ECE 273 Convex Optimization Project

*Topic 1: Quadratically Constrained Quadratic Program

Zhixian Ye
Department of Electrical and Computer Engineering
University of California San Diego
ye@ucsd.edu

I. Introduction

Quadratically constrained quadratic programming (QCQP), whether convex or not, has been discussed a lot and now widely used in solving many engineering problems, like signal processing, wireless communications.

In this project, we are to conclude the ways to solve QCQP with certain relaxations, and see under what conditions the QCQP will be a convex programming problem. In the end, we use some packages to test on synthetic data to compare the result with existing other methods.

II. Methods

A. Problem Formulation of QCQP

Usually, the quadratically constrained quadratic program can be written in the following form:

$$\begin{array}{ll} \text{minimize} & f_0(x) = x^T P_0 x + q_0^T x + r_0 \\ \text{subject to} & f_i(x) = x^T P_i x + q_i^T x + r_i \leq 0, \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$ is the optimization variable, $P_i \in \mathbb{R}^{n \times n}$, $q_i \in \mathbb{R}^n$, $r_i \in \mathbb{R}$, for $i = 0, 1, ..., m$, which are given as problem data. For the quadratic matrices $P, Q \in \mathbb{S}^n$, where $\mathbb{S}^n$ denotes the set of all real symmetric $n \times n$ matrices

Normally, for simplicity reason, we don't add equality constraint, as they can be expressed as two quadratic inequality constraints:

$$h_i(x) \leq 0, -h_i(x) \leq 0$$

An simple but important example of quadratic equality constraint is $x_i^2 = 1$, which let x_i to be either +1 or -1, and it is also called Boolean quadratic program (BQP)

Also, the constraint becomes affine, i.e. $f_i(x) \leq 0$ if $P_i = 0$. If we need to deal with affine constraints differently from the quadratic ones, we can also rearrange the constraints and let the equality ones be a single matrix A such that $Ax \leq b$ where $A \in \mathbb{R}^{pxn}$ and $b \in \mathbb{R}^p$, p is the number of equality. Then problem [1] is expressed as:

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, i = 1, .., m - p \\ & Ax \leq b \end{array} \tag{2}$$

The homogeneous form of [1] is a QCQP with n + 1 variables and m + 1 constraints, where the evaluation function f_0 and left-hand side of every constraint is a quadratic form in the variable, i.e., there are no linear terms in the variable. We define

$$\tilde{P}_i = \begin{bmatrix} P_i & \frac{1}{2} q_i \\ \frac{1}{2} q_i^T & r_i \end{bmatrix} i = 0, ..., m \tag{3}$$

So the homogeneous form of [1] is given by:

$$\begin{array}{ll} \text{minimize} & z^T \tilde{P}_0 z \\ \text{subject to} & z^T \tilde{P}_i z \leq 0, i = 1, ..., m \\ & z_{n+1}^2 = 1 \end{array} \tag{4}$$

where $z \in \mathbb{R}^{n+1}$. This problem is homogeneous in the sense that scaling z by a factor of $t \in \mathbb{R}$ scales both the objective function and lefthand sides of the constraints by a factor of t^2. Then if z^* is a solution of (4), then the vector $(\frac{z_1^*}{z_{n+1}^*}, ..., \frac{z_n^*}{z_{n+1}^*}$ is the solution of (1).

In this paper, we would like to denote the optimal value in (1) to be f^*, and the optimal solution is x^* accordingly.

B. Lagrangian Relaxation

Lagrangian relaxation was studied by since 1980s, it provides a easy way for the lower bound oof f^*, can be considered as a generalization of the spectral relaxation. We can find the value of λ that has the best spectral bound of the spectral relaxation. And finally we can get the relaxation candidate x^{rl}.

Here we define,

$$\tilde{P}(\lambda) = P_0 + \sum_{i=1}^{m} \lambda_i P_i, \quad \tilde{q}(\lambda) = q_0 + \sum_{i=1}^{m} \lambda_i q_i,$$

$$\tilde{r}(\lambda) = r_0 + \sum_{i=1}^{m} \lambda_i r_i$$

where $\lambda \in \mathbb{R}_+^m$. The the Lagrangian of (1) is given by

$$\mathcal{L}(x, \lambda) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) = x^T \tilde{P}(\lambda) x + \tilde{q}(\lambda)^T x + \tilde{r}(\lambda)$$

The dual function is then

$$g(\lambda) = \inf_x \mathcal{L}(x,\lambda)$$
$$= \begin{cases} \tilde{r}(\lambda) - (1/4)\tilde{q}(\lambda)^T \tilde{P}(\lambda)^\dagger \tilde{q}(\lambda) & \text{if} \tilde{P}(\lambda) \succeq 0, \tilde{q}(\lambda) \in \mathbb{R}(\tilde{P}(\lambda)) \\ -\infty & otherwise, \end{cases}$$

where $\tilde{P}(\lambda)^\dagger$ and $\mathbb{R}(\tilde{P}(\lambda))$ denotes the Moore-Penrose pseudoninverse and range of $\tilde{P}(\lambda)$ in [1].

C. Semidefinite Relaxation

Semidefinite Relaxation, or SDR, has been studied and used in lots of common optimization problems. Lovász [5] has introduced the idea in early year 1979, but was really be popular and started developing because of [2], which applied SDR to the maximum cut problem and derived a data-independent approximation factor of 0.87856.

1) Concept of Semidefinite Relaxation: Also, the SDR and the Lagrangian dual problem are duals of each other. To derive this, by observing that

$$x^T Px = Tr(x^T Px) = Tr(Pxx^T)$$

we can use the trick of lifting since both the objective function and constraints are linear in the matrix xx^T. That adding new variable $X = xx^T$ leads problem becomes:

$$\begin{aligned} \text{minimize} \quad & F_0(X,x) = Tr(P_0 X) + q_0^T x + r_0 \\ \text{subject to} \quad & F_i(X,x) = Tr(P_i, X) + q_i^T x + r_i \leq 0, \\ & \quad i = 1, ..., m \\ & X = xx^T \end{aligned} \tag{5}$$

We now embed the original problem with n variables to a much larger space (of $n^2 + n$ dimensions. And the good thing is that we have our objective and constraints functions to be affine in X and x. However, the last constraint $X = xx^T$ is non-convex, which needs to be dealt with. We can replace it with $X \succeq xx^T$ and get a convex relaxation. SDR is given by:

$$\begin{aligned} \text{minimize} \quad & Tr(P_0 X) + q_0^T x + r_0 \\ \text{subject to} \quad & Tr(P_0 X) + q_i^T x + r_i \leq 0, \\ & i = 1, ..., m \\ & Z(X,x) = \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0 \end{aligned} \tag{6}$$

We can get the optimal value of problem 6, f^{sdr} and it is a lower bound on the optimal value f^* of original problem.

Also, we have the pseudo-code of SDR from [6]:

```
cvx_begin
    variable X(n, n) symmetric
    minimize(trace(C*X));
    subject to
        for i=1:p
            trace(A(:,:,i)*X) >= b(i);
        end
        for i=p+1:m
            trace(A(:,:,i)*X) == b(i);
        end
    X == semidefinite(n);
cvx_end
```

2) Toeplitz Quadratic Form: In Konar [4], they found a special case of SDR, where all the matrices are Toeplitz-Hermitian. There is no additional structure assumed, except for $F_0 \succeq 0$. By this, they have shown that

1) It is feasible to establish in polynomial time
2) If the problem has feasible solution, then it can be solved to global optimality in polynomial-time as well.

If the matrices are all Toeplitz, the matrices can be written as:

$$F_i = \sum_{k=-p}^{p} f_{i,k} \Theta_k \tag{7}$$

where $p = n-1, \Theta_k$ is an n x n elementary Toeplitz matrix with ones on the k^{tj} diagonal and zeros elsewhere ($k = 0 \leftrightarrow$ main diagonal, ($k > 0 \leftrightarrow$ super-diagonals, ($k < 0 \leftrightarrow$ sub-diagonals).

Then the problem becomes:

$$\begin{aligned} \min_{X,r} \quad & Re(f_0^T r) \\ \text{s.t.} & Re(f_i^T r) \leq u_i, \forall i = 1, ...m \\ & r(k) = Trace(\Theta_k X), \forall k \in \mathcal{K}_+ \\ & X \succeq 0, \\ & Rank(X) = 1 \end{aligned} \tag{8}$$

where $\mathcal{K}\mathcal{K}_+ = 0, .., p, r(k) = Trace(\Theta_k X)$. By dropping the rank constraint, we can have the following convex SDP relaxation:

$$\begin{aligned} \min_{X,r} \quad & Re(f_0^T r) \\ \text{s.t.} & Re(f_i^T r) \leq u_i, \forall i = 1, ...m \\ & r(k) = Trace(\Theta_k X), \forall k \in \mathcal{K}_+ \\ & X \succeq 0, \end{aligned} \tag{9}$$

3) Circulant Quadratic Forms: Circulant quadratic forms for QCQPs is another special form of Toeplitz Quadratic form (section II-C2). It differs because of the circulant matrices, i.e. $\{F_i\}_{i=0}^{i}$ are circulant. By using the characterist of it, we for convert the QCQP problem into a LP problem which can be solved to global optimality, at a lower complexity cost. Circulant matrices are diagonalized by the Discrete Fourier Transform (DFT) matrix, i.e. $F_i = \mathbf{F} \Lambda_m \mathbf{F}^H, where \mathbf{F} \in \mathbb{C}^{nxn} istheunitaryDFTmatrix$

$$\mathbb{F} = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega_n & \omega_n^2 & \cdots & \omega_n^{n-1} \\ 1 & \omega_n^2 & \omega_n^4 & \cdots & \omega_n^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega_n^{n-1} & \omega_n^{2(n-1)} & \cdots & \omega_n^{(n-1)(n-1)} \end{bmatrix} \tag{10}$$

where $\omega_n = e^{-j\frac{2\pi}{n}}$ is the nth root of unity, and Λ_m is the diagonal matrix of the eigenvalues of F_i obtained by taking the discrete Fourier Transform of the first row of F_i. Then each term can be converted as:

$$\begin{aligned} x^H F_i x &= x^H \mathbf{F} \Lambda_m \mathbf{F}^H x \\ &= y^H \Lambda_m y \\ &= \sum_{k=1}^{n} \lambda_i(k) |y(i)|^2 \\ &= \lambda_i^T z, i = 1, ..., m \end{aligned} \tag{11}$$

Then we move all the things together, we can get the LP problem as below:

$$\begin{aligned} \min_{z \in \mathbb{R}^n} \quad & \lambda_0^T z \\ \text{s.t.} \quad & \lambda_i^T z \leq u_i, i = 1, ..., m \\ & z \succeq 0 \end{aligned} \tag{12}$$

D. Consensus-ADMM

Huang [3] has introduced a way to solve QCQP efficiently without considering whether it is convex or not. It reformulates the problem in consensus optimization form and use alternating direction method of multipliers to solve the QCQP with each of its sub-problems with only one constraint (QCQP-1). It is also memory efficient, parallelable and has smart initialization.

Here we consider the following optimization problem

$$\underset{x}{minimize} \sum_{i=1}^{m} f_i(x) + r(x)$$

in which our objective is to minimize the sum of the cost functions $f_1, \ldots, f_m$ with other penalty function r on x. Before using ADMM, we formulate it into consensus form by introducing m auxiliary variables $z_1, \ldots, z_m$, as

$$\begin{aligned} \underset{x, \{z_i\}_{i=1}^m}{minimize} \quad & \sum_{i=1} m f_i(z_i) + r(x) \\ \text{subject to} \quad & z_i = x, \forall i = 1, \ldots, m. \end{aligned}$$

Then we can use it to update our variables by ADMM iterates:

$$x \leftarrow \underset{x}{argmin} r(x) + \rho \sum_{i=1}^{m} ||z_i - x + u_i||^2$$

$$z_i \leftarrow \underset{z_i}{argmin} f_i(z_i) + \rho ||z_i - x + u_i||^2, \forall i = 1, ..., m,$$

$$u_i \leftarrow u_i + z_i - x, \forall i = 1, \ldots, m,$$

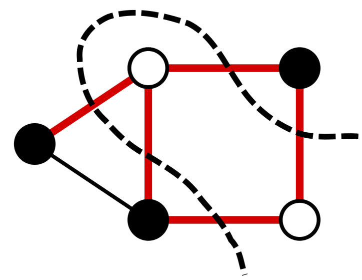

Fig. 1: An image of a galaxy

TABLE I: Result: QCQP test on different dimensions using Consensus-ADMM

Test	Upper bound	CD		ADMM	
		f_0	violation	f_0	violation
n=4	1.000	1.000	0.000	1.000	0.537
n=10	6.250	6.000	0.000	4.244	0.932
n=20	27.652	24.000	0.000	21.412	0.972
n=50	176.483	164.002	0.000	138.819	0.994
n=100	640.973	604.681	0.005	552.965	0.998

III. Results

A. Maximum cut problem

In graph theory, a cut is a partition of the vertices of a graph into two disjoint subsets [8]. On top of that, maximum cut is a cut whose size is at least the size of any other cut [9].

Here we define a cut as:

$$C = (S, T) = (u, v) \in E | u \in S, v \in T$$

where C is a partition of V of a graph $G = (V, E)$ into two subsets S and T.

And in this project, we use the package of QCQP of CVXPy [7], and use it to solve the maximum cut problem directly.

We generate the data using the random function of numpy. And then create the edge connection matrix M.

As we can see in table I, we have tested the QCQP solver on generated data with different dimensions. We compared the ADMM method with the coordinate descent algorithm. And in table II, we compared different ρ to see the effect of how will this hyper parameter affect the solution.

TABLE II: Result: QCQP test on different ρ using Consensus-ADMM

Test	Upper bound	CD		ADMM	
		f_0	violation	f_0	violation
n=10, ρ=10				3.211	0.930
ρ=14.3 (auto)	5.000	5.000	0.000	4.244	0.932
ρ=20				4.124	0.930s
ρ=9.252 (auto)				25.639	0.976
n=20, ρ=10	36.635	34.001	0.000	28.366	0.977
ρ=20				29.084	0.977

REFERENCES

[1] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. USA: Cambridge University Press, 2004. ISBN: 0521833787.

[2] Michel X Goemans and David P Williamson. “Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming”. In: *Journal of the ACM (JACM)* 42.6 (1995), pp. 1115–1145.

[3] K. Huang and N. D. Sidiropoulos. “Consensus-ADMM for General Quadratically Constrained Quadratic Programming”. In: *IEEE Transactions on Signal Processing* 64.20 (2016), pp. 5297–5310.

[4] A. Konar and N. D. Sidiropoulos. “Hidden Convexity in QCQP with Toeplitz-Hermitian Quadratics”. In: *IEEE Signal Processing Letters* 22.10 (2015), pp. 1623–1627.

[5] L. Lovasz. “On the Shannon capacity of a graph”. In: *IEEE Transactions on Information Theory* 25.1 (1979), pp. 1–7.

[6] Zhi-Quan Luo et al. “Semidefinite relaxation of quadratic optimization problems”. In: *IEEE Signal Processing Magazine* 27.3 (2010), pp. 20–34.

[7] Jaehyun Park. *General Heuristics for Nonconvex Quadratically Constrained Quadratic Programming*. 2017. URL: https://stanford.edu/~boyd/papers/qcqp.html.

[8] Wikipedia contributors. *Cut (graph theory) — Wikipedia, The Free Encyclopedia*. [Online; accessed 9-June-2020]. 2019. URL: https://en.wikipedia.org/w/index.php?title=Cut_(graph_theory)&oldid=931302747.

[9] Wikipedia contributors. *Maximum cut — Wikipedia, The Free Encyclopedia*. [Online; accessed 9-June-2020]. 2020. URL: https://en.wikipedia.org/w/index.php?title=Maximum_cut&oldid=946550608.