minimize} && x^T P x + c^T x \\ & \text{subject to} && A x = b, \\ & && x \in \mathbb{R}_+^n \end{aligned} \quad (\text{LCQP})$$

Where $x \in \mathbb{R}^n$ is variable, real $n \times n$ symmetric semidefinite matrix $P \in \mathbb{S}_+^n$, real $m \times n$ matrix A and vectors $b \in \mathbb{R}^m, c \in \mathbb{R}^n$ are given.

Definition 1.5. We say minimization problem is convex Quadratically Constrained Quadratic Program (or shortly convex QCQP) if it has form

$$\begin{aligned} & \text{minimize} && x^T P_0 x + q_0^T x + r_0 \\ & \text{subject to} && x^T P_k x + q_k^T x + r_k \leq 0, \quad k = 1, \dots, m \end{aligned} \quad (\text{QCQP})$$

Where $x \in \mathbb{R}^n$ is variable, $n \times n$ symmetric semidefinite matrices $P_0, P_1, \dots, P_m \in \mathbb{S}_+^n$ and vectors $q_0, q_1, \dots, q_m \in \mathbb{R}^n$ and scalars $r_0, r_1, \dots, r_m \in \mathbb{R}$ are given.

The dual form of standard convex quadratic programming problem (or shortly QCQD) is

$$\begin{aligned} & \text{maximize} && -q(\lambda)^T P(\lambda)^{-1} q(\lambda) + r(\lambda) \\ & \text{subject to} && \lambda \geq 0 \\ & && P(\lambda) \succeq 0 \end{aligned} \quad (\text{QCQD})$$

Where

$$P(\lambda) = P_0 + \sum_{k=1}^m \lambda_k P_k, \quad q(\lambda) = q_0 + \sum_{k=1}^m \lambda_k q_k, \quad r(\lambda) = r_0 + \sum_{k=1}^m \lambda_k r_k.$$

Where $\lambda \in \mathbb{R}^m$ is variable; and problem data $P_0, P_1, \dots, P_m, q_0, q_1, \dots, q_m, r_0, r_1, \dots, r_m$ are given from the primal QCQP above.

Relation to previous classes: Convex quadratic programs include LP as a special case, by taking matrices $P_0, P_1, \dots, P_m = 0$. Also it is easy to see that LP is a proper subclass of convex QCQP, since it does not include quadratic constraints. The more solid proof of that is that strong duality from Theorem 1.2 does not hold for all convex quadratic programs.

Example 1. Consider following convex QP with variable $x \in \mathbb{R}^2$ and its dual with variable $\lambda \in \mathbb{R}^2$

$$\begin{aligned} \text{(P)} \quad & \begin{aligned} & \text{minimize} && x^T x \\ & \text{subject to} && x^T x + (1, 0)x \leq 0 \\ & && (1, 0)x + 1 \leq 0 \end{aligned} \end{aligned} \tag{1.3}$$

$$\begin{aligned} \text{(D)} \quad & \begin{aligned} & \text{maximize} && -(\lambda_1 + \lambda_2, 0) \text{diag}\left(\frac{1}{\lambda_1 + 1}, \frac{1}{\lambda_1 + 1}\right)(\lambda_1 + \lambda_2, 0)^T + \lambda_2 \\ & \text{subject to} && \lambda \geq 0 \end{aligned} \end{aligned} \tag{1.4}$$

The primal problem (P) has only one feasible solution $x = (1, 0)^T$ and so its optimal value is 1. On the other hand solving (D) is not that trivial. Nevertheless, after some effort it can be shown that optimal value is $1/4$ for $\lambda = (0, 0.5)^T$.

TODO: try to find more trivial example

For convex quadratic programs, and for all the convex programs, there holds theorem 1.1, which is weaker than strong duality in LP and requires strictly feasible solution. Therefore the Slater's condition, which is not met in the previous example, is necessary.

1.1.3 Second order cone programming

SOCP is another convex optimization problem which can be solved with great efficiency by interior point methods. Let us first define second order cone.

Definition 1.6 (Second order cone). We say Q_n is second order cone of dimension n if

$$Q_n = \{x = (x_0, \bar{x}) \in \mathbb{R}^n \mid \|\bar{x}\| \leq x_0\},$$

where $\|\cdot\|$ is standard euclidean norm and $x_0 \in \mathbb{R}$, $\bar{x} \in \mathbb{R}^{n-1}$.

One can define SOCP problem as LP with additional second order cone constraints.

Definition 1.7 (SOCP). We say minimization problem is Second Order Cone Program (or shortly SOCP) if it has form of

$$\begin{aligned} & \begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b \\ & && x \in Q_n \end{aligned} \end{aligned} \tag{SOCP}$$

Where $x \in \mathbb{R}^n$ is variable, and $m \times n$ real matrix A , vectors $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and second order cone Q_n are given problem data.

$$\begin{aligned} & \begin{aligned} & \text{maximize} && b^T \lambda \\ & \text{subject to} && A^T \lambda + s = c \\ & && s \in Q_n \end{aligned} \end{aligned} \tag{SOCD}$$

Where $\lambda \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$ are variables, and problem data A, b, c, Q_n are given from the primal SOCP above.

Relation to previous classes: Second Order Cone Programming includes convex QP as special case. We will demonstrate procedure proposed in [4] used to reformulate convex QP as SOCP.

Let us have convex QCQP. In other words, suppose that $n \times n$ matrices A_k , $k = 0, \dots, m$ are positive semidefinite.

$$\begin{aligned} & \text{minimize} && x^T P_0 x + q_0^T x + r_0 \\ & \text{subject to} && x^T P_k x + q_k^T x + r_k \leq 0, \quad (k = 1, \dots, m) \end{aligned} \quad (1.5)$$

First of all, we can get rid of quadratic objective function by introducing new variable $t \in \mathbb{R}$ and constraint $x^T P_0 x + q_0^T x + r_0 \leq t$ and then rewrite the whole program in variable $(t, x) \in \mathbb{R}^{n+1}$, with objective minimize t . To avoid tedious notation WLOG suppose that considered program already has linear objective function (i.e. $P_0 = 0$). Also suppose that we have separated all the linear constraints (ones where $P_k = 0$) and arrange them into more compact form $Ax = b$. Even if we did not, the following procedure will still be correct, but will result into formulating those linear constraints in more complicated way.

We will omit indices for simplicity and transform each convex quadratic constraint

$$x^T P x + q^T x + r \leq 0 \quad (1.6)$$

into second order cone constraint. Suppose that $\text{rank } P = h \geq 1$. Then there exists $n \times h$ matrix L such that $P = LL^T$. Such L can be computed by Cholesky factorization of P . Now rewrite (1.6) as

$$(L^T x)^T (L^T x) \leq -q^T x - r. \quad (1.7)$$

It is known and also easily verified that $w \in \mathbb{R}^t$, $\xi \in \mathbb{R}$ and $\eta \in \mathbb{R}$ satisfy

$$w^T w \leq \xi \eta, \quad \xi \geq 0 \quad \text{and} \quad \eta \geq 0$$

if and only if they satisfy

$$\left\| \begin{pmatrix} \xi - \eta \\ 2w \end{pmatrix} \right\| \leq \xi + \eta.$$

If we take $w = L^T x$, $\xi = 1$ and $\eta = -q^T x - r$ we can convert inequality (1.7) into

$$\begin{pmatrix} v_0 \\ v \end{pmatrix} = \begin{pmatrix} 1 - q^T x - r \\ 1 + q^T x + r \\ 2L^T x \end{pmatrix} \in \mathbb{R}^{h+2} \quad \text{and} \quad \|v\| \leq v_0. \quad (1.8)$$

Which is a second order cone constraint. Now the intersection of all such constraints is again second order cone, thus we have obtained problem of SOCP.

TODO: is the following argument and example ok?

On the other hand, not all SOCP problems can be reformulated as convex QP. Therefore convex QP is a proper subclass of SOCP. Here is an example of SOCP equivalent to nonconvex QP.

Example 2. Consider problem (1.12) from the definition. The conic constraint $x = (x_1, \bar{x}) \in Q_n$ can be equivalently formulated as

$$\|\bar{x}\| \leq x_1 \Leftrightarrow \left\{ \begin{array}{l} \bar{x}^T \bar{x} \leq x_1^2 \\ 0 \leq x_1 \end{array} \right\} \Leftrightarrow \left\{ \begin{array}{l} x^T \begin{pmatrix} -1 & 0 \\ 0 & I_{n-1} \end{pmatrix} x \leq 0 \\ 0 \leq x_1 \end{array} \right\}. \quad (1.9)$$

Where last quadratic constraint is obviously not convex, since the matrix has negative eigenvalue.

1.1.4 Semidefinite programming

Firstly, let us introduce notation we will use to simplify the standard form..

Definition 1.8. Let A, X be real $n \times m$ matrices, we will denote their inner product

$$A \bullet X = \text{Tr}(A^T X).$$

Where $\text{Tr}(M)$ denotes trace of matrix M i.e. sum of the diagonal elements of M .

Definition 1.9 (SDP). We say minimization problem is Semidefinite program (or shortly SDP) if it has form

$$\begin{array}{ll} \text{minimize} & A_0 \bullet X \\ \text{subject to} & A_k \bullet X = b_k, \quad (k = 1, \dots, m) \\ & X \succeq 0. \end{array} \quad (\text{SDP})$$

Where $X \in S_+^n$ is variable, and matrices $A_0, A_1, \dots, A_m \in S^n$ and scalars $b_1, \dots, b_m \in \mathbb{R}$ are given.

We say minimization problem is Semidefinite Dual program (or shortly SDD) if it has form

$$\begin{array}{ll} \text{maximize} & b^T \lambda \\ \text{subject to} & \sum_{k=1}^m \lambda_k A_k + \alpha_k \leq 0, \quad (k = 1, \dots, m) \\ & X \succeq 0. \end{array} \quad (\text{SDD})$$

Where $\lambda = (\lambda_1, \dots, \lambda_m)^T \in \mathbb{R}^m$ and $X \in S_+^n$ are variables and problem data $A_1, \dots, A_m, b = (b_1, \dots, b_m)^T \in \mathbb{R}^m$ are given from the primal problem above.

1.2 Semidefinite programming

In general SDP can be formulated as problem of mathematical programming of the form

Definition 1.10 (SDP). We say minimization problem is Semidefinite program (or shortly SDP) if it has form

$$\begin{aligned} & \text{minimize} && A_0 \bullet X \\ & \text{subject to} && A_k \bullet X + \alpha_k \leq 0, \ (k = 1, \dots, m) \\ & && X \succeq 0. \end{aligned} \quad (1.10)$$

Where $A \bullet X = \text{TR}(A^T X) = \sum_{i=1}^n \sum_{j=1}^n A_{i,j} X_{i,j}$; and $X \in S_+^n$ is variable, and matrices $A_0, \dots, A_m \in S^n$ and scalars $\alpha_1, \dots, \alpha_m \in \mathbb{R}$ are given.

Note that adding appropriate slack variables into (1.10), one can obtain SDP with just equalities in linear constraints. In some literature, following is considered as standard form of SDP.

$$\begin{aligned} & \text{minimize} && A_0 \bullet X \\ & \text{subject to} && A_k \bullet X + \alpha_k = 0, \ (k = 1, \dots, m) \\ & && X \succeq 0. \end{aligned} \quad (1.11)$$

However, (1.11) and (1.10) are equivalent, the form of (1.10) suits better to our needs, we will not require equalities in SDP.

Remark. It is known, that $S_+^n = \{M \in S^n | M \succeq 0\}$ is closed convex cone in \mathbb{R}^{n^2} (for details see i.e.[6][7]).

Therefore notation " \succeq " really defines partial ordering of S^n . Moreover, looking at X as vector in S^n we see that SDP is very similar to linear programming (LP). The objective function and constraints are linear. However, in LP x must lie in non-negative orthant (which is cone), and in SDP X must lie in cone of positive semidefinite matrices. In other words, $x \geq 0$ is replaced with $X \succeq 0$. This gives an idea why solving SDP should be possible in polynomial time. The most widely used for SDPs are interior point methods [SOME REFS].

LMI FORMULATION? ADD DUAL PROBLEM AND SOMETHING ABOUT STRONG DUALITY

1.3 Second order cone programming

SOCP is another convex optimization problem which can be solved with great efficiency by interior point methods. Let us first define second order cone.

Definition 1.11 (Second order cone). We say Q_n is second order cone of dimension n if

$$Q_n = \{x = (x_0, \bar{x}) \in \mathbb{R}^n \mid \|\bar{x}\| \leq x_0\},$$

where $\|\cdot\|$ is standard euclidean norm and $x_0 \in \mathbb{R}$, $\bar{x} \in \mathbb{R}^{n-1}$.

One can define SOCP problem as LP with additional second order cone constraints.

Definition 1.12 (SOCP). We say minimization problem is Second Order Cone Program (or shortly SOCP) if it has form of

$$\begin{aligned} & \text{minimize} && f^T x \\ & \text{subject to} && \|A_k x + b_k\| \leq c_k^T x + \gamma_k, \quad (k = 1, \dots, m) \\ & && Fx = g \end{aligned} \quad (1.12)$$

Where $x \in \mathbb{R}^n$ is variable, and $n_k \times n$ matrices A_k , $p \times n$ matrix F , vectors $f \in \mathbb{R}^n$, $b_k \in \mathbb{R}^{n_k}$, $c_k \in \mathbb{R}^n$, $g \in \mathbb{R}^p$ and scalars $\gamma_k \in \mathbb{R}$ are given.

The constraints $\|A_k x + b_k\| \leq c_k^T x + \gamma_k$ are second order cones of dimension n_k for variables $x^k = (c_k^T x + d_k, A_k x + b_k) = (x_0^k, \bar{x}^k)$. Variables $x^k \in \mathbb{R}^{n_k+1}$ are just affine transformations of x , therefore conic constraints of the problem are just affine transformations of constraint $x \in Q_n$ where Q_n is second order cone from definition 1.11.

1.3.1 Relation between SOCP and QCQP

Getting from SOCP to QCQP is simple. Squaring the second order cone constraints we can formulate each SOCP as QCQP.

$$\|A_k x + b_k\| \leq c_k^T x + \gamma_k \Leftrightarrow \begin{cases} (A_k x + b_k)^T (A_k x + b_k) \leq (c_k^T x + \gamma_k)^2 \\ 0 \leq c_k^T x + \gamma_k \end{cases} \quad (1.13)$$

$$\Leftrightarrow \begin{cases} x^T (A_k^T A_k - c_k^T c_k I) x + 2(A_k^T b_k - d_k c_k)^T x + (b_k^T b_k - \gamma_k^2) \leq 0 \\ -c_k^T x - \gamma_k \leq 0 \end{cases} \quad (1.14)$$

It is not that simple other way around, but the case when QCQP is convex can be handled. That is not the only case, as we can see in the previous formula taking c_k with large norm, can make the first constraint nonconvex. We will demonstrate procedure proposed in [4]

Let us have convex QCQP. In other words, suppose that $n \times n$ matrices A_k , $k = 0, \dots, m$ are positive semidefinite.

$$\begin{aligned} & \text{minimize} && x^T A_0 x + a_0^T x \\ & \text{subject to} && x^T A_k x + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \end{aligned} \quad (1.15)$$

First of all, we can get rid of quadratic objective function by introducing new variable $t \in \mathbb{R}$ and constraint $x^T A_0 x + a_0^T x \leq t$ and then rewrite the whole

program in variable $(t, x) \in \mathbb{R}^{n+1}$, with objective minimize t . To avoid tedious notation WLOG suppose that considered program already has linear objective function (i.e. $A_0 = 0$).

We will omit indices for simplicity and transform each convex quadratic constraint

$$x^T A x + a^T x + \alpha \leq 0 \quad (1.16)$$

into second order cone constraint. Suppose that $\text{rank } A = h \geq 1$. Then there exists $n \times h$ matrix L such that $A = LL^T$. Such L can be computed by Cholesky factorization of C . Now rewrite (1.16) as

$$(L^T x)^T (L^T x) \leq -a^T x - \alpha. \quad (1.17)$$

It is known and also easily verified that $w \in \mathbb{R}^t$, $\xi \in \mathbb{R}$ and $\eta \in \mathbb{R}$ satisfy

$$w^T w \leq \xi \eta, \quad \xi \geq 0 \quad \text{and} \quad \eta \geq 0$$

if and only if they satisfy

$$\left\| \begin{pmatrix} \xi - \eta \\ 2w \end{pmatrix} \right\| \leq \xi + \eta.$$

If we take $w = L^T x$, $\xi = 1$ and $\eta = -a^T x - \alpha$ we can convert inequality (1.17) into

$$\begin{pmatrix} v_0 \\ v \end{pmatrix} = \begin{pmatrix} 1 - a^T x - \alpha \\ 1 + a^T x + \alpha \\ 2L^T x \end{pmatrix} \in \mathbb{R}^{h+2} \quad \text{and} \quad \|v\| \leq v_0. \quad (1.18)$$

In conclusion, we are able to transform SOCP into QCQP. We can also transform convex QCQP into SOCP.

1.4 Relaxations

As mentioned in the beginning, our strategy is to relax nonconvex QCQPs (1.1) to easier problem. Let us first explore what is relaxation and how can it be useful.

Definition 1.13 (Relaxation). Relaxation of minimization problem $z = \min\{c(x) | x \in X \subseteq \mathbb{R}^n\}$ is another minimization problem $z_R = \min\{c_R(x) | x \in X_R \subseteq \mathbb{R}^n\}$, with properties: $X \subseteq X_R$ and $c_R(x) \leq c(x) \forall x \in X$.

It easily follows, that solving relaxed problem will provide a lower bound on the optimal value of original problem. Sometimes we can from relaxation extract also feasible solution of original problem and obtain upper bound for optimal value. Moreover, these bounds may not only give us some idea about the optimal value, but can provide a means to find optimal solution of original problem.

1.5 SDP relaxation

Consider QCQP (1.1). Using identity $x^T A_k x = A_k \bullet x x^T$ it can be rewritten as follows

$$\begin{aligned} & \text{minimize} && A_0 \bullet X + a_0^T x \\ & \text{subject to} && A_k \bullet X + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \\ & && X = x x^T \end{aligned} \quad (1.19)$$

Since matrices A_k $k = 0, \dots, m$ are symmetric, we can WLOG suppose that X is also symmetric. In order to get form similar to SDP (1.10), we need to hide linear terms $a_k^T x$. This can be done by lifting problem into higher dimension. Introducing $n + 1 \times n + 1$ matrices

$$M_k = \begin{pmatrix} \alpha_k & \frac{1}{2} a_k^T \\ \frac{1}{2} a_k & A_k \end{pmatrix}, \quad Y = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix},$$

where $k = 0, \dots, m$, $\alpha_0 = 0$ and $X = x x^T$. We also suppose that $Y \in S^{n+1}$ (since we suppose $X \in S^n$). Using this notation (1.19) can be written as

$$\begin{aligned} & \text{minimize} && M_0 \bullet Y \\ & \text{subject to} && M_k \bullet Y \leq 0, \quad (k = 1, \dots, m) \\ & && X = x x^T \end{aligned} \quad (1.20)$$

This problem looks almost like SDP but the last constraint is not quite what we need. Following lemma shows how to write it in more suitable form.

Lemma 1.1. *Let us have a vector $x \in \mathbb{R}^n$, $n \times n$ symmetric matrix X , and $n + 1 \times n + 1$ symmetric matrix Y such that*

$$Y = \begin{pmatrix} 1 & x^T \\ x & X \end{pmatrix}.$$

Then

- (i) $X \succeq x x^T$ if and only if $Y \succeq 0$.
- (ii) $X = x x^T$ holds if and only if $Y \succeq 0$ and $\text{rank } Y = 1$.

Proof. Part (i) follows from Schur complement lemma for PSD (see Theorem 2.1 in appendices) To apply the this theorem we only need symmetry of Y and $1 \succ 0$ which holds.

Part (ii). (\Rightarrow) If $X = x x^T$, then also $X \succeq x x^T$, thus $Y \succeq 0$ holds by (i). And since $X = x x^T$,

$$Y = \begin{pmatrix} 1 & x^T \\ x & x x^T \end{pmatrix} = \begin{pmatrix} 1 \\ x \end{pmatrix} (1, x^T).$$

Hence $\text{rank } Y = 1$.

(\Leftarrow) Let $Y \succeq 0$ and $\text{rank } Y = 1$. Since $\text{rank } Y = 1$, each row of Y must be scalar

multiple of first (nonzero) row $(1, x^T)$. To match the first column it must be that $(i+1)$ -th row of Y is $x_i(1, x^T)$, for $i = 1, \dots, n$. Therefore, $X = xx^T$. \square

Remark. Notice that we have proven last implication of (ii) without using $Y \succeq 0$. In fact it is redundant. It also holds that $X = xx^T \Leftrightarrow \text{rank } Y = 1$. However, this redundant constraint $Y \succeq 0 \Leftrightarrow X \succeq xx^T$ will let us keep something from the rank 1 constraint after relaxing it.

Using the lemma and relaxing the nonconvex rank 1 constrain we obtain SDP relaxation of (1.1)

$$\begin{aligned} & \text{minimize} && M_0 \bullet Y \\ & \text{subject to} && M_k \bullet Y \leq 0, \quad (k = 1, \dots, m) \\ & && Y \succeq 0. \end{aligned} \tag{1.21}$$

Or expanding terms $M_k \bullet Y$ to see the original problem variable and data we have

$$\begin{aligned} & \text{minimize} && A_0 \bullet X + a_0^T x \\ & \text{subject to} && A_k \bullet X + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \\ & && X \succeq xx^T. \end{aligned} \tag{1.22}$$

Looking back to definition of relaxation we see, that above problem is not a relaxation of QCQP, because it has different variable space $(X, x) \in \mathbb{R}^{n \times n} \times \mathbb{R}^n$ instead of $x \in \mathbb{R}^n$. However, it can be resolved by constructing equivalent problem to (1.19) by adding new redundant variable and constraint.

Proposition 1. *Problem (1.22) is relaxation of QCQP (1.1) with additional variable X and additional constraint $X = xx^T$ (wich is equivalent to original QCQP).*

Proof. Described QCQP with additional variable and constraint can be reformulated as (1.19). Comparing with SDP relaxation (1.22) these problems are equal up to the last constraint where $X = xx^T$ is replaced by $X \succeq xx^T$. The first one implies the second, because

$$X = xx^T \Rightarrow X - xx^T = 0 \succeq 0 \Rightarrow X \succeq xx^T.$$

Hence feasible set of (1.19) is subset of feasible set of (1.22). Same objective function then implies the second property of relaxation. \square

Semidefinite program relaxations can provide tight bounds, but they can also be expensive to solve by classical interior point methods. Many researchers have suggested various alternatives to interior point methods for improving solution times. Others have studied different types of relaxations, for example, ones based on LP or SOCP. [1]

1.6 SOCP relaxation

These relaxations are expected to provide weaker bounds in less time compared to SDP relaxations. In fact, SOCP relaxations are often constructed as further relaxations of SDP relaxations. So, in certain sense, one can see that SOCP relaxations are never tighter than their SDP counterparts. [1]

Using the procedure from section 1.3.1, any convex QCQP relaxation can be easily formulated as SOCP relaxation. Therefore we can consider any convex QCQP relaxation as SOCP relaxation. Specifically, such a convex QCQP relaxation may be represented as

$$\begin{aligned} & \text{minimize} && x^T B_0 x + b_0^T x \\ & \text{subject to} && x^T B_k x + b_k^T x + \beta_i \leq 0, \quad (k = 1, \dots, l) \end{aligned} \quad (1.23)$$

where all $B_k \succeq 0$ for $k = 0, \dots, l$. We say that (1.23) is SOCP relaxation of (1.1) if that x is feasible for (1.1) implies that x is feasible for (1.23) and $x^T B_0 x + b_0^T x \leq x^T A_0 x + a_0^T x$ holds.

It is well known that SOCPs can also be solved as SDP. In [4] it is shown, that (1.23) is equivalent to the SDP

$$\begin{aligned} & \text{minimize} && B_0 \bullet X + b_0^T x \\ & \text{subject to} && B_k \bullet X + b_k^T x + \beta_i \leq 0, \quad (k = 1, \dots, l) \\ & && X \succeq x x^T. \end{aligned} \quad (1.24)$$

In [4] authors provided SOCP relaxation of QCQP (1.1) in original variable space $x \in \mathbb{R}^n$. First they WLOG assume that objective function is linear (otherwise we can introduce new variable $t \geq x^T A_0 x + a_0^T x$ and then minimize t). Then each A_k is written as

$$A_k = A_k^+ - A_k^-, \quad \text{where } A_k^+, A_k^- \succeq 0, \quad k = 1, \dots, m.$$

So that each constraint can be expressed as

$$x^T A_k^+ x + a_k^T x + \alpha_k \leq x^T A_k^- x.$$

Then an auxiliary variable $z_k \in \mathbb{R}$ is introduced to represent $x^T A_k^- x = z_k$, but also immediately relaxed as $x^T A_k^- x \leq z_k$, resulting in convex system

$$\begin{aligned} x^T A_k^+ x + a_k^T x + \alpha_k & \leq z_k \\ x^T A_k^- x & \leq z_k. \end{aligned} \quad (1.25)$$

Finally, z_k must be bounded in some fashion, say as $z_k \leq \mu \in \mathbb{R}$, or else the relaxation would be useless. In this way convex QCQP relaxation is constructed and it is simply transformed to SOCP using procedure from [4] described in section 1.3.1.

1.6.1 SOCP relaxation of semidefinite constraint

Semidefinite constraint $X - xx^T \succeq 0$ in (1.22) is equivalent to $C \bullet (X - xx^T) \geq 0$ for all $C \in S_+^n$. Using this, in [4] authors propose SOCP relaxation of the semidefinite constraint $X - xx^T \succeq 0$ by replacing it with multiple constraints of the form

$$X^T C_i x - C \bullet X \leq 0 \quad (i = 1, \dots, l).$$

Since $C_i \succeq 0$, these are convex quadratic constraints and using the procedure described earlier (in section 1.3.1) one can formulate them as second order cone constraints of the form

$$\begin{pmatrix} v_0^i \\ v^i \end{pmatrix} = \begin{pmatrix} 1 + C_i \bullet X \\ 1 - C_i \bullet X \\ 2L_i^T x \end{pmatrix}, \quad \|v^i\| \leq v_0^i, \quad (i = 1, \dots, l) \quad (1.26)$$

For further details about strength of this approach and suggested choice of matrices C_i based on problem data see [4].

1.7 Mixed SOCP-SDP relaxation

In [1] authors have introduced compromise, relaxation of QCQP (1.1) somewhere between SDP and SOCP. We will describe their approach on specific case they provide as gentle introduction. We are dealing with (1.1)

$$\begin{aligned} & \text{minimize} && x^T A_0 x + a_0^T x \\ & \text{subject to} && x^T A_k x + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \end{aligned} \quad (1.27)$$

Let $\lambda_{\min}(A_k)$ denote smallest eigenvalue of A_k . For all $k = 0, \dots, m$ define $\lambda_k = -\lambda_{\min}(A_k)$ so that $A_k + I\lambda_k \succeq 0$. Then (1.1) is equivalent to

$$\begin{aligned} & \text{minimize} && -\lambda_0 x^T x + x^T (A_0 + \lambda_0 I) x + a_0^T x \\ & \text{subject to} && -\lambda_k x^T x + x^T (A_k + \lambda_k I) x + a_k^T x + \alpha_k \leq 0, \quad (k = 1, \dots, m) \end{aligned}$$

which has following SOCP-SDP relaxation

$$\begin{aligned} & \text{minimize} && -\lambda_0 \text{Tr}(X) + x^T (A_0 + \lambda_0 I) x + a_0^T x \\ & \text{subject to} && -\lambda_k \text{Tr}(X) + x^T (A_k + \lambda_k I) x + a_k^T x + \alpha_k \leq 0, \\ & && (k = 1, \dots, m) \\ & && X \succeq xx^T \end{aligned} \quad (1.28)$$

Notice that other than $X \succeq xx^T$, the only variables in X to appear in the program are diagonal elements X_{jj} . Also, one can see that with fixed x the diagonal entries of X can be made arbitrarily large to satisfy all constraints with $\lambda_k > 0$, as well as drive objective to $-\infty$ if $\lambda_0 > 0$. Therefore, in general,

X_{jj} should be bounded to form a sensible relaxation. In the paper they use $x_j \in [0, 1]$ and $X_{jj} \leq x_j$ to establish boundedness.

Next proposition from [10] gives equivalent formulation of $X \succeq xx^T$ constraint, only in terms of x and diagonal entries of X .

Proposition 2 (Grone et. al). *Given a vector x and scalars X_{11}, \dots, X_{nn} , there exists a symmetric-matrix completion $X \in S^n$ of X_{11}, \dots, X_{nn} satisfying $X \succeq xx^T$ if and only if $X_{jj} \geq x_j^2$ for all $j = 1, \dots, n$. [10]*

Thus, in light of this proposition, the problem with additional bounding constraints $X_{jj} \leq x_j$, the problem (1.28) is equivalent to

$$\begin{aligned} & \text{minimize} && -\lambda_0 \text{Tr}(X) + x^T(A_0 + \lambda_0 I)x + a_0^T x \\ & \text{subject to} && -\lambda_k \text{Tr}(X) + x^T(A_k + \lambda_k I)x + a_k^T x + \alpha_k \leq 0, \\ & && (k = 1, \dots, m) \\ & && x_j^2 \leq X_{jj} \leq x_j \quad (j = 1, \dots, n) \end{aligned} \tag{1.29}$$

Compared to SDP relaxation (1.22), which has $O(n^2)$ variables, problem (1.29) has only $O(n)$ and hence is much faster to solve. On the other hand bound should be generally weaker than the SDP bound.

In the paper [1] is this approach further generalized and explored. Splitting $A = -D + (A + D)$ (instead of $-\lambda_k I + (A + \lambda_k I)$) with clever choice of C -block diagonal matrix D .

1.8 Using relaxations to solve original problem

1.8.1 Branch and bound

The basic idea of this method is to divide and conquer. There are 3 basics steps we will do repeatedly. Branch, compute lower bound, compute upper bound and prune.

- Branch. First divide by branching original problem into smaller subproblems i.e. by partition of the feasible set. One can do this for example by adding box constraints $l_i^j \leq x_i \leq u_i^j$, obtaining a new subproblem for each $j \in J$. It is important that union of feasible sets of all subproblems covers whole feasible set of original problem.
- Compute lower bound. Compute lower bounds for each subproblem (i.e. by solving relaxed problem).
- Compute upper bound and prune. Compute some upper bound for the optimal value (i.e. compute feasible solution for chosen subproblem or try to extract it from solution of relaxed problem). Having both lower

and upper bounds, we can conquer by pruning all the branches with lower bound greater than this upper bound. The optimal solution is surely not in these branches.

- Repeat. This process is repeated by branching remaining subproblems, computing new bounds for smaller problems, improving bounds and further pruning the tree of problems.

One can use more or less clever techniques to decide when, how and which subproblems to branch or how and when to compute bounds.

Chapter 2

Appendices

Theorem 2.1 (Schur complement lemma for PSD). *Let M be a symmetric matrix of the form*

$$M = \begin{pmatrix} A & B^T \\ B & C \end{pmatrix}.$$

It holds that if $A \succ 0$, then $M \succeq 0$ if and only if $C - B^T A^{-1} B \succeq 0$. [9]

Proof. (will be added later)

References

- [1] S. Burer, S. Kim, M. Kojima, *Faster, but Weaker, Relaxations for Quadratically Constrained Quadratic Programs*, April, 2013
<http://www.is.titech.ac.jp/~kojima/articles/socpsdp.pdf>
- [2] S. Kim, M. Kojima, *Exact Solutions of Some Nonconvex Quadratic Optimization Problems via SDP and SOCP Relaxations*, January 2002. Computational Optimization and Applications Vol.26 (2) 143-154 (2003)
<http://www.is.titech.ac.jp/research/research-report/B/B-375.ps.gz>
- [3] S. Kim, M. Kojima, M. Yamashita, *Second Order Cone Programming Relaxation of a Positive Semidefinite Constraint*, July 2002. Optimization Methods and Software Vol.18 (5) 535-541 (2003).
<http://www.is.titech.ac.jp/~kojima/articles/B-381.pdf>
- [4] S. Kim, M. Kojima, *Second Order Cone Programming Relaxations of Nonconvex Quadratic Optimization Problems*, Optim. Methods Softw., 15(3-4):201-224, 2001
http://www.researchgate.net/publication/2637515_Second_Order_Cone_Programming_Relaxation_of_Nonconvex_Quadratic_Optimization_Problems
- [5] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press 2004, seventh printing with corrections 2009
http://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf
- [6] R. M. Freund, *Introduction to Semidefinite Programming (SDP)*, 2009
http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-251j-introduction-to-mathematical-programming-fall-2009/readings/MIT6_251JF09_SDP.pdf
- [7] K. Pokorná, *Second order cone programming, (Master's thesis)*, 2013
<http://www.iam.fmph.uniba.sk/studium/efm/diplomovky/2013/pokorna/diplomovka.pdf>

- [8] M. S. Lobo, L. Vandenberghe, S. Boyd, H. Lebert, *Applications of second-Order cone programming*, Linear Algebra and its Applications 284, 1998, 193-228
<http://www.seas.ucla.edu/~vandenbe/publications/socp.pdf>
- [9] J. Gallier, *The Schur Complement and Symmetric Positive Semidefinite (and Definite) Matrices*, December 10, 2010
<http://www.cis.upenn.edu/~jean/schur-comp.pdf>
- [10] R. Grone, C. Johnson, E. Sá, H. Wolkowicz, *Positive definite completions of partial hermitian matrices*, Linear Algebra Applications, 58:109-124, 1984.