Fast and Low Complexity Algorithms for Compressed Training Based Massive MIMO

by

Ender Erkaya

A Dissertation Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Electrical and Electronics Engineering

KOÇ
UNIVERSITY

September 12, 2019

Fast and Low Complexity Algorithms for Compressed Training Based Massive MIMO

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Ender Erkaya

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

____

Prof. Alper T. Erdogan

____

Prof. Ozgur B. Akan

____

Prof. Fatih Alagoz

Date: ______________________________

Dedicated to my beloved family:

Ezgi, Nalan, Mustafa

ABSTRACT

The power of ℓ_∞-norm as a convex cost function in blind equalization arises from the property that minimizing ℓ_∞-norm of the output selects the sparse solution for the combined channel-equalizer impulse response. Compressed Training Adaptive Equalization approach proposed in [Yilmaz and Erdogan, 2016] combines the power of ℓ_∞-norm with the supervised channel equalization technique in order to reduce the required training length. [Yilmaz and Erdogan, 2019] proves that for Massive MIMO systems, the training symbols can be compressed about $log_2 K$, where K is the number of users.

This thesis investigates fast and low complexity solutions for Compressed Training Based Massive MIMO. For the purpose, the direct derivation of the proximal operator for ℓ_∞-norm is presented firstly and its clipping property is shown. The proximal methods are investigated for the solutions of ℓ_∞-norm regularized least-square problems. Specifically, Alternating Direction Method of Multipliers is proposed for the solution of Compressed Training Based Massive MIMO Noiseless Setting. Besides, an accelerated alternating projection based algorithm is proposed for solution of the Noisy Setting of Compressed Training Based Massive MIMO. Moreover, for further applications, the proximal operator of the K-norm is derived and its relation with ℓ_∞-norm and ℓ_1-norm is analyzed.

ÖZETÇE

Kör denkleştirmede dışbükey bir maliyet fonksiyonu olarak ℓ_∞ normun gücü, çıktının ℓ_∞-normunu minimize etmenin, kombine kanal-denkleştirici dürtü yanıtı icin seyrek bir çözüm seçmesinden kaynaklanır. Şıkıştırılmış Öğrenme Uyarlamalı Denkleştirme yaklaşımı, ℓ_∞ normu ve kare maliyet fonksiyonlarını kullanarak, sıkıştırılmış algılama yaklaşımını ISI kanal denkleştirme problemine bağlar. [Yilmaz and Erdogan, 2017], Masif MIMO sistemleri icin, öğrenme sembollerinin $log_2 K$ civarında sıkıştırılabileceğini kanıtlar, ki burada K kullanıcı sayısıdır. Bu tezde, Sıkıştırılmış Öğrenme Tabanli Masif MIMO için hızlı ve düşük karmaşıklıklı çözümler incelenmiştir. Bu amacla, ℓ_∞-normun proksimal operatörünün dogrudan türetilmesi sunuldu ve kırpma özelligi gösterildi. Daha sonra, ℓ_∞ -norm ile duzenlenmiş kare maliyet fonksiyonlu optimizasyon problemleri için proksimal algoritmalar araştırılır. Spesifik olarak, Sıkıştırılmış Öğrenme Tabanli Masif MIMO Gürültüsüz Optimizasyon Problemi icin ADMM yontemi sunulur. Ayrica, Sıkıştırılmış Ögrenme Tabanli Masif MIMO Gürültülü Ayar icin hızlandırılmış bir projeksiyon tabanlı algoritma onerilmistir. Ileriki araştrmalar için, K normun proksimal operatörü turetilmiştir. Ayrica, K-norm ile ℓ_∞ -norm ve ℓ_1 -normun proksimal operatörleri arasındakı ilişki analiz edilir.

ACKNOWLEDGMENTS

This dissertation is the result of a long work accompanied with personal challenges as well as academic challenges.

First of all, I would like to express my deepes gratitude to my thesis advisor Prof. Alper T. Erdogan. Providing a creative flexibility, with his strong tolerance to my downs, I always felt his support whenever I needed.

I would also like to thank my thesis committee member Prof. Ozgur B. Akan who strongly influences us with his research and teaching.

I would also like to acknowledge my committee member Prof. Fatih Alagoz from Bogazici University. I am grateful for his very valuable comments on this thesis.

This thesis wouldn't have been realized without the priceless moral and material support of my family. I sometimes could not perceive how big their support is.

I would like to thank my friends Ulas Ozipek and Ibrahim Yazar whose glances at my thesis are as a drop in the bucket comparing their invaluable support and friendship throughout this long journey.

I would like to acknowledge Meltem Civas, whom our path intersected in Koc University after Bilkent University. I appreciate her valuable comments and encouragement.

I would like to thank Muhammad Alipour and Toghrul Almammadov. Through my Koc University years, our friendship became stronger and I would like to thank for making these years more valuable.

I would like to acknowledge my friends Cagri Aydinel and Selcuk Turkmen. Without their friendship, this journey of life would become less meaningful.

There are many people whose encouragement and friendship has became very valuable for me. I thank you all and please forgive me if I forgot to name you.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

As computational capability increases, data and information that needs to be processed and transported increases. While a huge amount of data and information is produced, the practical needs such as time and energy become more valuable. Therefore, the importance of practical capability of data processing algorithms has risen. This has brought more research on finding fast and low complexity algorithms for many useful applications. As available algorithms vary, finding the best algorithm tailored for the specific application increases its importance in mathematical algorithms research.

Among mathematical optimization algorithms, convexity has become the key point instead of linearity due to the increase of available nonlinear algorithms. As polynomial time algorithms become available for nonlinear optimization problems, unique global solution property has brought the importance of convexity. Almost all convex problems are shown to be solvable in polynomial time and the significant part of them is solvable via many optimization algorithms.

Although convexity handles many problems of optimization algorithms, non-differentiability causes problems and eliminates some widely used methods such as gradient descent, Newton-Raphson method. Hence, more research is focused on non-differentiable optimization problems since they appear much in real-life problems. Specifically, compressed sensing approach proves the successful recovery of the data that has known to be unrecoverable [Candès, 2008]. The approach states that if incoherent sampling waveform is used for sampling a sparse signal, then the signal can be recovered even if it is sampled much less than Nyquist-Shannon criterion. The compressed sensing is later shown to be solvable in a convex optimization framework using nondifferentiable ℓ_1-norm that selects sparse solution.

The magic of the ℓ_1-norm is brought by its capability of choosing sparse solutions between many solutions. Since, its use for extracting sparsity become widespread, ℓ_1-norm based optimization frameworks become the focus of the non-differentiable convex optimization algorithms research. ℓ_1-norm based optimization problems are efficiently solvable via proximal methods due to the cheap proximity operator of ℓ_1-norm.

Through being the dual norm of ℓ_1-norm, ℓ_∞-norm has useful properties. On the contrary to ℓ_1-norm, it has anti-sparsity property which enables to select anti-sparse solution. Its limiting property make ℓ_∞- norm as a useful tool for Peak-to-Average power ratio in multicarrier communication systems(OFDM/DMT). In 1994, ℓ_∞-norm is shown to be a key cost function for blind equalization problems.[Vembu et al., 1994] The power of ℓ_∞-norm as a convex cost function in blind equalization arises from the property that minimizing ℓ_∞-norm of the output selects the sparse solution for the combined channel-equalizer impulse response. [Ding and Luo, 2000],[Luo et al., 2002] and [Erdogan and Kizilkale, 2005] use ℓ_∞-norm cost function in blind equalization.

In Inter-Symbol-Interference(ISI) problems, there were two main approaches: Blind Equalization and training based supervised channel equalization. Blind Equalization aims to directly equalizing the channel without utilizing any information about channel using training symbol. On the other hand, training based supervised channel equalization uses training symbols to estimate and equalize the channel. When only training symbols are used, the number of training length required for reliable channel estimation equals to channel length in single input single output(SISO) systems [Yilmaz and Erdogan, 2017]. In massive multiple input multiple output(MIMO) systems, the number of required training length equals to the number of users without making any sparsity assumption about channel matrix, if only training information is used [Yilmaz and Erdogan, 2019].

Recently, [Yilmaz and Erdogan, 2019] combined the unsupervised(blind) and supervised(training based) channel equalization to make reliable channel equalization minimizing the required training symbol length. They showed that for Massive MIMO systems, the training symbols can be compressed about $log_2 K$, where K is the number of users. The approach combines ℓ_∞-norm as cost function which sparsifies the combined channel and least-square cost function that learns the channel information using training symbols. This semi-blind approach enables to connect the compressive sensing approach with ISI channel equalization problem.

In this thesis, algorithms for Compressed Training Based Massive MIMO are investigated and derived. Fast and low complexity algorithms implementing Compressed Training Approach are aimed. This requires the connection of the convex optimization algorithms with Compressed Training Optimization settings. The Compressed Training Optimization settings involve composite cost function that combines ℓ_∞-norm and ℓ_2-norm.

For the purpose of obtaining fast and low complexity algorithms, the proximal operator of ℓ_∞-norm is derived firstly. Obtaining the proximal operator for ℓ_∞-norm is very important for the application of proximal optimization algorithms and splitting methods since the proximal or proximity operator is the kernel of the proximal based optimization algorithms. While obtaining the proximal operator of ℓ_∞-norm, we investigate the connections between its dual norm, ℓ_1-norm, and its projection and proximity operators. The derivation and result explicity show that the proximity operation of ℓ_∞-norm is a clipping operation where a clipping threshold needs to be calculated.

Secondly, for both Noiseless Case Compressed Training Optimization Setting and Full Noise Consideration-Compressed Training Optimization Setting, the available optimization algorithms are investigated in this thesis. For Noiseless Case Compressed Training Optimization Setting, we propose Alternating Direction Method of Multipliers using ℓ_∞-norm proximal operator. For Noisy Case Compressed Training Setting, we develop an alternating projection approach and derive the required projection operators for the algorithm.

Additionally, proximal operator for the K-norm is presented for the future applications that use the K-norm as a convex cost function. As K-norm can be regarded as a generalization of the both ℓ_1 and ℓ_∞-norm, we show that the proximal operator for the K-norm is hybrid proximal operator. This hybrid operator applies soft-thresholding and clipping operation at the same time. The soft-thresholding operation is linked with the proximity operator of ℓ_1-norm and the clipping operation is linked with the proximal operator of the ℓ_∞-norm. The statistical properties of K-norm need further investigation.

1.1 Contributions

The contributions of this thesis are summarized below:

- The direct derivation of proximity operator of ℓ_∞-norm is presented and the clipping behavior is explicitly shown.

- Alternating Direction Method of Multiplier ,using ℓ_∞-norm proximal operator, is proposed for the solution of the Noiseless Setting of Compressed Training Based Massive MIMO.

- Accelerated Alternating Projection Method is proposed for the solution of the Noisy Setting of Compressed Training Based Massive MIMO.

- The direct derivation of the proximity operator of K-norm is given. Its relation to ℓ_∞-norm and ℓ_1-norm is shown.

1.2 Organization

The thesis is organized as follows: Section 2 presents the direct derivation of the proximal operator of ℓ_∞-norm as well as example application in blind equalization. Section 3 investigates the possible fast and low complexity algorithms for the Noiseless Setting of the Compressed Training Based Massive MIMO. We develop and present fast and low complexity algorithms for Noisy Setting of MIMO CoTA in Section 4. Section 5 presents the derivation of the proximal operator for the K-norm and its connection to ℓ_1-norm and ℓ_∞-norm.

Chapter 2

A DIRECT DERIVATION OF ℓ_∞ NORM PROXIMAL OPERATOR

2.1 Introduction

Optimization settings featuring ℓ_∞-norm appear in various application areas: they naturally arise in min-max criterion based approaches such as frequency sampling based filter design and minimum sidelobe beamforming. Peak-to-average power ratio (PAPR) minimization for multicarrier (OFDM/DMT) communication systems is also a prototype ℓ_∞-norm minimization problem.

Inverse problems is another area where ℓ_∞-norm has been proven to be useful. Based on its duality with ℓ_1-norm , the minimization of the ℓ_∞-norm of the output of an LTI system, leads to sparsification of the corresponding impulse response, under some constraints on the inputs and inverse system parameters. This feature was successfully exploited in blind equalization domain to derive convex cost function based adaptive algorithms [Vembu et al., 1994, Ding and Luo, 2000, Luo et al., 2002, Erdogan and Kizilkale, 2005], and later extended to Blind Source Separation (BSS) domain leading the development of Bounded Component Analysis algorithms [Erdogan, 2006b, Erdogan, 2013]. More recently, the *compressed training* approach has been proposed which effectively uses ℓ_∞-norm as a regularizer for the least-squares based training setup leading to logarithmic decrease in the training length requirements of the communication systems [Yilmaz and Erdogan, 2016, Yilmaz and Erdogan, 2017].

As the use of ℓ_∞-norm has spread to wide range of applications, the need for fast and low complexity algorithms to solve such problems has also become critical.

From this perspective, the convexity of ℓ_∞ provides us with the opportunity to search among wide range algorithms studied in the convex optimization area for several decades. However, its nondifferentiability eliminates some algorithms such as gradient descent and Newton's method. Although most problems involving ℓ_∞ can be cast as a second order cone programming and solved via interior point methods, large data sizes and dense matrices annihilate advantages of the complex interior point algorithms that diminishes them as a fast and good option [Beck and Teboulle, 2009]. As an explicit approach, subgradient based methods can be used to solve ℓ_∞-norm based optimization problems by handling the nondifferentiability of the ℓ_∞-norm. However, slow convergence of subgradient methods push us to search for faster algorithms.

In the recent literature, many fast and low complexity algorithms based on proximal operators have been proposed for convex and non-differentiable problems, and successfully applied to especially ℓ_1-norm based cost functions arising in signal processing and machine learning applications. In order to adapt these algorithms to ℓ_∞ based optimization problem schemes, we primarily need to get a fast, low complexity and easy to implement proximal operator of the function $f = \|.\|_\infty$.

The ℓ_∞-norm operator can be derived based on Moreau identity[Parikh et al., 2014] and considering the dual problem of projection to ℓ_1-norm ball. In reference [Duchi et al., 2008], an efficient algorithm for projection to ℓ_1-norm ball is proposed, which is derived based on the projection to unit-simplex in [Shalev-Shwartz and Singer, 2006]. In [Goldstein et al., 2013], the use of the ℓ_∞-norm proximal operator based on this approach was proposed. The main purpose of this article is to obtain the proximal operator for the ℓ_∞-norm using the optimality condition for the ℓ_∞-proximal operator cost function. We show that the ℓ_∞-norm proximal operator is simply equal to the clipping function. The proposed derivation is more direct and reflects the clipping function behavior in a clear manner. Furthermore, this derivation approach can be easily generalized to k-peak function, i.e. the sum of the k largest magnitude entries.

The article is organized as follows: in Section 5.2, we first remind the basic definition of the proximal operator. Section 2.3 is the main section, where the proximal operator for ℓ_∞-norm is derived. Finally, in Section 2.4, we provide a numerical example to illustrate the use of the proposed proximal operator.

2.2 Proximal Operators

The proximal operator for the convex function $f(\mathbf{x})$, with $\mathbf{x} \in \mathbb{R}^n$ is defined as:

$\text{prox}_{\lambda f}(\mathbf{v}) = \arg\min_{\mathbf{X}} \; f(\mathbf{x}) + \frac{1}{2\lambda} \|\mathbf{x} - \mathbf{v}\|_2^2$.(2.1)$\mathbf{x}^*$ is the solution to the above optimization problem, output of the proximal operator, if and only if the optimality condition

$$\mathbf{0} \in \lambda \partial f(\mathbf{x}^*) + (\mathbf{x}^* - \mathbf{v}), \tag{2.2}$$

holds. Furthermore, $\mathbf{x}^*$ is the unique minimizer of the problem due to strong convexity coming from the quadratic part, $\|\mathbf{x} - \mathbf{v}\|_2^2$.

Rearranging the terms, the optimal condition turns to $\mathbf{x}^* = \mathbf{v} - \lambda \mathbf{s}$ where $\mathbf{s} \in \partial f(\mathbf{x}^*)$. In other words, we are essentially applying a subgradient algorithm step with stepsize λ where the subgradient is evaluated at output $\mathbf{x}^*$ on the contrary to the conventional subgradient method which uses the current subgradient information. Due to use of future(next step) subgradient information, proximal operator is considered as backward or implicit method in opposition to explicitness of (sub)gradient based methods [Goldstein et al., 2014].

2.3 ℓ_∞-norm Proximal Operator

The following theorem shows that the proximal operator of the ℓ_∞-norm is equivalent to elementwise (absolute) clipping operation, where the clipping level is determined by λ:

Theorem: Consider the proximal operator $prox_{\lambda f}$ where $f = \|.\|_\infty$. The solution $\mathbf{x}^*$ to the problem

$$prox_{\lambda f}(\mathbf{v}) = \arg\min_{\mathbf{x}} \lambda \|\mathbf{x}\|_\infty + (1/2)\|\mathbf{x} - \mathbf{v}\|_2^2$$

is given by the elementwise clipping operator

$$\mathbf{x}^* \;=\; C_\mu(\mathbf{v}) = \begin{bmatrix} c_\mu(v_1) & c_\mu(v_2) & \dots & c_\mu(v_n) \end{bmatrix}^T$$

where

- $c_\mu(\cdot)$ is the scalar clipping function given in Fig. 7.1, which can be written as

$$c_\mu(q) = \begin{cases} -\mu & q < -\mu, \\ q & |q| \le \mu, \\ \mu & q > \mu. \end{cases} \tag{2.3}$$

- $\mu = \frac{1}{L}\{\mathbf{1}_L{}^T \mathbf{v}^{(s)} - \lambda\}^+$ is the clipping level,

- $\mathbf{v}^{(s)}$ is the vector obtained by taking the absolute values of the elements of $\mathbf{v}$ and sorting them in a descending order,

- $\mathbf{1}_L \in \mathbb{R}^n$ is the partial ones vector defined as $1_L = [\underbrace{1, 1....., 1, 1}_{\text{L}} 0, 0....0, 0]^T$,

- $\{\mathbf{u}\}^+ = max(\mathbf{u}, \mathbf{0})$, is the elementwise Rectified Linear Unit (ReLU) operator,

- L is the the number of clipped elements. It can be determined as the smallest index with a nonnegative element of the vector $\mathbf{Av}^{(s)} - \lambda\mathbf{1}$ if $\mathbf{Av}^{(s)} - \lambda\mathbf{1} \notin \mathbb{R}_-^n$, otherwise $L = n$. Here, $\mathbf{A} = [A_{ij}] \in \mathbb{R}^{n\times n}$ is the index checking matrix defined by

$$A_{ij} = \begin{cases} 0 & i < j - 1, \\ -\text{i} & i = j - 1, \\ 1 & i \ge j. \end{cases}$$

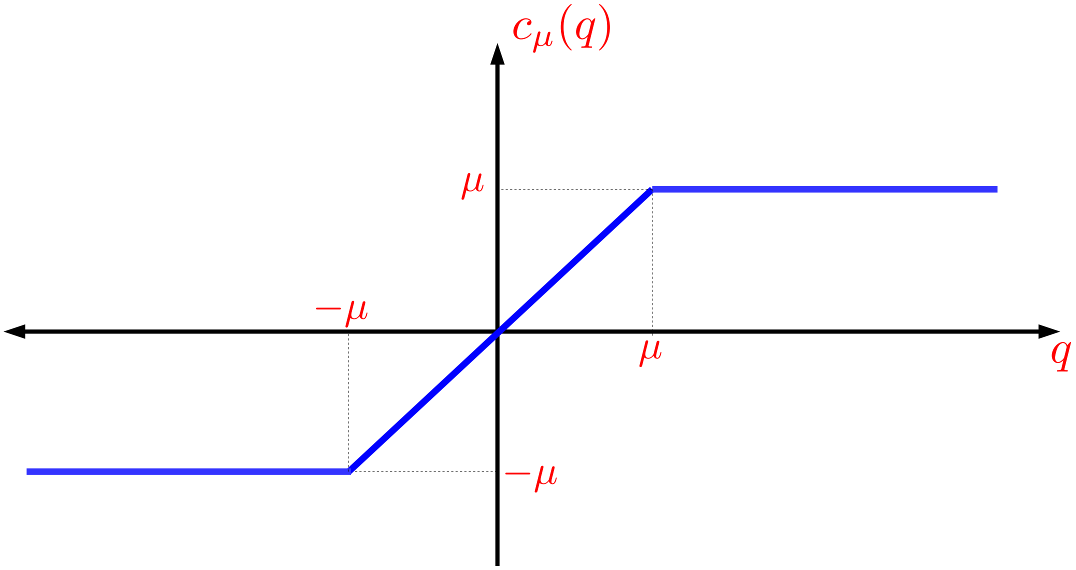

Figure 2.1: Scalar clipping function.

Proof: $\mathbf{x}^*$ is the solution, if it satisfies the optimality condition in (2.2) where the subdifferential set of f at $\mathbf{x} \in \mathbb{R}^n - \mathbf{0}$ is given by

$$\begin{aligned}\partial f(\mathbf{x}) = \{\mathbf{s} \in \mathbb{R}^n \mid \quad & \mathbf{s} = \sum_{j \in I} \beta_j sign(x_j)\mathbf{e}_j \ \ ; \\ & \sum_{j \in I} \beta_j = 1, \beta_j \in \mathbb{R}_+\}\end{aligned}$$

where I is the index set of absolute maximum elements defined as $I = \{j \mid |x_j| = \|\mathbf{x}\|_\infty \ , j \in \{1, 2, ..n\}\}$ and $\mathbf{e}_j$ is the standard basis vector in $\mathbb{R}^n$ with all elements equal to zero except the j^{th} element is equal to 1. When $\mathbf{x} = \mathbf{0}$,

$$\partial f(\mathbf{0}) = \{\mathbf{s} \in \mathbb{R}^n \mid \|\mathbf{s}\|_1 \leq 1\},$$

i.e., it is equal to the ℓ_1 unity norm-ball.

If $\mathbf{x}^* = 0$, the optimality condition in (2.2) implies that there exists $\mathbf{s}^* \in \mathbb{R}^n$ with $\|\mathbf{s}^*\|_1 \leq 1$, such that

$$\mathbf{0} = \lambda \mathbf{s}^* - \mathbf{v} \Rightarrow \mathbf{v} = \lambda \mathbf{s}^*$$

As a result, this case holds only when

$$\|\mathbf{v}\|_1 = \sum_{k=1}^{n} |v_k| \leq \lambda. \tag{2.4}$$

If $\mathbf{x}^* \neq 0$, focusing on (2.2), for the components with index $j \notin I$, the optimality condition simplifies to

$$0 = x_j^* - v_j \Rightarrow x_j^* = v_j, \tag{2.5}$$

i.e., proximal operator output component is equal to the corresponding input for $j \notin I$.

For the components with index $j \in I$, the optimality condition in (2.2) implies that there exists a β_j^* for which

$$x_j^* - v_j + \lambda sign(x_j)\beta_j^* = 0,$$

or equivalently,

$$x_j^* = v_j - \lambda sign(x_j^*)\beta_j^*. \tag{2.6}$$

Multiplying both sides with $sign(x_j^*)$ yields:

$$|x_j^*| = sign(x_j^*)v_j - \lambda\beta_j^*.$$

Now, we observe that $sign(x_k^*)$ must be equal to $sign(v_k)$. Because, if x_k^* had opposite sign, that would cause extra cost on the objective function compared to the equal signed version due to the quadratic term $\|\mathbf{x} - \mathbf{v}\|_2^2$. In other words, there always exists an equal signed version of x_k^* which yields same cost on $\|\mathbf{x}\|_\infty$ and less cost on $\|\mathbf{x} - \mathbf{v}\|_2^2$ relative to the opposite signed version. Therefore, replacing $sign(x_k^*)$ with $sign(v_k)$, we obtain

$$|x_j^*| = |v_j| - \lambda\beta_j^*. \tag{2.7}$$

Based on the definition of the index set I, $\forall k, l \in I; |x_k^*| = |x_l^*|$, and therefore, we have:

$$|v_k| - \lambda\beta_k^* = |v_l| - \lambda\beta_l^*$$

which leads to

$$\beta_k^* = \beta_l^* + \frac{|v_k| - |v_l|}{\lambda}.$$

Summing the equations for $\forall l \in I$, and replacing $\sum\limits_{l \in I} \beta_l^* = 1$, we obtain

$$L\beta_k^* = 1 + \frac{L|v_k| - \sum\limits_{l \in I} |v_l|}{\lambda},$$

where $L = card(I)$. Dividing by L, we get

$$\beta_k^* = \frac{|v_k| - \overline{v}}{\lambda} + \frac{1}{L},$$

where $\overline{v} = \frac{1}{L} \sum\limits_{l \in I} |v_l|$. Inserting this expression in (2.2), we obtain

$$x_k^* = sign(v_k)(\overline{v} - \frac{\lambda}{L}). \tag{2.8}$$

The relation $sign(x_k^*) = sign(v_k)$ requires

$$\overline{v} > \frac{\lambda}{L}. \tag{2.9}$$

Equation (2.8) implies all the components corresponding to the index set I, are set to the same magnitude level $\mu = \overline{v} - \frac{\lambda}{L}$.

Moreover, $\forall j \notin I, k \in I$, we can write

$$|x_j^*| < |x_k^*|, \tag{2.10}$$

$$|v_j| < (\overline{v} - \frac{\lambda}{L}) \leq |v_k|, \tag{2.11}$$

where the inequality on the right is obtained from (2.7). Therefore, based on (2.11), we can conclude that index set I corresponds to the indexes of the L largest magnitude elements of $\mathbf{v}$. In effect, this fact together with equalities (2.5) and (2.8) imply that the L largest magnitude elements of $\mathbf{v}$ are clipped to the same level

$$\mu = \overline{v} - \frac{\lambda}{L}. \tag{2.12}$$

by the clipping function in (2.3) (and shown in Fig. 2.1). Therefore, $x_k = c_\mu(v_k)$ (or $\mathbf{x} = C_\mu(\mathbf{v})$).

As a result, what remains is to identify L, i.e., the cardinality of I, which also identifies the clipping level μ. For this purpose, we first define the vector $\mathbf{v}^{(s)} \in \mathbb{R}^n$ which is obtained by taking the absolute values of the elements of $\mathbf{v}$, and sorting them in a descending manner, i.e., $v_1^{(s)} \geq v_2^{(s)} \geq \ldots v_n^{(s)}$. In terms of $\mathbf{v}^{(s)}$, the inequalities in (2.11) can be reorganized as

$$\sum_{k=1}^{L-1} v_k^{(s)} - (L-1)v_L^{(s)} - \lambda \leq 0 \text{ if } 2 \leq L \leq n \tag{2.13}$$

$$\sum_{k=1}^{L} v_k^{(s)} - Lv_{L+1}^{(s)} - \lambda > 0 \text{ if } 1 \leq L \leq n-1. \tag{2.14}$$

We consider the following cases for L:

- If $L = 1$, then (2.14) implies

$$v_1^{(s)} - v_2^{(s)} - \lambda > 0. \tag{2.15}$$

- If $L \in \{2, \ldots, n-1\}$, we can rewrite the conditions in (2.13-2.14) as

$$\begin{bmatrix} 1 & .. & 1 & -L+1 & 0 & 0 & .. & 0 \\ 1 & .. & 1 & 1 & -L & 0 & .. & 0 \end{bmatrix} \mathbf{v}^{(s)} - \lambda\mathbf{1} \begin{matrix} \leq \\ > \end{matrix} 0. \tag{2.16}$$

- If $L = n$, the inequality in (2.13) implies

$$\sum_{k=1}^{n-1} v_k^{(s)} - (n-1)v_n^{(s)} - \lambda \leq 0, \tag{2.17}$$

  and the inequality in (2.9) implies

$$\sum_{k=1}^{n} v_k^{(s)} - \lambda > 0. \tag{2.18}$$

In order to establish the choice of L, based on the inequalities in (2.15-2.18), we define

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 1 & -2 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 & \dots & 1 & -(n-1) \\ 1 & 1 & 1 & 1 & \dots & 1 & 1 \end{bmatrix}$$

and $\mathbf{z} = \mathbf{A}\mathbf{v}^{(s)} - \lambda\mathbf{1}$.

Regarding $\mathbf{z} \in \mathbb{R}^n$, we make the following useful observation: $\mathbf{z}$ has non-decreasing elements (with increasing index). This can be demonstrated by checking

$$z_{i+1} - z_i = \begin{cases} (i+1)(v_i^{(s)} - v_{i+1}^{(s)}), & i \leq n-1, \\ nv_n^{(s)} & i = n, \end{cases} \tag{2.19}$$

which is non-negative as $\mathbf{v}^{(s)}$ is non-increasing with non-negative elements. Based on the definition of $\mathbf{z}$ and its monotonic property, we can list the following three mutually exclusive cases to identify L:

i. The sign switching ($z_{k-1} \leq 0$ and $z_k > 0$ for $k \in \{2, \dots, n\}$): Based on the inequalities (2.16-2.18), there is a sign switching (from non-positive to positive) $\mathbf{z}$ when L is in $\{2, \dots, n\}$. In this case identifies L as the smallest index of $\mathbf{z}$ with positive value.

ii. All elements are positive $\mathbf{z} > \mathbf{0}$: This case identifies L as equal to 1. If $L = 1$, due to (2.15), $z_1 > 0$ and therefore $\mathbf{z} > \mathbf{0}$ due to its non-decreasing property.

iii. All elements are non-positive ($\mathbf{z} \leq \mathbf{0}$): This case implies $z_n \leq 0$, and therefore, $\|\mathbf{v}\| \leq \lambda$. By the inequality (2.4), this identifies the case $\mathbf{x}^* = 0$. Note that this case corresponds to clipping all elements (and therefore, $L = n$) with level $\mu = 0$.

We note that the clipping level formula in (2.12), which is derived for $\mathbf{x}^* \neq 0$, can be modified as

$$\mu = \{\overline{v} - \frac{\lambda}{L}\}^+ = \frac{1}{L}\{\mathbf{1}_L{}^T\mathbf{v}^{(s)} - \lambda\}^+, \tag{2.20}$$

so that the case iii, i.e., $\mathbf{x}^* = 0$, can be covered with this formula as the argument of the rectified linear mapping would be negative which would correspond to 0 clipping level ■

The algorithm implementation for the proximal operator consists of three main steps:

1. Sorting $\mathbf{v}$ to obtain $\mathbf{v}^{(s)}$,

2. Identifying L and μ, and,

3. Clipping operation with the computed μ.

The main computational complexity is due to sorting in the first step (which requires $O(nlog(n))$ operations) and the second step, where the computation of $\mathbf{z}$ can be greatly simplified (to $O(n)$ operations) by using the recursion in (2.19).

As a final note, the above proximal operator derived for real vectors can be easily extended to the complex vectors. Skipping the derivation due to length constraints, only difference is essentially that the clipping expression (2.8) is replaced with

$$x_k^* = e^{j\theta(v_k)}(\overline{v} - \frac{\lambda}{L}). \tag{2.21}$$

where $v_k = |v_k|e^{j\theta(v_k)}$, and therefore, the complex version of the elementwise clipping function is equal to

$$c_\mu(x_k) = \begin{cases} v_k & |v| \leq \mu, \\ \mu e^{j\theta(v_k)} & \text{otherwise.} \end{cases} \tag{2.22}$$

2.4 Numerical Example

As an example to illustrate the benefit of ℓ_∞-norm operator, we consider the ℓ_∞-norm based blind equalization problem [Vembu et al., 1994]. In the corresponding setting, we assume $4QAM$ signals are sent through an LTI channel whose output is corrupted by Gaussian noise. In the blind equalization, the purpose is to invert the channel adaptively, using only measurements at the receiver, without any training informations. In the convex blind equalization approach of [Vembu et al., 1994], the ℓ_∞-norm of the equalizer output is minimized while one of the taps of the equalizer is fixed as constant. The corresponding optimization problem can be formulated as the minimization of $\|\mathbf{Y}\mathbf{w} + \mathbf{y}\|_\infty$, where $\mathbf{w}$ is the non-fixed components of the equalizer filter, $\mathbf{Y}$ and $\mathbf{y}$ are the is the sub-matrices of the convolution matrix containing measurements corresponding to free and fixed parts of the equalizer impulse response.

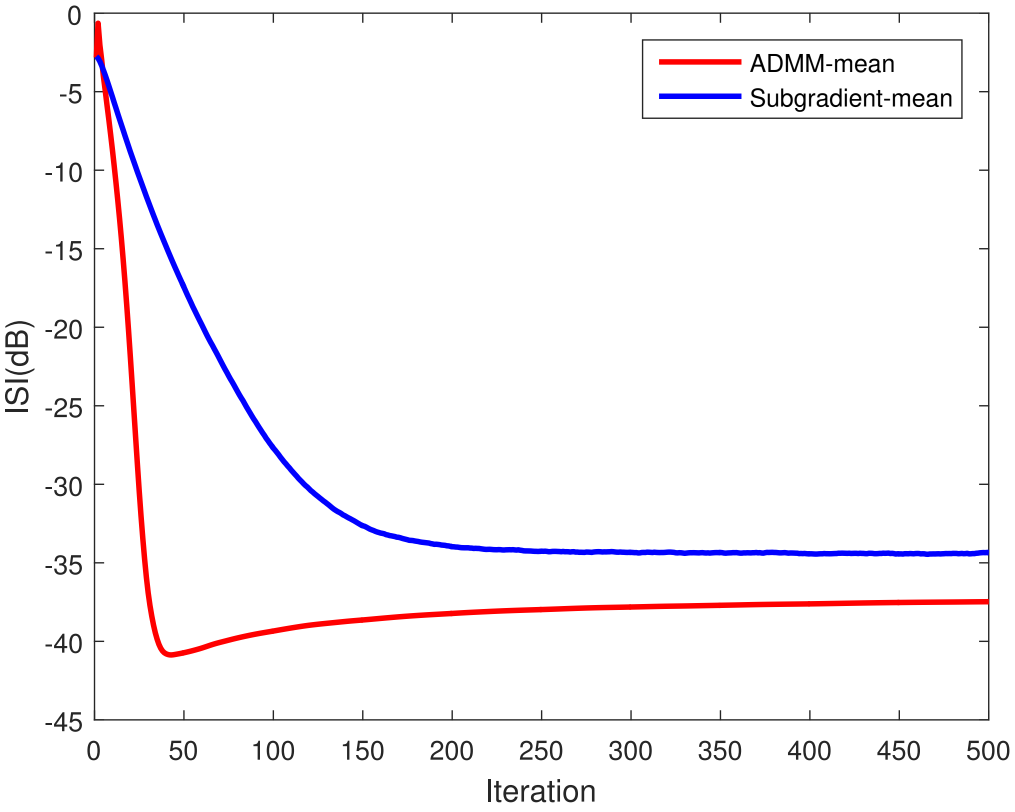

Figure 2.2: ISI(dB) ADMM vs Subgradient

We consider the 4-tap complex channel

$$\mathbf{h} = \begin{bmatrix} -1.0493 + 0.2305i \\ 1.4129 - 1.4497i \\ -0.2540 + 0.2021i \\ 0.5302 - 0.7732i \end{bmatrix}, \tag{2.23}$$

in [Proakis, 2000]. In Fig. 2.2, we compare average ISI values (the ratio of the residual ISI energy to signal energy) per iteration for Subgradient [Erdogan and Kizilkale, 2005] and ADMM [Boyd et al., 2011, Parikh et al., 2014] using the ℓ_∞-norm proximal operator, which shows that the proximal operator clearly improves the convergence behavior in terms of both target level and iteration count.

Chapter 3

ℓ_∞ NORM REGULARIZED LEAST-SQUARE PROBLEM AND COMPRESSED TRAINING BASED MASSIVE MIMO

3.1 Introduction

Through the development of computational capability and the algorithms, nonlinear optimization based problem solving have arisen in many engineering problems [Combettes and Pesquet, 2011] such as signal reconstruction, compressed sensing, blind equalization, optimal control[Boyd et al., 1994], machine learning [Goodfellow et al., 2016, Alpaydin, 2014]. As nonexistence of local minimum points and existence of many algorithms and analysis canonize convex optimization in nonlinear optimization. After the foundation of interior point algorithms, many convex optimization problems are solvable in polynomial time. But, the complexity and convergence of the solution become important due to practicability, real-time applicability, power and memory resource, etc. The choice of the right algorithm, that converges fast and costs low, make the solution valuable for the real time applications. The use of ℓ_∞-norm as a convex cost function in blind equalization lay the foundations for the use of convex optimization in equalization problems [Vembu et al., 1994]. As the solutions develop, recently proposed Compressed Training approach use the composite convex cost functions, that is defined as the sum of two or more convex cost functions. Specifically, ℓ_∞-norm representing the bounded-ness of the equalizer output and sparseness of the combined channel response and ℓ_2 norm that results from the least-square fit of the training. In this chapter, we investigate the algorithms for an ℓ_∞-norm regularized least-square problem given in Setting 3.1, as well as examples for Compressed Training Adaptive Equalization.

Setting 3.1

$$\underset{x}{\text{minimize}} \quad \frac{1}{2}\left\|Ax-b\right\|_2^2+\lambda\left\|\Gamma x\right\|_\infty$$

The above objective function consists a least-square estimator cost function and an ℓ_∞-norm selection cost. The second part includes a matrix, Γ which sends x to input of ℓ_∞-norm. The case $\Gamma = I$, the problem turns into a Bayesian estimator whose prior information is embedded in ℓ_∞-norm cost function. Hence, x is selected as anti-sparse among the solutions of the least-square for the case $kernel(A)$ is nonempty. The case $\Gamma \neq I$ represents any linear operation on x. The important application for this case where Γ is a convolution matrix. Then, Γx represents the output of the corresponding linear time(shift) invariant system. The problem becomes least-square fitting while trying to minimize the maximum of the output of the linear system Γ, embedded in ℓ_∞-norm cost function. This case comes into existence in Compressed Training Adaptive Equalization.

The chapter 3 is organized as follows: Section 3.2 introduces the proximal mapping and proximal splitting methods. Section 3.3 investigates the algorithms for the solution of Setting 3.1. The numerical examples including Compressed Training Adaptive Equalization are given in Section 3.4.

3.2 Proximal Operators and Splitting

Consider the orthogonal projection operator defined onto set C such that $P_C(v) = \underset{x}{\operatorname{argmin}}\ \frac{1}{2}\left\|x-v\right\|_2^2$ for $x \in C$. Plugging the constraint into the objective function with a scalar factor λ,

$$P_C(v) = \underset{x}{\operatorname{argmin}}\ I_C(x) + \frac{1}{2\lambda}\left\|x-v\right\|_2^2$$

where I_C is the indicator function of set C. Generalizing the function I_C to f we get the proximal operator:

$$prox_{\lambda f}(v) = \underset{x}{\operatorname{argmin}}\ f(x) + \frac{1}{2\lambda}\left\|x-v\right\|_2^2$$

x^* is the solution to the above optimization problem, output of the proximal operator, if and only if

$$0 \in \lambda\partial f(x^*) + (x^* - v)$$

Furthermore, x^* is the unique minimizer of the problem due to strong convexity coming from the quadratic part, $\|x - v\|_2^2$. Accordingly proximal operator can be interpreted as a generalization of the projection [Combettes and Pesquet].

Rearranging the terms, the optimal condition turns to $x^* = v - \lambda F$ where $F \in \partial f(x^*)$. In other words, we applying a gradient descent step with stepsize λ where gradient is evaluated at output x^* on the contrary to the gradient descent method which we use the current gradient information. Due to use of future(next step) gradient information, proximal operator is considered as backward or implicit method in opposition to explicitness of gradient descent based methods[Goldstein et al., 2014].

The output of proximal mapping is also written as

$$x^* = (I + \lambda\partial f)^{-1} v \ \ or \ \ (v, x^*) \in (I + \lambda\partial f)^{-1}$$

. The operator $(I + \lambda\partial f)^{-1}$ is called the resolvent of operator $\lambda\partial f$. One of the most important properties of resolvent operators is that if T is a maximal monotone operator, its resolvent R_T converges to the fixed point which is a zero of T. In other words, to find a zero of operator T, we can apply resolvent operator to an initial point iteratively in order to find the fixed point of the resolvent. It is also proved that subdifferential operator of any closed, convex, proper function is a maximal monotone operator[Minty, 1962]. Thus, we can use resolvent of a subdifferential operator denoted as ∂f in order to find a zero of ∂f, which is a minimizer of the corresponding closed,convex and proper function f. This convergence property of proximal operator makes it useful for finding optimum points in convex optimization problems.

In the literature, iterative algorithms of the general form $x_{k+1} = R_{\lambda T}(x_k)$ are called proximal point algorithms. They are guaranteed to converge if T is a subdifferential operator of any CCP function. The proof of convergence and stability conditions on λ is given in [Ryu and Boyd, 2016, Rockafellar, 1976, Eckstein and Bertsekas,].

If T has a simple form, ie comprises a single operator ∂f, then we merely apply $(I + \lambda \partial f)^{-1}$ iteratively. If a simple closed form solution of proximal operator is known, then we do not have any problem. However, frequently, specifically for composite problems, we need to find a zero of a composite operator T, that is $T = \sum\limits_i \partial f_i$, whose resolvent is harder to evaluate. In order to achieve this, using resolvents that are easier and cheaper to evaluate, eg $(I + \lambda \partial f_i)^{-1}$ resolvents of its components, called splitting. Many splitting methods have been proposed such as forward-backward splitting, used in proximal gradient and dual proximal gradient, [Boyd and Vandenberghe, 2001, Beck and Teboulle, 2009]; Douglas-Rachford splitting, used in ADMM [Boyd et al., 2011], etc.

The main aim of splitting algorithms is that finding a zero of T by using simpler forms of their components. For example, in forward-backward splitting, we find the zero of T or the fixed point of R_{λ_T} for $T = A + B$ by finding the fixed point of $R_{\lambda_A}(I - \lambda B)$. In Douglas-Rachford splitting, we use Cayley operators, defined as $C_F = 2R_F - I$ on F, and resolvents.[Ryu and Boyd, 2016, Eckstein and Bertsekas,]

For a further discussion about proximal operators and operator splitting methods, the reader is referred to [Parikh et al., 2014, Combettes and Pesquet, 2011]. In the sequel, we investigate the algorithms for Setting 3.1.

3.3 Algorithms

Setting 3.1 is a convex optimization problem whose objective function is convex but non-differentiable. Otherwise stated, we assume that for any differentiable objective function, the gradient is Lipschitz continuos, i.e. $\|\nabla f(x) - \nabla f(y)\| < L(\|x - y\|)$ and $\sigma_{max}(A) < L$ for Setting 3.1. The non-differentiability of the cost function eliminates the gradient descent as a first order method and Newton method as a second order method. The one of the well known solutions of non-differentiable optimization is subgradient based algorithms [Boyd et al., 2003, Vandenberghe, 2019]. The method is proved to be convergent to optimal value if step size is chosen as zero-limit-divergent-sum(e.g. $t_k = \frac{1}{k}$ or $t_k = \frac{1}{\sqrt{k}}$). However, the convergence rate is proved to be $O(1/\sqrt{k})$ and cannot be improved [Vandenberghe, 2019]. The other well known solutions are Interior Point algorithms which is firstly suggested by John von Neumann and used by Karmarkar for linear programming. Later [Nesterov and Nemirovskii, 1994] extended Karmarkar's algorithm for nonlinear convex optimization problems. The interior point algorithms are used for convex optimization problems by doping the constraints onto the objective function using self-concordant logarithmic barrier functions. Most non-differentiable convex optimization problems involving problems involving ℓ_∞-norm can be cast as a second order cone programming and solved via interior point methods. However, large data sizes and dense matrices decreases the advantages of the relatively complex interior point algorithms[Beck and Teboulle, 2009]. Although generality of the interior point algorithms, the complexity and scalability problems of the interior point algorithms diminishes their value as a fast and low complexity solutions. These problems together with the application of Nesterov's acceleration approach highlight the proximal algorithms as a good solution for suitable convex optimization problems arising in many fields. In 2009, [Beck and Teboulle, 2009] show that functional convergence of their algorithm Fast-Iterative-Shrinkage-Thresholding(FISTA) is $0(1/k^2)$. FISTA is an accelerated version of proximal gradient algorithm, i.e. forward backward splitting, applied for ℓ_1-norm. After Nesterov proved the optimality of his acceleration approach for gradient descent methods, many application of his approach proved to be successful, fastens the convergence rate remarkably. [Nishihara et al., 2015] proves linear convergence of ADMM when one of the objective functions is strongly convex. Furthermore, an accelerated dual proximal gradient is introduced by [Beck and Teboulle, 2014] and functional convergence proved to be $O(1/k^2)$ and primal convergence to be $O(1/k)$. For the investigation of the solution for general Setting 3.1., we consider Alternating Direction Method of Multipliers known as ADMM, as first algorithm. It is proved to be a Douglas-Rachford Splitting applied to dual problem [Boyd et al., 2011] and can be considered as alternating minimization applied to method of multipliers. We present ADMM algorithm for our setting 3.1:

Algorithm 1 ADMM

```
1: **Initialize** $\rho > 0$
2: **repeat** for k = 1,2,...
3:     $x^{k+1} \leftarrow \underset{x}{\operatorname{argmin}} \frac{1}{2}\|Ax-b\|_2^2 + \frac{\rho}{2}\left\|\Gamma x - z^k + u^k\right\|_2^2$
4:     $z^{k+1} \leftarrow prox_{\frac{\lambda}{\rho}\ell_\infty}(\Gamma x^{k+1} + u^k)$
5:     $u^{k+1} \leftarrow u^k + \Gamma x^{k+1} - z^{k+1}$
6: **until** convergence
```

In ADMM, both primal optimization functions (3,4) are smooth. Hence, (3,4) results unique solution for $\forall\rho > 0$. Step 3 is equivalent to:

$$x^{k+1} = (A^TA + \rho\Gamma^T\Gamma)^{-1}(A^Tb + \rho\Gamma^T(u^k - z^k))$$

Step 4 is given as proximal mapping, which actually optimization of same objective function(Lagrangian with smoothing term added) over z. However, the convergence of the algorithm depends highly on the value of ρ. Hence, for each setting ADMM requires proper tuning of smoothing term ρ for high performance. ADMM results the convergence of dual variable and the objective function. As a convergence criteria, primal residual $\left\|r^k\right\|_2 = \|\Gamma x - z\|_2 < \epsilon^{pri}$ and dual residual $\left\|s^k\right\|_2 = \left\|\rho\Gamma^T(z^{k+1} - z^k)\right\|_2 < \epsilon^{dual}$ can be used [Boyd et al., 2011].

To improve the convergence rate of ADMM, a relaxation process can be applied as Tx^{k+1} replaced with $\widehat{Tx^{k+1}}$ given below for $\alpha \in (0, 2)$:

$$\widehat{Tx^{k+1}} = \alpha Tx^{k+1} - (1-\alpha)z^k \tag{3.1}$$

As a second algorithm, for the generalized Setting 3.1, we present fast dual proximal gradient [Beck and Teboulle, 2014] as an accelerated form of dual proximal gradient also known as Alternating minimization algorithm(AMA).

Algorithm 2 Fast Dual Proximal Gradient

```
1: Initialize $\rho > 0$
2: repeat for k = 1,2,...
3:     $x^{k+1} \leftarrow \underset{x}{\text{argmin}} \frac{1}{2} \|Ax - b\|_2^2 + w^{k^T}(\Gamma x - z)$
4:     $z^{k+1} \leftarrow prox_{\lambda L \ell_\infty}(\Gamma x^{k+1} + Lw^k)$
5:     $y^{k+1} \leftarrow y^k + \frac{1}{L}(\Gamma x^{k+1} - z^{k+1})$
6:     $t^{k+1} \leftarrow \frac{1+\sqrt{1+4(t^k)^2}}{2}$
7:     $w^{k+1} \leftarrow y^{k+1} + \frac{t^k - 1}{t^{k+1}}(y^{k+1} - y^k)$
8: until convergence
```

In fast dual proximal gradient algorithm, step 1 is the solution of the Lagrangian without extra smoothing term, instead of dual variable y, Nesterov momentum variable w is used as input. Step 2 is the same with ADMM, except momentum term w is used as input again. Steps (6,7) implements Nesterov's Acceleration approach. Nesterov variable w is updated in step 7 using the current and the previous value of y. By this approach, the acceleration mimics the second order method's information without using any second order differentiation.

Fast Dual Proximal Gradient algorithm as well as Dual Proximal Gradient Algorithm requires strong convexity of one of the objective functions. Hence, in our setting it requires $\sigma_{min}(A) > 0$. This requirement is due to the unique solution of step 1, where the extra smoothing term is not added.

3.4 Numerical Results

As the first experiment, we generate a random sample of Setting 3.1. For this purpose, we generate a random matrix $A \in \mathbb{R}^{80\times 40}$, $b \in \mathbb{R}^{80}$ and $T = \mathbb{R}^{2048\times 40}$. We compare the convergence behavior of the objective function for ADMM, ADMM with Relaxation and FDPG. ρ is set as 0.1 for both ADMM and its relaxation form and α is chosen as

1.8 for ADMM with relaxation. The normalized objective function convergence curve is given below as Figure 3.1:

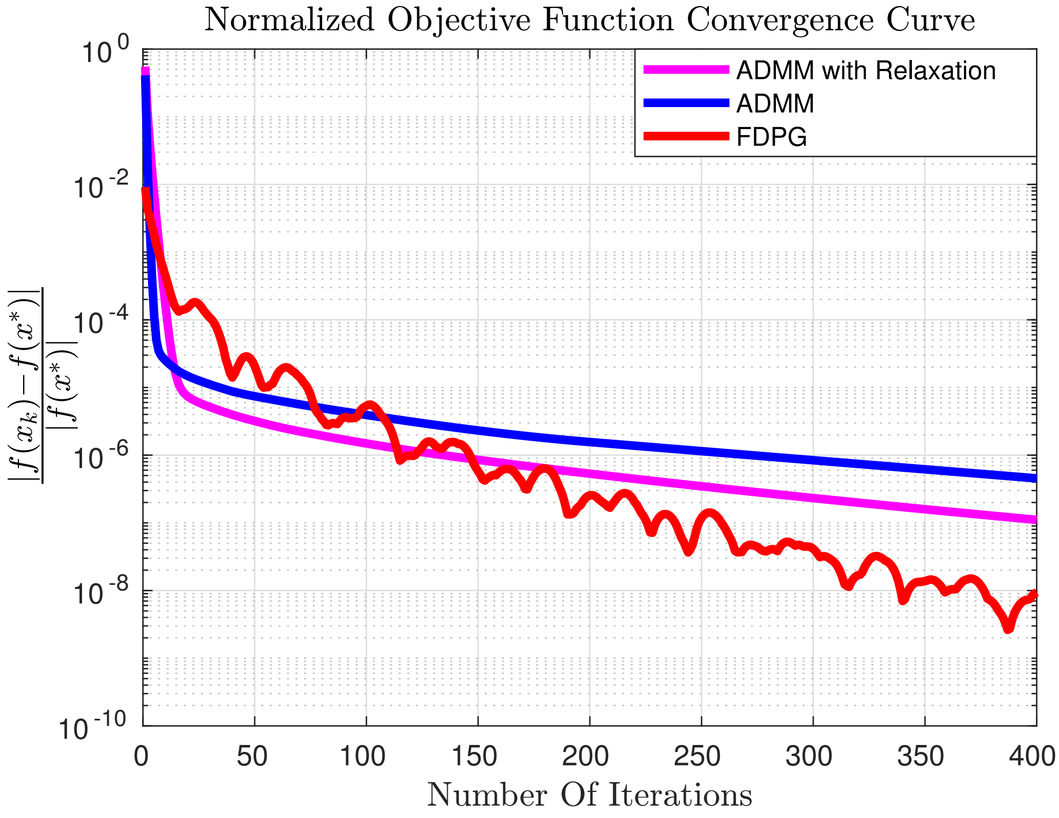

Figure 3.1: The normalized objective function convergence curve

According to the plot, both ADMM has faster initial convergence, however lower asymptotic convergence than FDPG. Furthermore, relaxation process improves convergence behavior but not significantly for our example.

As a second example, we consider Compressed Training Massive MIMO setting with $K = 40$ users and $M = 1000$ antennas. We set SNR as 5 dB. We set packet length $\Gamma = 300$, training length $L_T = 40$, data length $L_D = 260$. As modulation scheme, we use 4-QAM. We follow the notation introduced in [Yilmaz and Erdogan, 2019]. We consider the below optimization setting for each $k \in \{1, 2, ..., K\}$ as a form of Setting 3.1.:

MIMO CoTA Noiseless Optimization Setting

$$\underset{W_{k,:}}{\text{minimize}} \quad \frac{1}{2}\left\|W_{k,:}Y_T - S_{T_{k,:}}\right\|_2^2 + \lambda \left\|W_{k,:}Y\right\|_\infty$$

In setting 3.1, as introduced in [Yilmaz and Erdogan, 2019]:

- $W_{k,:} \in \mathbb{C}^{(1\times M)}$ represents the k'th row of the equalization matrix $W \in \mathbb{C}^{(K\times M)}$

- $Y \in \mathbb{C}^{(M\times\Gamma)}$ represents the received data matrix at the receiver of the base station with M antennas.

- $Y_T \in \mathbb{C}^{(M\times L_T)}$ represents the training data matrix at the base station. Y_T is a sub-matrix of Y.

- $S_{T_{k,:}} \in \mathbb{C}^{(1\times L_T)}$ represents the k'th row of the transmitted training symbol matrix $S_T \in \mathbb{C}^{(K\times L_T)}$, which is a sub-matrix of the transmitted sequence matrix $S \in \mathbb{C}^{(K\times\Gamma)}$

First, we look at mean SNR convergence(averaged for each k) for ADMM and FDPG. We use ADMM with $\rho = 0.003$. We present the average equalizer output SNR convergence plot in Figure 3.2.

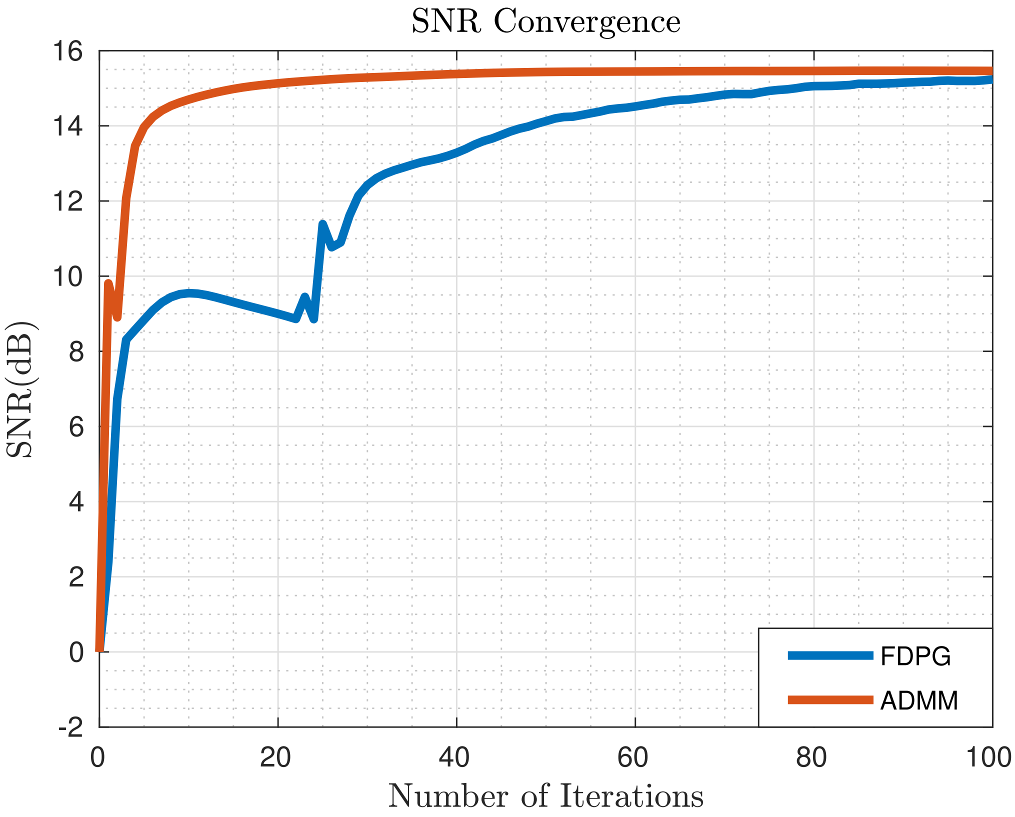

Figure 3.2: Average SNR Convergences for ADMM vs FDPG

We observe both ADMM and FDPG reaches same SNR level. While ADMM's convergence behavior is faster and more monotonic than FDPG's convergence behavior.

We present the average SNR convergences of ADMM for training length $L_T = 20$ and the packet lengths $\Gamma \in \{300, 500, 1000\}$ in Figure 3.3. Due to the loss of strong convexity of Y_T, fast dual proximal gradient algorithm is not applicable for the case, $L_T < 40$.

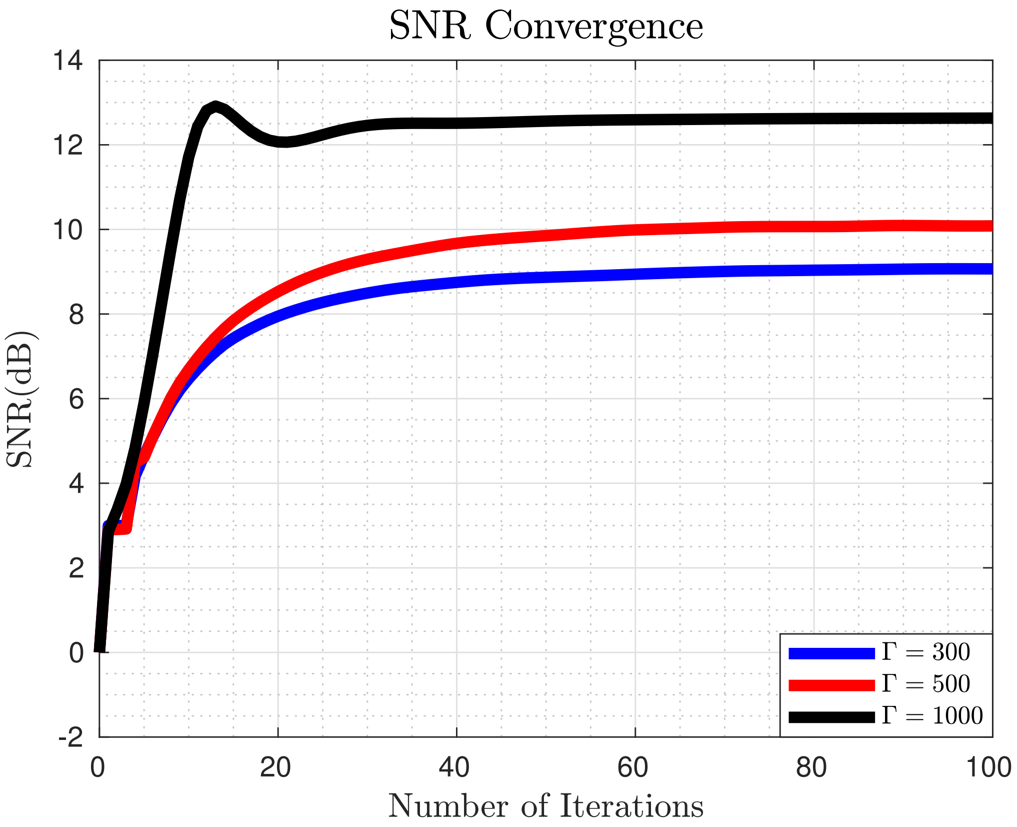

Figure 3.3: Average SNR Convergences of ADMM for $L_T = 20$

Chapter 4

FAST AND LOW COMPLEXITY COMPRESSED TRAINING BASED MASSIVE MIMO VIA ACCELERATED ALTERNATING PROJECTIONS

4.1 Introduction

As wireless connection technology emerges, the need for robust, flexible connections of many elements through wireless communication have become more important. In the era of internet of everything, the demand for high spectral efficiency constantly preserve its importance. To achieve target spectral efficiency as well as power efficiency, the use of large scale antenna arrays, Massive MIMO has emerged [Larsson et al., 2014, Lu et al., 2014]. As large scale, even thousand numbers of antennas are used, the major task of base station is processing received signals to get the correct transmitted signal compromising spectral efficiency. In order to resolve the signal interference, channel estimation and de-convolution is needed. Conventionally, training symbols are used to identify the channel and then correctly neutralize its effect. This conventional approach could be implemented in two steps with linear operations. However, it requires the number of training length equals to the number of users, K. As nonlinear optimization techniques improves and the use of convex cost functions such as ℓ_∞-norm introduced, blind equalization based approaches has emerged as direct approaches.[Vembu et al., 1994] Later, [Yilmaz and Erdogan, 2017] combines the convex cost functions and conventional training length approach, denoted as Compressed Training(CoTA), to reduce the required training length $\log_2 K$. The authors proved that with use of convex optimization settings and training length less than number of users higher Signal-to-Noise-Power-Ratio(SNR) and lower Bit-

Error-Rates are achievable. The authors also proposed subgradient based optimization algorithms for the solution of convex optimization settings.

In this chapter, we aim to develop a fast, low complexity algorithm for Compressed Training Approach introduced for Massive MIMO using the Noisy Optimization Setting [Yilmaz and Erdogan, 2017]. We propose an accelerated alternating projections algorithm. Appropriately, we derive the projection operators. We utilize ℓ_∞-norm in the constraints. Furthermore, we use an oblique(non-Euclidean) projection onto ellipsoid set for the sake of complexity.

The chapter is organized as follows: Section 4.2 introduces the problem and convex optimization setting. Section 4.3 introduces the used projection operators. Section 4.4 introduces the accelerated alternating projection algorithm. In Section 4, we present the experimental results.

4.2 Setting

For the notation, we use the notations in [Yilmaz and Erdogan, 2019], and we will consider the full noise consideration setting. As proposed by the authors, the convex cost functions to minimize for each $k \in \{1, .., K\}$ for K users:

$$\frac{\|O_{k,:}\|_\infty^2}{(\sqrt{\beta}-1)^2} + \sigma^2 \|W_{k,:}\|_2^2$$

with subject to two equality constraints:

- $W_{k,:}Y_T = S_{T_{k,:}}$ which reflects the requirement that output of the equalizer, when training region of the received samples is input, must be equal to training symbols. The equalizer must equalize the training region of the received signal.

- $W_{k,:}Y = O_{k,:}$ which reflects the equalizer output is convolution of the received input Y and the equalizer $W_{k,:}$.

Here, $O_{k,:} \in \mathbb{C}^{(1\times\Gamma)}$ represents the k'th row of the output matrix of the equalizer $O \in \mathbb{C}^{(K\times\Gamma)}$.

In order to reflect the true noise behaviour of the adaptive equalization process, two adjustment is applied by the authors on the optimization setting:

- Due to the imperfection of channel, the uplink training region Y_T contains noise. Due to the noise of Y_T, the relaxation of the equality $W_{k,:}Y_T = S_{T_{k,:}}$ by $\left\|W_{k,:}Y_T - S_{T_{k,:}}\right\|_2^2 \leq \sigma_n^2 L_T \left\|W_{k,:}\right\|_2^2$ represents the noise behaviour better.

- Due to the noise of the channel output Y, the relaxation of the equality constraint $W_{k,:}Y = O_{k,:}$ by $\left\|W_{k,:}Y - O_{k,:}\right\|_2^2 \leq \sigma_n^2 L_D \left\|W_{k,:}\right\|_2^2$ reflects the noise behaviour better.

In order to utilize the very low complexity of the projection operator for the ℓ_∞-norm ball, we place ℓ_∞-norm at the constraint by defining an auxiliary variable τ. We consider the below optimization setting denominated as Setting 4.1:

Setting 4.1

$$\begin{aligned}
&\underset{W_{k,:},O_{k,:}}{\text{minimize}} && \gamma^2 + \sigma_n^2 \left\|W_{k,:}\right\|_2^2 \\
&\text{subject to} && \\
& && \left\|W_{k,:}Y_T - S_{T_{k,:}}\right\|_2^2 \leq \sigma_n^2 L_T \left\|W_{k,:}\right\|_2^2 \\
& && \left\|W_{k,:}Y - O_{k,:}\right\|_2^2 \leq \sigma_n^2 L_D \left\|W_{k,:}\right\|_2^2 \\
& && \left\|O_{k,:}\right\|_\infty \leq \gamma
\end{aligned}$$

4.3 Projection-Based Algorithm for Compressed Training

Exploring the constraints of quadratic functions, Constraint 1 can be reformulated as:

$$(W_{k,:} - \alpha_1)X_1(W_{k,:} - \alpha_1)^H \leq f(S_{T_{k,:}}) \tag{4.1}$$

where

$$\alpha_1 = S_{T_{k,:}} Y_T^H X_1^{-1} \tag{4.2}$$

$$X_1 = Y_T Y_T^H - \sigma_n^2 L_T I \tag{4.3}$$

$$f(S_{T_{k,:}}) = S_{T_{k,:}} (Y_T^H X_1^{-H} Y_T - I) S_{T_{k,:}}^H \tag{4.4}$$

This representation reveals that a quadratic function centered on α at the left hand side is bounded by a known scalar quadratic function of $S_{T_{k,:}}$. Since the convexity of the problem is already satisfied, i.e. X to be positive semi-definite matrix, the constraint set 1 is a convex set ($epif(W_{k,:})$ is convex). Foremost, the form (4.1) defines an identified convex shape of $W_{k,:}$s, an ellipsoid.

In Constraint 2, on the left hand side the function is a quadratic function both $W_{k,:}$s and $O_{k,:}$. For the purpose to derive an explicit projection operator, we define an auxiliary vector $Z_{k,:} \in \mathbb{C}^{2n}$ as:

$$Z_{k,:} = [W_{k,:} O_{k,:}] \tag{4.5}$$

Now, constraint 2 is reformulated as:

$$Z_{k,:} X_2 Z_{k,:}^H \leq 0 \tag{4.6}$$

where

$$X_2 = \left[\begin{array}{c|c} YY^H - \sigma_n^2 L_D I & -Y \\ \hline -Y^H & I \end{array} \right] \tag{4.7}$$

Inequality (4.6) represents a set of $Z_{k,:}$s. We observe that $YY^H - \sigma_n^2 L_D I \geq 0$ due to the convexity of the problem and $det(X_2) = -\sigma_n^2 L_D I < 0$. Hence, the matrix is indefinite that the constraint set described by (4.6) is feasible.

To depict the above observations, we consider the below form of Setting 4.1 denoted as Setting 4.2:

Setting 4.2

$$\begin{aligned} &\underset{W_{k,:},O_{k,:}}{\text{minimize}} && \gamma^2 + \sigma_n^2 \left\| W_{k,:} \right\|_2^2 \\ &\text{subject to} && \\ & & (W_{k,:} - \alpha_1) X_1 (W_{k,:} - \alpha_1)^H &\leq f(S_{T_{k,:}}) \\ & & Z_{k,:} X_2 Z_{k,:}^H &\leq 0 \\ & & \left\| O_{k,:} \right\|_\infty &\leq \gamma \end{aligned}$$

where α_1, α_2, X_1, X_2, f given by (4.2), (4.3), (4.6), (4.7), (4.8). We denote the constraint set 1 of $W_{k,:}$s as C_1, the constraint set 2 of $W_{k,:}$s and $O_{k,:}$s as C_2 and the constraint set 3 of $O_{k,:}$s, the ℓ_∞-norm ball, as C_3.

4.3.1 Projection Operators

To develop the algorithm, we aim to derive fast, low complexity projection operators. For the purpose, three projection operators are used and denoted as P_{C_1}, P_{C_2} and P_{C_3} respectively.

- P_{C_1} is the projection operator for the ellipsoid set. To get a low complexity projection operator, we use an oblique(non-Euclidean) projection onto the ellipsoid set. The corresponding operator for the set C_1, centered at α, is given as (the proof is given as Appendix A.1):

$$\mathbb{P}_{C_1,X}(x) = \begin{cases} x & \left\| x - \alpha \right\|_X \leq 1 \\ \frac{x-\alpha}{\left\| x-\alpha \right\|_X} + \alpha & \left\| x - \alpha \right\|_X > 1 \end{cases} \tag{4.8}$$

For the projection onto ellipsoid set C_1, we use quadratic norm, which is defined as $\|\phi\|_X = (\phi X \phi^H)^{1/2}$, where X is the condition matrix of the ellipsoid. Accordingly, the projection operation is a non-orthogonal, oblique projection.

- P_{C_2} operator is equivalent to solving a quadratically constraint quadratic programming(QCQP) with 1 constraint. The projection operator for C_2 defined by the matrix X is given below:

$$\mathbb{P}_{C_2,X}(x) = x(I + \lambda^* X)^{-1} \tag{4.9}$$

where λ^* is given by the following procedure:

1. If λ^* satisfies $I + \lambda^* X > 0$, then λ^* is the solution of the below equation:
$$\sum_{1}^{n} (\frac{\lambda_i |y_i|^2}{1 + \lambda^* \lambda_i}) = 0 \tag{4.10}$$
2. Otherwise, λ^* is $\frac{-1}{\lambda_{min}}$ or $\frac{-1}{\lambda_{max}}$ whom satisfies $I + \lambda^* X \geq 0$

The proof is given as Appendix A.2. For further discussions about QCQP problems, the reader is referred to [Park and Boyd, 2017]. As the authors in [Appendix B], an iterative approach for finding λ^* could be used to solve the generalized QCQP with 1 constraint. Due to the construction of our problem, using preconditioned matrix results that Y has 3 distinct eigenvalues. Hence, the equation is algebraically solvable and procedure simplifies greatly.

- The projection operator for the set C_3, ℓ_∞ τ-unity norm ball is the element-wise clipping operation(proof given as Appendix A.3):

$$P_{C_3,\tau}(x) = x - S_\tau(x) = C_\tau(x) \tag{4.11}$$

4.3.2 Accelerated Alternating Projections Algorithm

So far, we derived the projection operators where we build the algorithm on. We will use an alternating projection approach where we apply three projection operators consecutively. Furthermore, we will add a Nesterov acceleration scheme in order to improve convergence. We refer the below algorithm as accelerated alternating projections.

Algorithm 3 Accelerated Alternating Projections Algorithm

```
1: for each k=1:K
2: Initialize $\rho^1 = Z^0_{k,:} = 0$, $\gamma^1 = 0$,$\tau = \tau*$
3: repeat
4:     $[\rho^t_1, \rho^t_2] \leftarrow \rho^t$
5:     $\rho^t_1 \leftarrow P_{C_1,X_1}(\rho^t_1)$
6:     $\rho^t_2 \leftarrow P_{C_3,\tau}(\rho^t_2)$
7:     $Z^t_{k,:} \leftarrow P_{C_2}([\rho^t_1, \rho^t_2])$
8:     $\gamma^{t+1} \leftarrow \frac{1+\sqrt{1+4(\gamma^t)^2}}{2}$
9:     $\rho^{t+1} \leftarrow Z^t_{k,:} + \frac{\gamma^t-1}{\gamma^{t+1}}(Z^t_{k,:} - Z^{t-1}_{k,:})$
10: until convergence
11: set $[W^*_{k,:}, O^*_{k,:}] \leftarrow Z_{k,:}$
```

In accelerated alternating projections, we consecutively apply three projection operators where we derived the previous section. Additionally, a Nesterov momentum variable is introduced as $\rho = [\rho_1, \rho_2]$ where $\rho_1 \in \mathbb{C}^{1\times M}$ and $\rho_2 \in \mathbb{C}^{1\times\Gamma}$. We remark that the adjoint Nesterov parameter ρ is used as input in the steps (4,5,6,7) to update $Z_{k,:} = [W_{k,:}, O_{k,:}]$. In steps (8,9), the Nesterov's acceleration approach is used and ρ is updated using the current and previous value of $W_{k,:}$ and $O_{k,:}$.

4.3.3 Numerical Result

As the experiment, we consider the uplink transmission scenario with packet length $\Gamma = 300$, training length $L_T = 20$ with $K = 40$ users and $M = 1000$ antennas. SNR per received branch is fixed as 5dB. We first investigate the output SNR performance per iteration. Additionally, to investigate the utility of the Nesterov scheme, we plot accelerated alternating projection and alternating projection(without Nesterov step) together with MIMO-CoTA2 [Yilmaz and Erdogan, 2019] in Figure.

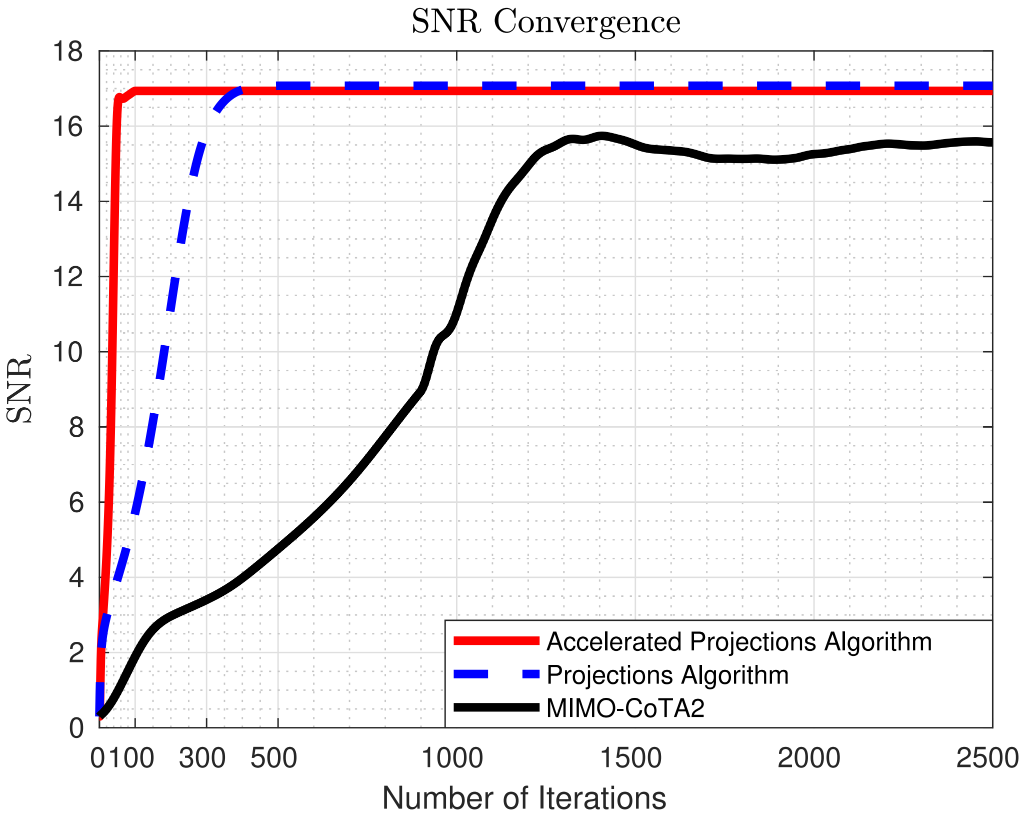

Figure 4.1: Average SNR Convergence for $L_T = 20$,$\Gamma = 300$

We further look at the convergence of the algorithm for different training lengths and packet lengths. We plot the mean convergence for (L_T,Γ)$\in \{(20, 300), (20, 500), (50, 300), (50, 500)\}$.

We also look at the Signal-to-Interference Plus Noise Ratio(SINR),defined by (4.12), that algorithm achieves for different training lengths and packet lengths as $L_T \in \{20, 50\}$ and $\Gamma = 200, 300, 500, 1000$, given as Figure 3.

$$SINR = \frac{\|diag(W_*H)\|_2^2}{\|idiag(diag(W_*H)) - W_*H\|_F^2 + \|W_*\|_F^2 \sigma_N^2} \tag{4.12}$$

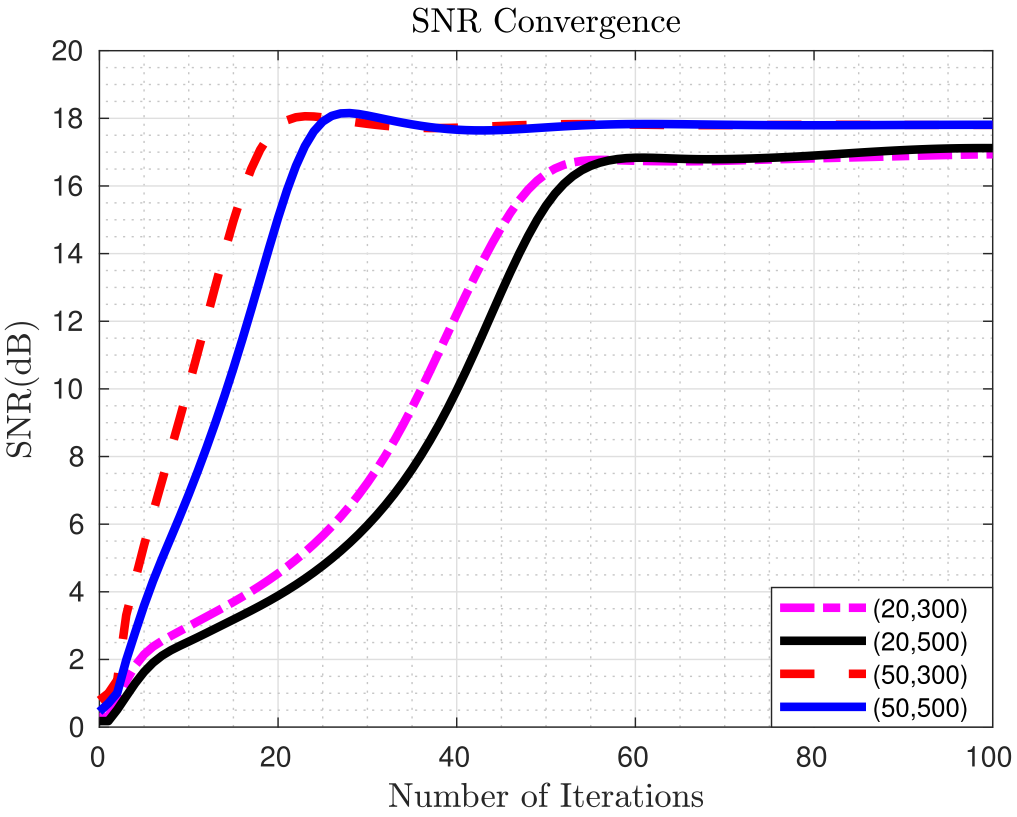

Figure 4.2: Average SNR Convergences

4.3.4 Complexity and Practical Considerations

The complexity of Accelerated Alternating Projections algorithm concentrates on the projection operator for set 2 which includes matrix inversion. The projection matrix $(I+\lambda^*X)^{-1}$ is equivalent to (13) when a preconditioner $W_p : \mathbb{R}^{(m\times\Gamma)} \rightarrow \mathbb{R}^{(n\times\Gamma)}$ and $\tilde{Y} = W_pY$, $\tilde{Y}\tilde{Y}^H = \alpha I$:

$$P_2 = \left[\begin{array}{c|c} \frac{\lambda^*+1}{\gamma}I & \frac{\lambda^*}{\gamma}\tilde{Y} \\ \hline \frac{\lambda^*}{\gamma}\tilde{Y}^H & \frac{1}{\lambda^*+1}I + \frac{\lambda^{*2}}{(\lambda^*+1)\gamma}\tilde{Y}^H\tilde{Y} \end{array}\right] \tag{4.13}$$

where

$$\gamma = \beta\lambda^* + \beta - \alpha\lambda^{*2}$$

$$\beta I = (I+\lambda^*X)_{11}; \beta = 1 + \lambda^*(\alpha - \sigma_n^2 L_D)$$

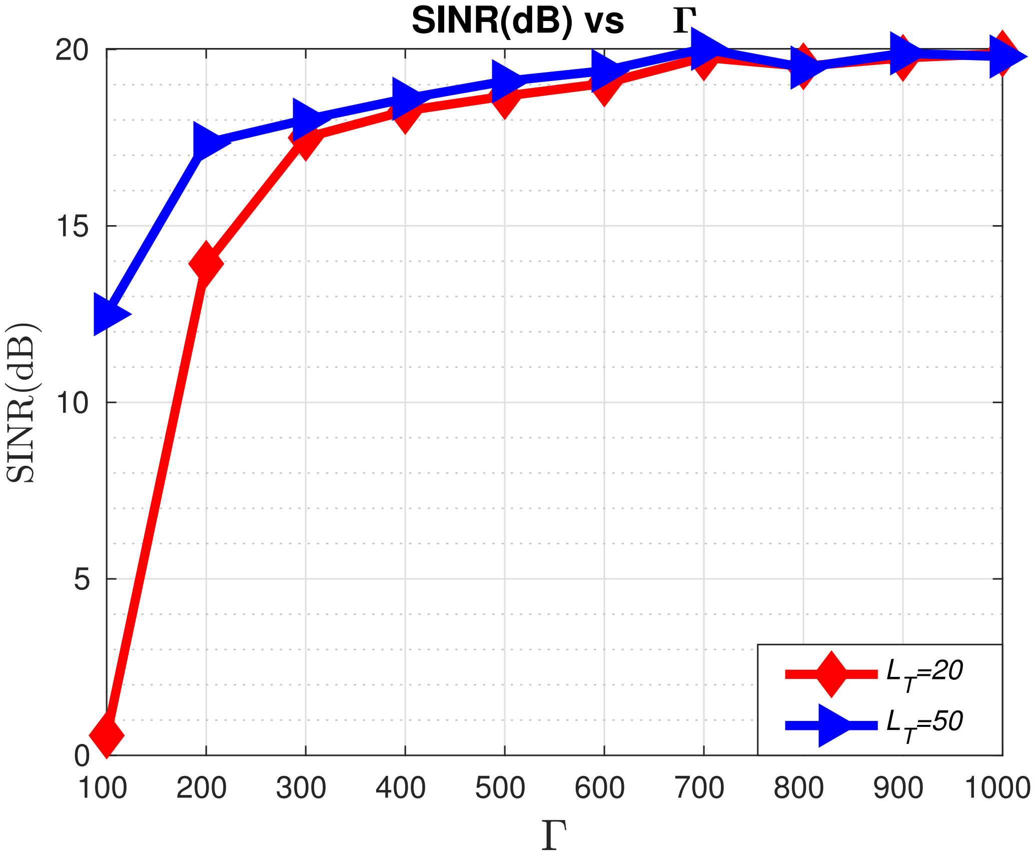

Figure 4.3: SINR Performance for $L_T = 20$ and $L_T = 50$

Using projection matrix P_2, the projection is calculated by block matrix-vector multiplication as:

$$P_2(\begin{bmatrix} \rho_1^T \\ \rho_2^T \end{bmatrix}) = \begin{bmatrix} \frac{\lambda^*+1}{\gamma}\rho_1^T + \frac{\lambda^*}{\gamma}\tilde{Y}\rho_2^T \\ \hline \frac{\lambda^*}{\gamma}\tilde{Y}^H\rho_1^T + \frac{1}{\lambda^*+1}\rho_2^T + \frac{\lambda^{*2}}{(\lambda^*+1)\gamma}\tilde{Y}^H\tilde{Y}\rho_2^T \end{bmatrix} \tag{4.14}$$

Hence, the cost is $O(8\Gamma n)$ flops per iteration.

On the other hand, the complexity of MIMO-CoTA-2 algorithm is dominated by the update of W which includes the inversion of the matrix, $(\lambda_1 Y_T Y_T^H + \lambda_2 Y Y^H + \tilde{\lambda} I)^{-1}$, and calculation of $O^{(t)}Y^H$. Using the same preconditioner, caching $Y_T Y_T^H$ and YY^H and applying the Cholesky decomposition, the complexity of update of W is $O(\frac{1}{3}n^3 + 2\Gamma n)$ flops per iteration that W is updated. The cost of updating U is dominated by the calculation of $W_{k,:}^{(t)}Y - S_{k,:}$. This cost is $O(2\Gamma n)$ per iteration that W is updated. Convergence waveforms are very similar for both algorithms if 50 iterations of Accelerated Projections and 1250 iterations of MIMO-CoTA-2 are applied. When W is updated at even every 5 iterations and the number of users, $K = 40$, total complexity of MIMO-CoTA2 is $O(4\Gamma e4 + 5.3e6)$ flops for each user index k. On the other the total cost of Accelerated Projections Algorithm is $O(1.6\Gamma e4)$ for each user index k.

Chapter 5

ON PROXIMAL OPERATOR FOR THE K-NORM

5.1 Introduction

The development of low complexity convex optimization algorithms for non-differentiable cost functions has been at the center of both signal processing and machine learning research. The momentum is mainly due to optimization settings promoting sparsity through the use of ℓ_1-norm.

The proximal approach offers fast and low complexity solutions for non-differentiable optimization settings [Parikh et al., 2014, Combettes and Pesquet, 2011]. It has been successfully applied especially to ℓ_1-norm based problem formulations due to the existence of the convenient soft-thresholding operator as the proximal operator for the ℓ_1-norm. Similarly, we can obtain an explicit form for the proximal operator of the ℓ_∞-norm, which turns out to be a clipping function whose level is dependent on its argument.

K-norm is defined as the sum of the K largest absolute elements. It actually defines a norm family based on the different choices of K. ℓ_1-norm is a special case with $K = n$, i.e., the dimension of the argument vector, and ℓ_∞-norm is another special case with $K = 1$.

The use of K-norm can be motivated from different directions. For example, in applications where the argument vector to ℓ_∞-norm is noisy, the actual peak value becomes ambiguous. In such a case, the set of large magnitude entries may represent potential actual peak locations and their average (sum with a proper scaling factor) can provide an estimate for the actual peak value. This perspective is potentially useful for applications using ℓ_∞-norm for blind (see for example [Vembu et al., 1994, Ding and Luo, 2000, Luo et al., 2002, Erdogan and Kizilkale, 2005] and compressed training based [Yilmaz and Erdogan, 2017] equalization. In [Lapin et al., 2015], k-norm cost function is used to extend the multi-class support vector machine approach to handle the class ambiguity. An alternative perspective on K-norm is regarding potential applications where it is more sensible to reduce the sum of the higher values than just the peak value, which can, for example, offer a balancing option between approaches minimizing the peak error versus the total error. As a practical tool, K-norm was used to reduce the complexity of Peak-to-Average Ratio minimization problem for multicarrier systems [Erdogan, 2006a].

K-norm is a special case of ordered (partial) ℓ_1-norm [Bogdan et al., 2015] where K non-zero norm-weights are set as equal. In reference [Bogdan et al., 2015], a proximity algorithm for the sorted ℓ_1-norm has already been introduced. In [Wu et al., 2014], the proximal operator for k-norm has been derived based on Moreau's decomposition which requires projection onto dual norm unity ball. In this article, we apply an alternative approach to derive the proximal operator for the K-norm and show that it has an explicit elementwise operation form which is the sum of soft-thresholding and clipping operators as shown in Fig. 5.1. We note that the soft thresholding, the proximal operator for the ℓ_1-norm, tends to equalize values around zero (to zero) while shifting the rest, and the clipping, the proximal operator for the ℓ_∞-norm tends to equalize the values around the peak value to a constant while leaving the remaining entries unchanged. Whereas the K-norm's proximal operator in Fig. 5.1 performs this equalization over an adjustable range based on the selection of the K parameter and the proximal operator's weighting parameter.

The article is organized as follows: in Section 5.2 is a reminder about the basic definition of the proximal operator. The proximal operator for the K-norm is derived in Section 5.3. Section 5.4 provide numerical examples illustrating the use of proximal operators.

Notation: Let $\mathbf{v}, \mathbf{x} \in \mathbb{R}^n$

$\mathbf{x} = [\mathbf{v}]^+$	Rectified Linear Unit Operator: $x_i = v_i$ if $v_i > 0$ and $x_i = 0$ otherwise.
$\mathbf{x} = [\mathbf{v}]^-$	Negative Rectified Linear Unit Operator: $x_i = v_i$ if $v_i < 0$ and $x_i = 0$ otherwise.
$\mathbf{1}$	Vector of all 1's (of appropriate size)
Δ	Unit Simplex in $\mathbb{R}^n$: $\Delta = \{\mathbf{x} \vert \sum_{i=1}^n x_i = 1, \mathbf{x} \in \mathbb{R}_+^n\}$

Table 5.1: Notation Table

5.2 Proximal Operators

The proximal operator for the convex function $f(\mathbf{x})$, with $\mathbf{x} \in \mathbb{R}^n$ is defined as [Parikh et al., 2014, Combettes and Pesquet, 2011]

$$prox_{\lambda f}(\mathbf{v}) = \arg\min_{\mathbf{x}} \ f(\mathbf{x}) + \frac{1}{2\lambda} \|\mathbf{x} - \mathbf{v}\|_2^2 . \tag{5.1}$$

$\mathbf{x}^*$ is the solution to the above optimization problem, output of the proximal operator, if and only if the condition

$$\mathbf{0} \in \lambda \partial f(\mathbf{x}^*) + (\mathbf{x}^* - \mathbf{v}), \tag{5.2}$$

holds. Furthermore, $\mathbf{x}^*$ is the unique minimizer of the problem due to strong convexity coming from $\|\mathbf{x} - \mathbf{v}\|_2^2$.

5.3 Proximal Operator of the K-Norm

We define K-norm as $\|\mathbf{x}\|_{[K]} = \sum\limits_{i=1}^{K} |x|_{[i]}$ where $|x|_{[i]}$ is the ith largest magnitude element of $\mathbf{x}$ or ith index value of descendingly sorted absolute form of $\mathbf{x}$. Then, the proximal operator of $f = \|.\|_{[K]}$ for $\mathbf{v} \in \mathbb{R}^n$ can be written as

$$prox_{\lambda f}(\mathbf{v}) = \underset{\mathbf{x}}{\operatorname{argmin}} \quad \lambda \|\mathbf{x}\|_{[K]} + (1/2) \|\mathbf{x} - \mathbf{v}\|_2^2 .$$

Before we introduce the expression for this proximal operator, we provide the following definitions:

- $C_\lambda(\mathbf{v}) = [\mathbf{v} - \lambda\mathbf{1}]^- + \lambda\mathbf{1} + [-\mathbf{v} - \lambda\mathbf{1}]^+$ is the elementwise clipping operation.

- $S_\lambda(v) = [\mathbf{v} - \lambda\mathbf{1})]^+ - [-\mathbf{v} - \lambda\mathbf{1}]^+$ is the elementwise soft-thresholding operation.

- $prox_{\lambda\ell_\infty}(\cdot)$ is the proximal operator for the ℓ_∞-norm.

The following theorem offers an explicit form for the proximal operator of the K-norm:

Theorem: The solution $\mathbf{x}^* = prox_{\lambda\|\cdot\|_{[K]}}(\mathbf{v})$ is given by the elementwise clipping combined subthresholding operator:

$$\mathbf{x}^* \quad = \quad C_\mu(\mathbf{v}) + S_{\mu+\lambda}(\mathbf{v}), \tag{5.3}$$

which is illustrated in Fig. 5.1, where

- $\mu = \left\| prox_{(K-\tau)\lambda\ell_\infty}(\mathbf{v}^{(s)}_{\tau+1:N}) \right\|_\infty$ is clipping parameter,

- $\mathbf{v}^{(s)} = |\mathbf{v}|$ such that $v^{(s)}_i \geq v^{(s)}_j$ for all $i, j \in \{1, 2, ..., n\}$, is the sorted absolute vector of $\mathbf{v}$, and is the permutation matrix corresponding to this sorting,

- $\tau \in \{1, \ldots, K-1\}$ is the number of components above $\mu + \lambda$, which satisfy the condition
  $v^{(s)}_\tau - \lambda - \left\| prox_{(K-\tau)\lambda\ell_\infty}(\mathbf{v}^{(s)}_{\tau+1:N}) \right\|_\infty > 0\})$.

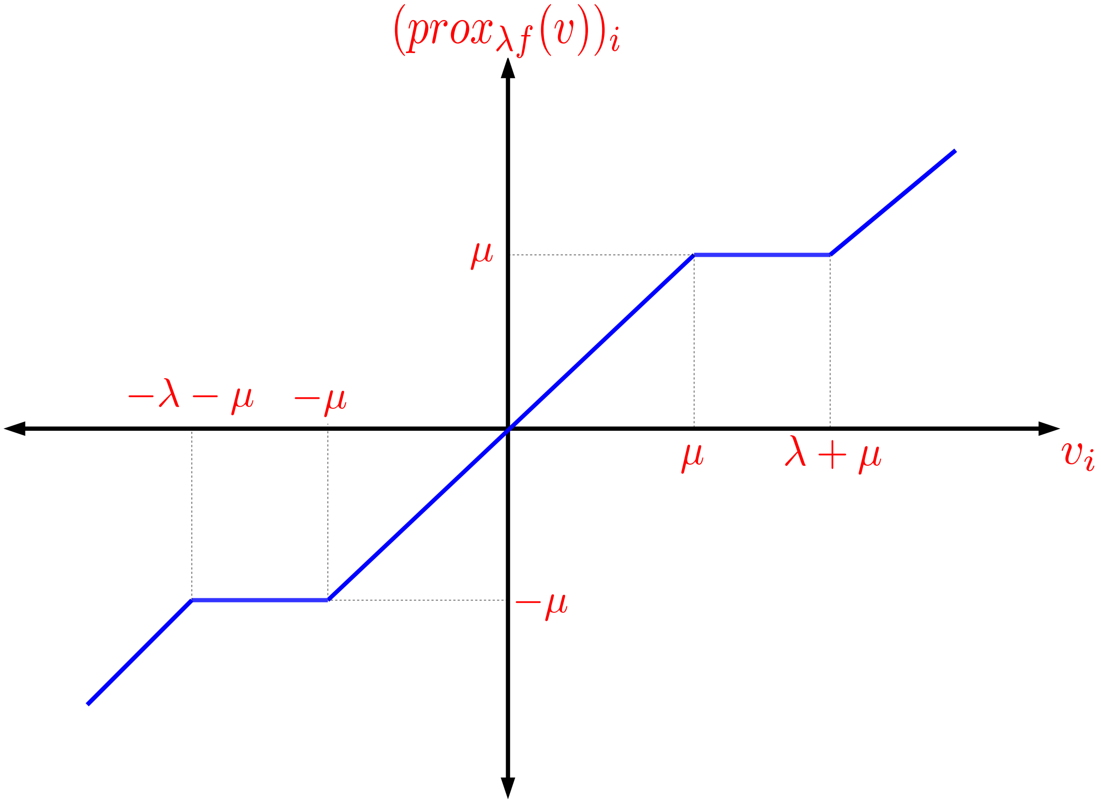

Figure 5.1: $(prox_{\lambda f}(\mathbf{v}))_i = C_\mu(\mathbf{v}_{_i}) + S_{\lambda+\mu}(\mathbf{v}_{_i})$

Proof: For the optimality condition in (2.2), we first need to obtain the expression for the subdifferential set of the K-norm. We start by providing the following definitions:

- We can write the K-norm function $f(\mathbf{x})$ as the maximum of some linear functionals in the form,

$$f(\mathbf{x}) = \max_{i=1,\dots,P} f_i(\mathbf{x}), \tag{5.4}$$

  where $f_i(\mathbf{x}) =_i^T \mathbf{x}$, and $_i$'s are distinct vectors with exactly K non-zero entries, and the non-zero entries take their values from the set $\{-1, 1\}$. There are $P = 2^K \binom{n}{k}$ such $_i$ vectors.

- At the optimal point $\mathbf{x}^*$, $f(\mathbf{x}^*) = f_i(\mathbf{x}^*)$ for $i \in \mathcal{W}^*$ where $\mathcal{W}^* \subset \{1, \dots, P\}$, and the subdifferential set is $\partial f(\mathbf{x}^*) = \mathbf{Co} \bigcup_{i \in \mathcal{W}^*} \{_i\}$.

- We refer to a set $\mathcal{T}_i^* \subset \{1, \ldots, n\}$ as an *active set* of the optimal point if and only if

  $$\sum_{l \in \mathcal{T}_i^*} |x_l^*| = f(\mathbf{x}^*),$$

  and $card(T_i)^* = K$. Let $\{\mathcal{T}_i^*, i = 1, \ldots, M\}$ represent the set of all active sets for the optimal point, where M is the number of distinct active sets.

- We define the set $\mathcal{I}^*$ as the union of all active sets, i.e., $\mathcal{I}^* = \bigcup_{i=1}^M \mathcal{T}_i^*$. We note that $C_{\mathcal{I}^*} \triangleq card(\mathcal{I}) \geq K$.

- We define the set $\mathcal{J}_2^*$ as the subset of $\mathcal{I}^*$

  $$\mathcal{J}_2^* = \{l ||x_l^*| = \min_{i \in \mathcal{I}} |x_i^*|, l \in \mathcal{I}^*\}, \tag{5.5}$$

  and $\mathcal{J}_1^*$ as its complement in $\mathcal{I}^*$, i.e., $\mathcal{J}_1^* = \mathcal{I} \cap \bar{\mathcal{J}}_2^*$. We also define $\tau \triangleq card(\mathcal{J}_1^*)$ and $L \triangleq card(\mathcal{J}_2^*)$ for $m = 1, 2$. Note that $\mathcal{J}_1^*$ contains the indexes that are contained in all active sets, i.e., $\mathcal{T}_i^*$'s, i.e., $\mathcal{J}_1^* \subset \mathcal{T}_i^*$ for all $i = 1, \ldots, M$.

- We also define an indicator function

  $$\gamma_i^*[j] = \begin{cases} 1 & \text{if } j \in \mathcal{T}_i^* \\ 0 & \text{otherwise,} \end{cases}$$

  i.e., it indicates whether a given index j is included in the active set $\mathcal{T}_i^*$. We observe that $\sum_{j \in \mathcal{I}^*} \gamma_i^*[j] = K$, and $\gamma_i^*[j] = 1$ for all $j \in \mathcal{J}_1^*$ and $i = 1, \ldots, M$.

Based on these definitions, we can write the subdifferential of the K-Norm at $\mathbf{x}^*$ as follows:

- if $\mathbf{x}^* = \mathbf{0}$, then $\mathcal{W}^* = \{1, \ldots, P\}$ and the subdifferential set is given by the polytope

  $$\partial f(\mathbf{0}) = \mathcal{P}_{n,K} = \{\sum_{i=1}^P \beta_{ii}| \in \Delta\}. \tag{5.6}$$

- if $\mathbf{x}^* \neq \mathbf{0}$ then we have two different conditions to consider

  i. When $1 \leq \|\mathbf{x}^*\|_0 \leq K-1$: In this case, $card(\mathcal{J}_2^*) = n - \tau$ and the components corresponding to J_2^* are equal to zero. The corresponding subdifferential set is given by

  $$\partial f(\mathbf{x}) = \{\sum_{j \in \mathcal{J}_1^*} sign(x_j^*)\mathbf{e}_j + \mathbf{s} | \mathbf{s} \in \mathcal{P}_{n,K-\tau}\} \tag{5.7}$$

  where

  $$\mathcal{P}_{n,K-\tau} = \{\sum_{i=1}^{P} \beta_{\bar{i}i} | \in \Delta\}, \tag{5.8}$$

  and $\bar{}_i$'s are distinct vectors with only $K-\tau$ non-zero entries (with values -1 or $+1$) and zero for indexes in $\mathcal{J}_1^*$.

  ii. When $\|\mathbf{x}^*\|_0 \geq K$, we have

  $$\begin{aligned}\partial f(\mathbf{x}^*) &= \{\sum_{i=1}^{M} \beta_i \sum_{j \in T_i^*} sign(x_j^*)\mathbf{e}_j | \in \Delta\} \\ &= \{\sum_{i=1}^{M} \beta_i \sum_{j \in \mathcal{I}^*} \gamma_i^*[j] sign(x_j^*) | \in \Delta\} \\ &= \{\sum_{j \in \mathcal{J}_1^*} \sum_{i=1}^{M} \beta_i \gamma_i^*[j] sign(x_j^*) + \\ &\qquad \sum_{j \in \mathcal{J}_2^*} \sum_{i=1}^{M} \beta_i \gamma_i^*[j] sign(x_j^*) | \in \Delta\} \\ &= \{\sum_{j \in \mathcal{J}_1^*} sign(x_j^*) + \sum_{j \in \mathcal{J}_2^*} \alpha_j() sign(x_j^*) | \in \Delta\}, \end{aligned} \tag{5.9}$$

  where, in the last line, we used $\sum_{i=1}^{M} \beta_i \gamma_i^*[j] = 1$ for $j \in \mathcal{J}_1^*$ and defined $\alpha_j() \triangleq \sum_{i=1}^{M} \beta_i \gamma_i^*[j]$.

Based on the subdifferential set expressions, we are ready to analyze the optimality condition in (5.2):

For $\mathbf{x}^* = 0$, the optimality condition in (5.2) implies that there exists $\mathbf{s}^* \in \mathcal{P}_{n,K}$

$$\mathbf{0} = \lambda \mathbf{s} - \mathbf{v}^* \Leftrightarrow \mathbf{v} = \lambda \mathbf{s}^*. \tag{5.10}$$

When $\mathbf{x}^* \neq 0$, based on (5.9) and (5.2), we analyze different cases outlined in the subdifferential derivations:

i. When $1 \left\|\mathbf{x}^*\right\|_0 \leq K-1$: for $j \in \mathcal{J}_1^*$ we obtain $x_j^* - v_j + \lambda sign(x_j^*) = 0$. from which we obtain $\mathbf{x}_j^* = v_j - \lambda sign(x_j^*)$. Here we observe that $sign(x_j^*)$ must be equal to $sign(v_j)$. Because, if x_j^* had opposite sign, that would cause extra cost on the objective function compared to the equal signed version due to the quadratic term $\left\|\mathbf{x}-\mathbf{v}\right\|_2^2$. In other words, there always exists an equal signed version of x_j^* which yields same cost on f and less cost on $\left\|\mathbf{x}-\mathbf{v}\right\|_2^2$ relative to the opposite signed version. As a result, the above expression simplifies to

$$|x_j^*| = |v_j| - \lambda, \text{for } j \in \mathcal{J}_1^*. \tag{5.11}$$

for $j \in \mathcal{J}_2$: $v_j = \lambda s_j^*$ for some $\mathbf{s} \in \mathcal{P}_{n,K-\tau}$.

ii. When $\left\|\mathbf{x}\right\|_0 \geq K$:

If the component index satisfies $j \notin \mathcal{I}^*$, then we have $x_j^* - v_j = 0$, or equivalently,

$$x_j^* = v_j \tag{5.12}$$

If the component index satisfies $j \in \mathcal{J}_1^*$, then we have

$$x_j^* - v_j + \lambda sign(x_j^*) = 0, \tag{5.13}$$

or equivalently, multiplying both sides with $sign(x_j^*)$, we obtain

$$|x_j^*| = v_j sign(x_j^*) - \lambda. \tag{5.14}$$

Based on the previous argument $sign(x_j^*) = sign(v_j)$, and therefore, we obtain the expression in (5.11).

As the final case, for the components with $j \in \mathcal{J}_2^*$, the optimality condition in (5.2) and the subdifferential set in (5.9) implies that there exists $^* \in \Delta$ such that

$$x_j^* = v_j - \lambda\alpha_j(^*)sign(x_j^*).$$

Due to the same reasoning as the previous case, we have $sign(x_j^*) = sign(v_j)$, and therefore,

$$|x_j^*| = |v_j| - \lambda\alpha_j(^*). \tag{5.15}$$

For a pair of indexes $j, j' \in \mathcal{J}_2^*$, due to the definition of $\mathcal{J}_2^*$, we have $|x_j^*| = |x_{j'}^*|$, which implies

$$|v_j| - \lambda\alpha_j(^*) = |v_{j'}| - \lambda\alpha_{j'}(^*), \tag{5.16}$$

or equivalently,

$$|v_j| = |v_{j'}| - \lambda\alpha_{j'}(^*) + \lambda\alpha_j(^*). \tag{5.17}$$

Summing both sides for all $j' \in \mathcal{J}_2^*$, we obtain

$$L|v_j| = \sum_{j' \in \mathcal{J}_2^*} |v_{j'}| - \lambda \sum_{j' \in \mathcal{J}_2'} \alpha_{j'}(^*) + L\lambda\alpha_j(^*).$$

Here, the second sum expression on the right simplifies to

$$\begin{aligned}\sum_{j' \in \mathcal{J}_2'} \alpha_{j'}(^*) &= \sum_{j' \in \mathcal{J}_2'} \sum_{i=1}^{M} \beta_i^* \gamma_i^*[j'] \\ &= \sum_{i=1}^{M} \beta_i^* \sum_{j' \in \mathcal{J}_2'} \gamma_i^*[j'] = \sum_{i=1}^{M} \beta_i^*(K - \tau) = (K - \tau).\end{aligned}$$

Therefore, we can write $|v_j| = \bar{v} - \lambda\frac{K-\tau}{L} + \lambda\alpha_j(^*)$, where $\bar{v} \triangleq \frac{1}{L}\sum_{j' \in \mathcal{J}_2^*} |v_{j'}|$. Manipulating this expression, we obtain an expressing for $\alpha_j(^*)$ as

$$\alpha_j(^*) = \frac{|v_j| - \bar{v}}{\lambda} + \frac{K - \tau}{L}. \tag{5.18}$$

Plugging (5.18) in (5.15), for $j \in \mathcal{J}_2^*$, we get

$$x_j^* = sign(v_j)(\bar{v} - \frac{\lambda(K - \tau)}{L}). \tag{5.19}$$

As a summary, the components corresponding to $\mathcal{J}_2^*$ are clipped to the same magnitude level

$$\mu = \bar{v} - \frac{\lambda(K - \tau)}{L}. \tag{5.20}$$

Since $0 \le \alpha_j \le 1$, based on (5.18), we can write

$$\mu \le |v_j| \le \mu + \lambda \;\text{ for } j \in \mathcal{J}_2^*. \tag{5.21}$$

Now, using the definitions of $\mathcal{J}_1^*$, $\mathcal{J}_2^*$ and equalities (5.12,5.11,5.19), $\forall j_1 \in \mathcal{J}_1^*, j_2 \in \mathcal{J}_2^*$ and $j_3 \notin I$ we have the following relations:

$$\left|x_{j_1}^*\right| > \left|x_{j_2}^*\right| = \mu = \bar{v} - \frac{(K-\tau)\lambda}{L} > \left|x_{j_3}^*\right| \tag{5.22}$$

$$|v_{j_1}| - \lambda > \bar{v} - \frac{(K-\tau)\lambda}{L} > |v_{j_3}| \tag{5.23}$$

We can summarize the characterization of the optimal point as

$$\mathbf{x}_j^* = \begin{cases} v_j & |v_j| < \mu, \\ \mu sign(v_j) & \mu \leq |v_j| \leq \mu + \lambda, \\ v_j - \lambda sign(v_j) & |v_j| > \mu + \lambda, \end{cases} \tag{5.24}$$

which corresponds to the mapping in Fig. 5.1, i.e., $C_\mu(\mathbf{v}) + S_{\mu+\lambda}(\mathbf{v})$ operator.

As a result, what remains is to identify τ and L which in turn determines μ. For this purpose, we first define the vector $\mathbf{v}^{(s)}$ as the vector obtained by taking the absolute values of the elements of $\mathbf{v}$ and then sorting them in descending order, we also define $= [\eta_1, \ldots, \eta_n]^T$ as the sorting index vector, i.e.. $v_m^{(s)} = v_{\eta_m}$ Note that the first τ elements of , is equal to the set $\mathcal{J}_1^*$ and its next L elements is equal to the set $\mathcal{J}_2^*$.

As an important observation, the optimality condition in (5.23) for indexes $\eta_{\tau+1}$, $\ldots \eta_n$, i.e., the components that do not correspond to $\mathcal{J}_1^*$ is exactly the same as the condition obtained for the ℓ_∞-norm proximal operator, where λ is replaced with $(K-\tau)\lambda$. Therefore, we can write

$$\begin{aligned} \mathbf{x}_{\eta_{\tau+1},\ldots,\eta_n} &= prox_{(K-\tau)\lambda\ell_\infty}(\mathbf{v}_{\eta_{\tau+1},\ldots,\eta_n}) \\ &= prox_{(K-\tau)\lambda\ell_\infty}(\mathbf{v}_{\tau+1,\ldots,n}^{(s)}). \end{aligned}$$

In the above expressions, τ, the cardinality of $\mathcal{J}_1^*$ needs to be determined based on the optimality conditions. In fact, if we define

$$\mu(i) = \|prox_{(K-\tau)\lambda\ell_\infty}(\mathbf{v}_{\tau+1,\ldots,n}^{(s)})\|_\infty. \tag{5.25}$$

where i needs to be selected in $\{1, \ldots, K-1\}$, for the correct choice $i = \tau$, we satisfy ,from (5.22), the condition that we can write $|x_{\eta_i}^*| > |x_{\eta_{i+1}}^*|$ which in turn implies,

$|v_{\eta_i}| - \lambda > \mu$, or equivalently $|v_i^{(s)}| - \lambda > \mu$. In fact, if we start with $i = K - 1$ and reduce it till we achieve this condition, which will in fact be an optimality certificate as also confirmed with Lemma 2.3 in [Bogdan et al., 2015] ■

5.4 Numerical Example

In order to illustrate the use of the proximal operator, we consider a Peak-to-Average power Ratio problem where we demonstrate over an OFDM signal with 256 carriers. In [Erdogan, 2006a]. the use of K-norm was suggested for this purpose.

In the numerical experiment, we chosen the set

$$\{5, 25, 54, 102, 125, 131, 147, 200, 204, 209, 247\}$$

as the indices of the reserved tones, and we assume that the data tones are loaded with 4-QPSK signals. The problem reduces to the optimization scheme where we use k-norm: min $\|\mathbf{\Gamma}\rho + \gamma\|_{[K]}$. We use Alternating Direction Method(ADMM) algorithm [Boyd et al., 2011], using the proximal operator derived in this article, choosing $K = 5$.

Fig. 5.2 demonstrates the convergence of objective function. As a result of 10^{th} iteration using 5-norm, we reduced PAR value about $2dB$ (from 8.47dB to 6.51dB) on average.

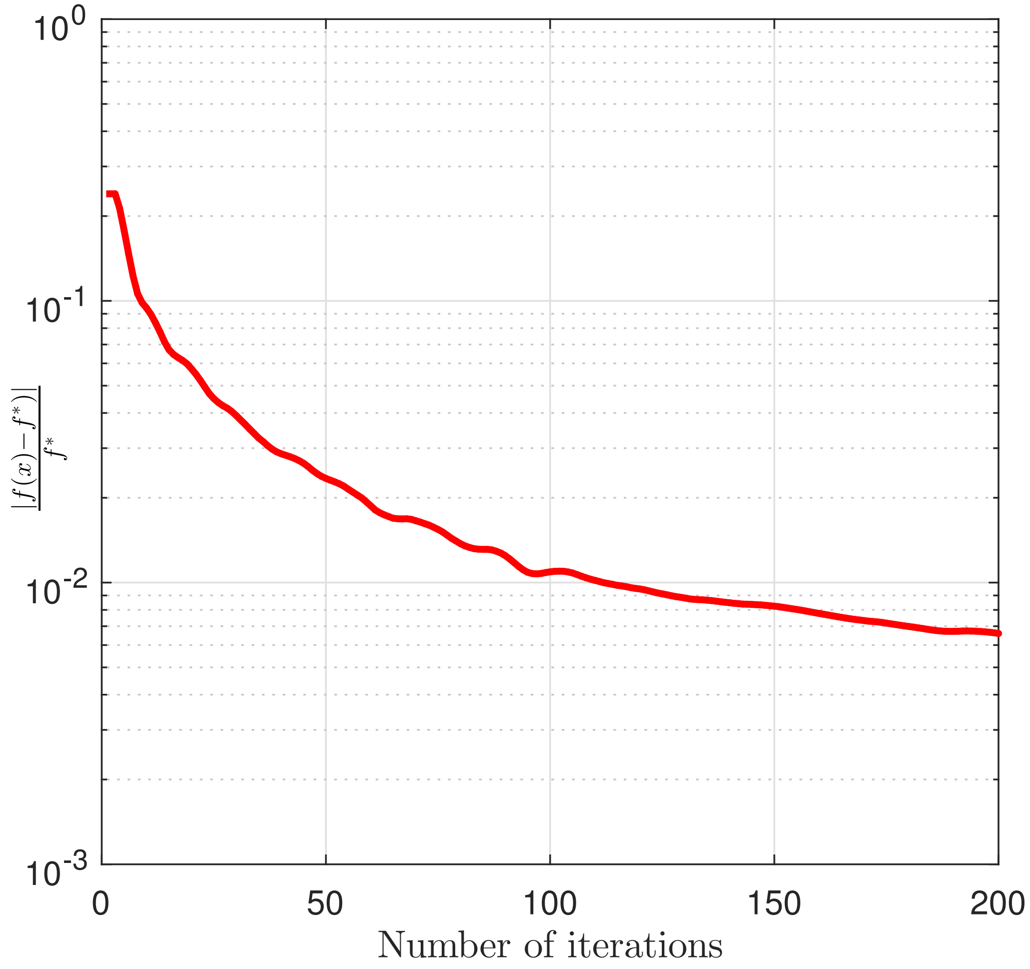

Figure 5.2: The normalized objective function convergence curve for ADMM algorithm

Chapter 6

CONCLUSION

ℓ_∞-norm emerges in many optimization settings for the real life problems such as Peak-to-Average-Power Ratio(PAPR) for multicarrier(OFDM/DMT) communication systems, frequency sampling based filter design. Also, ℓ_∞-norm is used as a cost function in blind equalization by minimizing ℓ_∞-norm of the output of the equalizer that provides sparsification of the combined impulse response. Moreover, the recently introduced semi blind approach , Compressed Training(CoTA), [Yilmaz and Erdogan, 2016] utilizes ℓ_∞-norm as a cost function. CoTA combines the least-square cost function with ℓ_∞-norm. By this composite approach, it utilizes the combined channel sparsification of ℓ_∞-norm and channel knowledge of the training symbols at the same time in order to significantly reduce the required training length for reliable channel equalization.

Due to nondifferentiability of ℓ_∞-norm, obtaining the proximal operator for ℓ_∞-norm is vital in order to achieve fast and low complexity solutions for the composite optimization settings involving ℓ_∞-norm as one of the cost function. Accordingly, the direct derivation of the proximity operator of ℓ_∞-norm is introduced and the clipping behavior of the operation is demonstrated explicity. A blind equalization example is presented as an application of the proximal operator of ℓ_∞-norm.

Compressed Training Based Massive MIMO approach results a composite cost function involving ℓ_∞-norm as one of the cost functions. Therefore, proximal splitting methods and proximal algorithms are required for efficient solutions of the problem. Chapter 2 investigates the proximal optimization algorithms and compares the Alternating Direction Method of Multipliers and Accelerated Fast Dual Proximal Gradient methods as the solutions of the CoTA Noiseless Setting.

Chapter 3 proposes an accelerated alternating projection algorithm as a fast and low complexity for the solution of Compressed Training Based Massive MIMO considering the setting due to full noise consideration. The required projection operators are derived and given as Appendices. On the contrary to proximal methods, the proposed algorithm uses projection operator of ℓ_∞-norm instead of the proximal operator.

The generalization of both ℓ_1-norm and ℓ_∞-norm results K-norm. For applications that peak elements replaces due to high Signal-to-Noise-Ratio, it may be more efficient to observe K peakest component instead the peakest component. As K-norm considers the biggest K elements, it can be considered as a relaxed form of ℓ_∞-norm. We present a direct derivation for its proximity operator while observing its relation to the proximal operators of ℓ_∞-norm and ℓ_1-norm. Additionally, an example execution of the proximal operator of K-norm is given in PAPR problem using ADMM.

K-norm has possible applications where we want to minimize K number of peakest components. As its proximity operation includes both soft thresholdind and clipping behaviour, it may be used as a trade-off measure between sparsity and anti-sparsity. It can be used to adjust sparsity or anti-sparsity. The applications and statistical properties of K-norm requires further investigations.

BIBLIOGRAPHY

[Alpaydin, 2014] Alpaydin, E. (2014). *Introduction to Machine Learning*. The MIT Press.

[Beck and Teboulle, 2009] Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

[Beck and Teboulle, 2014] Beck, A. and Teboulle, M. (2014). A fast dual proximal gradient algorithm for convex minimization and applications. *Oper. Res. Lett.*, 42:1–6.

[Bogdan et al., 2015] Bogdan, M., van den Berg, E., Sabatti, C., Su, W., and Candès, E. J. (2015). Slope adaptive variable selection via convex optimization. *The annals of applied statistics*, 9(3):1103.

[Boyd et al., 1994] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics.

[Boyd et al., 2011] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122.

[Boyd and Vandenberghe, 2001] Boyd, S. and Vandenberghe, L. (2001). *Convex Optimization*. Stanford University.

[Boyd et al., 2003] Boyd, S., Xiao, L., and Mutapcic, A. (2003). Subgradient methods. *Notes for EE392o, Stanford University, Autumn Quarter, 2003*, 2004.

[Candès, 2008] Candès, E. (2008). The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9):589–592.

[Combettes and Pesquet, 2011] Combettes, P. L. and Pesquet, J.-C. (2011). Proximal Splitting Methods in Signal Processing. In Bauschke, H.H.; Burachik, R. C. P. E. V. L. D. W. H. E., editor, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer.

[Ding and Luo, 2000] Ding, Z. and Luo, Z. (2000). A fast linear programming algorithm for blind equalization. *IEEE TCOM.*, 48:1432–1436.

[Duchi et al., 2008] Duchi, J., Shalev-Shwartz, S., Singer, Y., and Chandra, T. (2008). Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 272–279, New York, NY, USA. ACM.

[Eckstein and Bertsekas,] Eckstein, J. and Bertsekas, D. Monotone operators and the proximal point algorithm. *Mathematical Programming*, 55(1:3):293318.

[Erdogan, 2006a] Erdogan, A. T. (2006a). A low complexity multicarrier par reduction approach based on subgradient optimization. *Signal Processing*, 86(12):3890–3903.

[Erdogan, 2006b] Erdogan, A. T. (2006b). A simple geometric blind source separation method for bounded magnitude sources. *IEEE Trans. on Signal Processing*, 54:438–449.

[Erdogan, 2013] Erdogan, A. T. (2013). A class of bounded component analysis algorithms for the separation of both independent and dependent sources. *IEEE Transactions on Signal Processing*, 61(22):5730–5743.

[Erdogan and Kizilkale, 2005] Erdogan, A. T. and Kizilkale, C. (2005). Fast and low complexity blind equalization via subgradient projections. *IEEE Trans. on Signal Processing*, 53:2513–2524.

[Goldstein et al., 2013] Goldstein, T., Li, M., Yuan, X., Esser, E., and Baraniuk, R. (2013). Adaptive primal-dual hybrid gradient methods for saddle-point problems. *arXiv preprint arXiv:1305.0546*.

[Goldstein et al., 2014] Goldstein, T., Studer, C., and Baraniuk, R. G. (2014). A field guide to forward-backward splitting with a FASTA implementation. *CoRR*, abs/1411.3406.

[Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

[Kiseliov, 1994] Kiseliov, Y. N. (1994). Algorithms of projection of a point onto an ellipsoid. *Lithuanian Mathematical Journal*, 34(2):141–159.

[Lapin et al., 2015] Lapin, M., Hein, M., and Schiele, B. (2015). Top-k multiclass svm. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 325–333, Cambridge, MA, USA. MIT Press.

[Larsson et al., 2014] Larsson, E. G., Edfors, O., Tufvesson, F., and Marzetta, T. L. (2014). Massive MIMO for next generation wireless systems. *IEEE Communications Magazine*, 52(2):186–195.

[Lu et al., 2014] Lu, L., Li, G. Y., Swindlehurst, A. L., Ashikhmin, A., and Zhang, R. (2014). An overview of massive MIMO: Benefits and challenges. *IEEE Journal of Selected Topics in Signal Processing*, 8(5):742–758.

[Luo et al., 2002] Luo, Z.-Q., Meng, M., Wong, K. M., and Zhang, J.-K. (2002). A fractionally spaced blind equalizer based on linear programming. *IEEE Trans. on Signal Processing*, 50:1650–1660.

[Minty, 1962] Minty, G. J. (1962). Monotone (nonlinear) operators in hilbert space. *Duke Math. J.*, 29(3):341–346.

[Nesterov and Nemirovskii, 1994] Nesterov, Y. and Nemirovskii, A. (1994). *Interior-Point Polynomial Algorithms in Convex Programming.* Society for Industrial and Applied Mathematics.

[Nishihara et al., 2015] Nishihara, R., Lessard, L., Recht, B., Packard, A., and Jordan, M. (2015). A general analysis of the convergence of admm. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 343–352, Lille, France. PMLR.

[Parikh et al., 2014] Parikh, N., Boyd, S., et al. (2014). Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239.

[Park and Boyd, 2017] Park, J. and Boyd, S. (2017). General Heuristics for Nonconvex Quadratically Constrained Quadratic Programming. *arXiv e-prints*, page arXiv:1703.07870.

[Proakis, 2000] Proakis, J. G. (2000). *Digital Communications.* McGraw Hill.

[Rockafellar, 1976] Rockafellar, R. (1976). Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14:877–898.

[Ryu and Boyd, 2016] Ryu, E. K. and Boyd, S. (2016). A primer on monotone operator methods: Survey. *Appl. Comput. Math.*, 15(1):3–43.

[Shalev-Shwartz and Singer, 2006] Shalev-Shwartz, S. and Singer, Y. (2006). Efficient learning of label ranking by soft projections onto polyhedra. *Journal of Machine Learning Research*, 7(Jul):1567–1599.

[Vandenberghe, 2019] Vandenberghe, L. (2019). Optimization methods for large scale systems. *Lecture Notes for ECE236C, UCLA, Spring, 2019*, 2019.

[Vembu et al., 1994] Vembu, S., Verdu, S., Kennedy, R., and Sethares, W. (1994). Convex cost functions in blind equalization. *IEEE Trans. on Signal Processing*, 42:1952–1960.

[Wu et al., 2014] Wu, B., Ding, C., Sun, D., and Toh, K.-C. (2014). On the moreau-yosida regularization of the vector k-norm related functions. 24:766794.

[Yilmaz and Erdogan, 2016] Yilmaz, B. B. and Erdogan, A. T. (2016). Compressed training adaptive equalization. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4920–4924. IEEE.

[Yilmaz and Erdogan, 2017] Yilmaz, B. B. and Erdogan, A. T. (2017). Compressed training adaptive equalization: Algorithms and analysis. *IEEE Transactions on Communications*, 65(9):3907–3921.

[Yilmaz and Erdogan, 2019] Yilmaz, B. B. and Erdogan, A. T. (2019). Compressed training based massive mimo. *IEEE Transactions on Signal Processing*, 67(5):1191–1206.

Chapter 7

APPENDICES

7.1 Appendix A: Projection Onto Ellipsoid Set

The projection operator for an ellipsoid set can be found by the following convex optimization problem. We denote projection operator of the closed convex and bounded set $W \in \mathbb{H}$ as P_W where $\mathbb{H}$ represents finite n-dimensional Hilbert space.

The following optimization setting is to find the projection operator for a point $x' \in \mathbb{H}$ onto an ellipsoid $W \in \mathbb{H}$.

Projection Onto Ellipsoid Setting-1

$$\begin{aligned} &\underset{w'}{\text{minimize}} && \|x' - w'\|_2^2 \\ &\text{subject to} && \\ & && (w' - \alpha)X(w' - \alpha)^H \leq 1 \end{aligned}$$

We reformulate the problem via variable exchange by setting $w = w' - \alpha$, origin is translated to α. Hence, solving the current formulation is as hard as solving the below problem (x,w translated as: $x = x' - \alpha$ and $w = w' - \alpha$):

Projection Onto Ellipsoid Setting-2

$$\begin{aligned}&\underset{w}{\text{minimize}} && \|x-w\|_2^2\\ &\text{subject to} && \\ & && wXw^H \leq 1\end{aligned}$$

To find the projection operator, we can use Lagrangian function. Firstly, the projection operator $P = I$ for $\forall x \in W$. We analyze the case: $x \notin W$ We write Lagrangian function as:

$$L(w,\lambda) = \|x-w\|_2^2 + \lambda wXw^H - 1 \tag{7.1}$$

Minimizing over w, $\nabla_w L(w,\lambda)|_\lambda^* = 0$,

$$\nabla_w L(w,\lambda^*) = 2(w-x) + 2\lambda^* wX = 0 \tag{7.2}$$

From KKT conditions, complementary slackness for $\lambda^* > 0$ reveals:

$$\nabla_\lambda L(w,\lambda) = wXw^H - 1 = 0 \tag{7.3}$$

The solution is given by:

$$w^* = x(I + \lambda^* X)^{-1} \tag{7.4}$$

where scalar optimal dual λ^* is found by solving the Equation [Kiseliov, 1994]:

$$w^* X w^{*H} - 1 = 0; \tag{7.5}$$

Incorporating Equation (7.4) into (7.5)

$$x(I + \lambda^* X)^{-1} X (I + \lambda^* X)^{-H} x^H = 1 \tag{7.6}$$

$$x(X^{-1/2} + \lambda^* X^{1/2})^{-1})^2 x^H = 1 \tag{7.7}$$

We eject the equality $\|x\|_X = \gamma$ as $\frac{1}{\gamma}xXx^H = 1$ into (7.7) to get:

$$xq_{\lambda^*}(X)x^H = 0 \tag{7.8}$$

where

$$q_{\lambda^*}(X) = (X^{-1/2} + \lambda^* X^{1/2})^{-2} - \frac{1}{\gamma}X \tag{7.9}$$

Since, $X = X^H \in \mathbb{H}^{n \times n}$, X is decomposable as $X = Q\Omega Q^H$ where Q is a complex unitary matrix and Ω is a real diagonal matrix. The eigenvalue decomposition is inherited by $q_{\lambda^*}(X)$ as:

$$q_{\lambda^*}(X) = Q((\Omega^{-1/2} + \lambda^* \Omega^{1/2})^{-2} - \frac{1}{\gamma}\Omega)Q^H \tag{7.10}$$

$$q_{\lambda^*}(X) = Q(\Omega(I + \lambda^* \Omega)^{-2} - \frac{1}{\gamma}\Omega)Q^H \tag{7.11}$$

Setting $p_{\lambda^*}(\Omega) = (I + \lambda^*\Omega)^{-2} - \frac{1}{\gamma}I$ and $y = xQ\Omega^{1/2}$ the requirement (7.8) becomes:

$$xQp_{\lambda^*}(\Omega)Q^H x^H = yp_{\lambda^*}(\Omega)y^H \tag{7.12}$$

$$= tr(xQp_{\lambda^*}(\Omega)Q^H x^H) \tag{7.13}$$

$$= tr(p_{\lambda^*}(\Omega)Q^H x^H xQ) = 0 \tag{7.14}$$

$y = xQ\Omega^{1/2}$ is calculated, λ^* is calculated by solving:

$$yp_{\lambda^*}(\Omega)y^H = 0 \tag{7.15}$$

The optimum value λ^* can be found via an iterative Newtonian process
[Kiseliov, 1994].

Observing the objective function is monotonically decreasing on λ^*, $g'(\lambda^*) < 0$, and has positive curvature, $g''(\lambda^*) > 0$, the solution can be found by iteration given as 7.16, depicted in the figure below:

$$\lambda^{k+1} = \lambda^k - \frac{g(\lambda^k)}{g'(\lambda^k)} \tag{7.16}$$

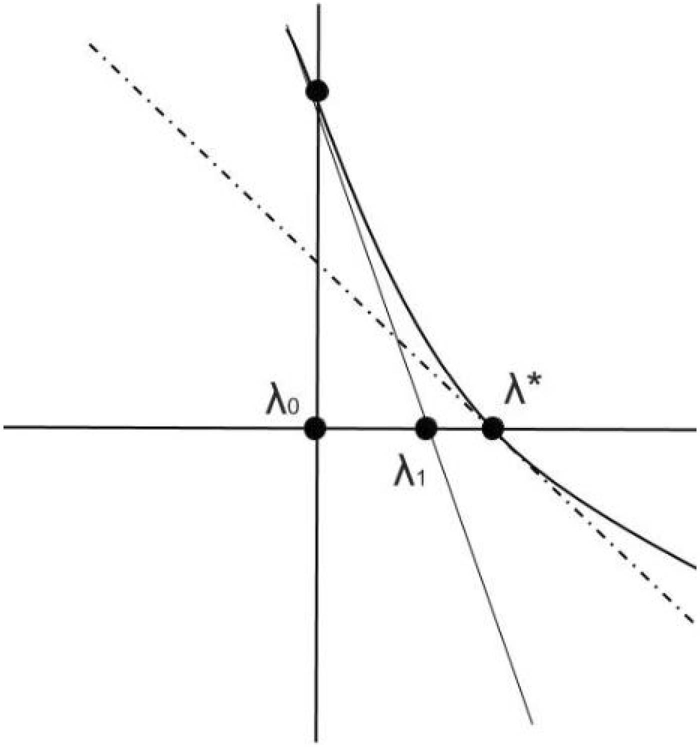

Figure 7.1: Newton Iteration Step

Lemma: The projection is defined as the mapping $x' \in \mathbb{H}$ onto a closed convex ellipsoid set W centered on α. The projection is casted as a convex optimization problem as Setting 1. Let $P_{W-\alpha}$ be the translated projection operator defined by translated vectors in Setting 2 as $P_{W-\alpha}(x) = xP_{W-\alpha} = x(I + \lambda^* X)^{-1}$. Then the projection operator of W, P_W, is defined by:

$$w'^* = P_W(x') = (x' - \alpha)P_{W-\alpha} + \alpha \tag{7.17}$$

$$P_W(x') = x'P_{W-\alpha} + \alpha(I - P_{W-\alpha}) \tag{7.18}$$

$$P_{W-\alpha} = (I + \mathbb{I}(\lambda^*)X)^{-1} \tag{7.19}$$

$$\mathbb{I}(\lambda^*) = \lambda^* \ , \quad x' \notin W \tag{7.20}$$

$$0 \ , \quad x' \in W \tag{7.21}$$

where $\lambda^* > 0$ is found by solving (7.15).

Since $P_{W-\alpha}$ is a linear function of x (7.4). Then, P_W is an affine function of x'.

7.1.1 Special Case:

In this section, we derive a closed form simple solution for the defined projection operator using quadratic norms. The new projection operator illustrates a non-Euclidean projection. We consider the below setting:

Projection Onto Ellipsoid Setting-3

$$\underset{w}{\text{minimize}} \quad \|x - w\|_X^2$$

subject to

$$wXw^H \leq 1$$

In this case, projection is applied with matrix conditioned norm. Then, equation (7.2) changes to (7.22), and (7.4) to (7.23):

$$\nabla_w L(w, \lambda) = 2(w - x)X + 2\lambda w X = 0 \tag{7.22}$$

$$w^* = \frac{1}{1 + \lambda^*} x \tag{7.23}$$

Thus, requirement (7.5) becomes (7.24):

$$(\frac{1}{1+\lambda^*})^2 xXx^H - 1 = 0 \tag{7.24}$$

which simplifies to (7.25):

$$\lambda^* = (\sqrt{\frac{1}{xXx^H}})^{-1} - 1 \tag{7.25}$$

Projection operator transforms to a clipping-like behaviour, i.e. normalization function for upper-threshold values, when matrix X induced quadratic norm ($<\phi,\phi>_X=<\phi,\phi X>=\phi X\phi^H = \|\phi\|_X^2$) is used.

$$P_{W-\alpha,X} = \begin{cases} x & \|x\|_X \leq 1 \\ \frac{x}{\|x\|_X} & \|x\|_X > 1 \end{cases} \tag{7.26}$$

Hence, (7.17) changes to:

$$P_{W,X}(x') = \begin{cases} x' & \|x'-\alpha\|_X \leq 1 \\ \frac{x'-\alpha}{\|x'-\alpha\|_X} + \alpha & \|x'-\alpha\|_X > 1 \end{cases} \tag{7.27}$$

Remark: For the projection operator, significant part of the computational complexity is only computation of $\|x'-\alpha\|_X$ which is $O(2n^2)$. Total computation of operator for a dense X matrix is approximately $2n^2$ multiplication. If X is a diagonal matrix, the multiplication diminishes to $2n$. However, this case requires X to be positive definite($X > 0$) and positive semi-definite($X \geq 0$) to be respectively norm or semi-norm.

7.2 Appendix B: Projection onto ℓ_∞-norm ball

We consider finding the projection operator for ℓ_∞-norm ball, denoted as P_{ℓ_∞}. The problem is casted as a convex optimization framework by the following setting:

Projection onto ℓ_∞-norm ball

$$\begin{aligned} &\underset{o}{\text{minimize}} \quad \frac{1}{2}\left\|x-o\right\|_2^2 \\ &\text{subject to} \\ &\qquad\qquad \left\|o\right\|_\infty \leq \tau \end{aligned}$$

Proof: We denote the τ unit ℓ_∞-norm ball as B_τ and closure of the set B_τ as ∂B_τ, defined as $\partial B_\tau = B_\tau - int(B_\tau)$.

We embark upon derivation by digging into Lagrangian cost function where $\lambda \geq 0$

$$L(o,\lambda) = \frac{1}{2}\left\|x-o\right\|_2^2 + \lambda(\left\|o\right\|_\infty - \tau) \tag{7.28}$$

We regard two situations about x:

$$(i) : \left\|x\right\|_\infty \leq \tau$$

$$(ii) : \left\|x\right\|_\infty > \tau$$

(i) describes the situation where input x is already inside the ℓ_∞-norm ball, $x \in B_\tau$. Bearing primal feasibility($\left\|o^*\right\| \leq \tau$) and dual feasibility($\lambda \geq 0$ and complementary slackness($\lambda(\left\|o\right\|_\infty - \tau) = 0$) properties of KKT(Karush-Kuhn-Tucker) conditions, $\lambda^* = 0$ reveals, thereby for (i):

$$o^* = x \tag{7.29}$$

(ii) depicts the condition where input x is outside the ℓ_∞-norm ball, $x \in ext(B_\tau)$. Complementary slackness property brings that $\exists\ o_1 \in \partial B_\tau,\ \ \forall\ o_2 \in int(B_\tau)$: $\left\|x-o_1\right\|^2 < \left\|x-o_2\right\|^2$ since $\exists\ o_1 \in l(x,o_2)$ line. Therefore, $o^* \in \partial B_\tau$ which equals $\left\|o^*\right\|_\infty = \tau$. Noting that $\lambda^* > 0$, this result is also confirmed by

$$\nabla_\lambda L(o^*,\lambda)|_{\lambda^*} = \left\|o^*\right\|_\infty - \tau = 0 \tag{7.30}$$

$$\|o^*\|_\infty = \tau \tag{7.31}$$

Now, we construct o^* by minimizing over o such that $\nabla_o L(o, \lambda^*)|_{o^*} = 0$. For the purpose, we define an effective index set I as $I = \{i : \forall i \in (1, 2, ..., n) : |x_i| > \tau\}$. The equation becomes for ineffective components, $\forall i \in I^c$, as:

$$\nabla_o L(o, \lambda)|_{o^*i} = o_i^* - x_i = 0 \tag{7.32}$$

which leads to

$$o_i^* = x_i, \quad i \in I^c \tag{7.33}$$

Secondly, for the effective components,the requirement becomes:

$$\nabla_o L(o, \lambda)_i = (o_i^* - x_i) + \lambda^* \beta_i sign(o_i^*) = 0 \tag{7.34}$$

where

$$\sum_{i \in I} \beta_i = 1 \tag{7.35}$$

$$o_i^* = x_i - \lambda^* \beta_i sign(o_i^*) \tag{7.36}$$

Multiplying each by $sign(o_i^*)$ and recalling that $sign(x_i) = sign(o_i^*)$ for each $i \in I$:

$$|o_i^*| = |x_i| - \lambda^* \beta_i \tag{7.37}$$

Setting $m = card(I)$ and summing over all $i \in I$, we get dual variable λ^*, and β_is as:

$$\lambda^* = \sum_{i \in I} (|x_i|) - m\tau \tag{7.38}$$

$$\beta_i = \frac{|x_i| - |o_i^*|}{\lambda^*} = \frac{|x_i| - \tau}{\sum_{i \in I} (|x_i| - \tau)} \tag{7.39}$$

Putting β_is and λ into (63) confirms that for $i \in I$:

$$o_i^* = sign(o_i^*)\tau \tag{7.40}$$

Putting all together reveals projection operator for ℓ_∞-norm ball as an elementwise clipping operator $C_\tau(.)$(2.3)(Fig. 2.1)

$$P_{B_\tau}(.) = C_\tau(.) \tag{7.41}$$

■

7.2.1 Relation to ℓ_1-norm

Duality between ℓ_∞-norm and ℓ_1-norm leads to the duality between projection operator for norm balls and proximal operators of the norms, that can be observed beatifully by Moreau's Decomposition Theorem. The theorem reveals that projection operator onto one norm ball can be deconstructed using proximal operator of its dual norm. (Bausche and Combette Monotone Operator,Combettes,Boyd,Vandenberghe)

Explicitly, Moreau's decomposition theorem states that an element of a Hilbert space, $\forall x \in \mathbb{H}$, can be directly decomposable into two resolvents of monotone operators such as:(Chambolle, A.)

$$x = (I + \tau T)^{-1}(x) + \tau(I + \frac{1}{\tau}T^*)^{-1}(\frac{x}{\tau}) \tag{7.42}$$

The decomposition of $\forall x \in \mathbb{H}$ is regarded as a sum of orthogonal projections of x onto subspaces(Vandenberghe), ie any $\mathbb{H}$ can be decomposable to two orthogonal subspaces denoted as L, $L^\perp$ defined using monotone operators:

$$\mathbb{H} = L \oplus L^\perp \tag{7.43}$$

Setting $T = \partial f$, subgradient of f, where f is any convex function, (monotonicity of subgradient operator requires convexity), and f^* is Legendre-Fenchel dual of f identity becomes:

$$Cx = \mathrm{prox}_{\tau f}(x) + \tau \mathrm{prox}_{\tau^{-1} f^*}(\tau^{-1} x) \tag{7.44}$$

$$= \mathrm{prox}_{\tau f}(x) + \mathrm{prox}_{(\tau f)^*}(x) \tag{7.45}$$

Normalizing τ reveals kernel of Moreau Decomposition:

$$x = prox_f(x) + prox_{f^*}(x) \tag{7.46}$$

Remark 6.1:Let C be closed, convex and bounded set in a Hilbert space. Projection operator for the orthogonal projection onto C is denoted as P_C and The Indicator function for the set is denoted as $\mathbb{I}_C$.Then, P_C is identical with proximal operator of $\mathbb{I}_C$.

$$P_C(x) = \mathrm{prox}_{\mathbb{I}_C}(x) \tag{7.47}$$

From this point of view, proximal operator can be regarded as a generalization of orthogonal projection operator. [Combettes and Pesquet, 2011] Briefly, an orthogonal projection can be defined as a proximal operator using indicator functions.

For the purpose, we use the notation $P_{\infty,\tau}$ for the representation of orthogonal projection onto τ-unit ℓ_∞-norm ball and $I_{\infty,\tau}$ to represent indicator function of a τ-unit ℓ_∞-norm ball.

Remark 6.2: $I_{\infty,\tau}(x) = \tau_{\infty,1}(\tau^{-1}x)$ and similarly $I_{1,\tau}(x) = \tau I_{1,1}(\tau^{-1}x)$.

Actually $\forall \alpha > 0$, $I_{\infty,\tau}(x) = \alpha_{\infty,1}(\tau^{-1}x)$ and similarly $I_{1,\tau}(x) = \alpha I_{1,1}(\tau^{-1}x)$.

Lemma 6.3: $P_{\infty,1}(x) = \tau^{-1} P_{\infty,\tau}(\tau x)$

Proof: Following the remark, orthogonal projection operators can be written as proximal operators:

$$P_{\infty,1}(x) = \mathrm{prox}_{I_{\infty,1}}(x) = \tau^{-1}\mathrm{prox}_{I_{\infty,\tau}}(\tau x) = \tag{7.48}$$

$$\tau^{-1} P_{\infty,\tau}(\tau x) \tag{7.49}$$

Writing explicitly,

$$\operatorname{prox}_{I_{\infty,1}}(x) = \underset{y}{\operatorname{argmin}} \frac{1}{2} \|y - x\|_2^2 + I_{\infty,1}(y) = \tag{7.50}$$

$$= \underset{z}{\operatorname{argmin}} \frac{1}{2} \|z - \tau x\|_2^2 + I_{\infty,\tau}(z) = \operatorname{prox}_{I_{\infty,\tau}}(\tau x) \tag{7.51}$$

Writing optimality conditions:

$$0 \in (y^* - x) + \partial I_{\infty,1}(y^*) \tag{7.52}$$

$$0 \in (z^* - \tau x) + \partial I_{\infty,\tau}(z^*) \tag{7.53}$$

Putting y^* and z^* at the left hand side:

$$y^* \in x - \partial I_{\infty,1}(y^*) \tag{7.54}$$

$$z^* \in \tau x - \partial I_{\infty,\tau}(z^*) \tag{7.55}$$

Using Remark 6.2, (81) becomes:

$$z^* \in \tau x - \partial_z(\tau I_{\infty,1}(\tau^{-1} z^*)) \tag{7.56}$$

$$z^* \in \tau(x - \partial I_{\infty,1}(\tau^{-1} z^*)) \tag{7.57}$$

Using (80) and (83) concludes:

$$\tau^{-1} z^* = y^* \tau^{-1} \tag{7.58}$$

$$\tau^{-1} P_{\infty,\tau}(\tau x) = P_{\infty,1}(x) \tag{7.59}$$

■

We construct orthogonal projection operator onto τ-unit ℓ_∞-norm ball using Moreau's Decomposition:

$$x = \operatorname{prox}_{\ell_1}(x) + \operatorname{prox}_{I_{\infty,1}}(x) \tag{7.60}$$

$$x = \operatorname{prox}_{\ell_1}(x) + P_{\infty,1}(x) \tag{7.61}$$

We replace $\text{prox}_{I_{\infty,1}}(x)$ with $\tau^{-1}P_{\infty,\tau}(\tau x)$ using Lemma 6.3, and $\text{prox}_{\ell_1}(x)$ with $\tau^{-1}\text{prox}_{\tau\ell_1}(\tau x)$:

$$x = \tau^{-1}\text{prox}_{\tau\ell_1}(\tau x)) + \tau^{-1}P_{\infty,\tau}(\tau x) \tag{7.62}$$

and $\text{prox}_{\ell_1}(x)$ with $\tau^{-1}\text{prox}_{\tau\ell_1}(\tau x)$ Hence,

$$P_{\infty,\tau}(\tau x) = \tau x - \text{prox}_{\tau\ell_1}(\tau x) \tag{7.63}$$

which leads to:

$$P_{\infty,\tau}(x) = x - \text{prox}_{\tau\ell_1}(x) \tag{7.64}$$

Denoting $\text{prox}_{\tau\ell_1}(x)$ as the shrinkage operator with parameter τ as $S_\tau(x)$, we conclude as:

$$P_{\infty,\tau}(x) = x - S_\tau(x) = C_\tau(x) \tag{7.65}$$

■

7.3 Appendix C: Projection Onto the Constraint Set 2

We consider the below setting for projection onto the constraint set 2.

$$\begin{aligned} &\underset{z}{\text{minimize}} && \|x - z\|^2 \\ &\text{subject to} && \\ & && zXz^H \leq 0 \end{aligned}$$

We use $K > 0$ matrix induced norm as the cost function. This optimization setting as well as the other Euclidean or non-Euclidean projection onto ellipsoids defined by quadratic or matrix induced norms is a quadratically constrained quadratic programming (QCQP). We do not assume the convexity of the problem since for CoTA Setting-1, X is indefinite and the constraint set is not convex yet feasible. Since the problem is feasible, the strong duality is satisfied by Lagrangian relaxation solution [Park and Boyd, 2017]. The Lagrangian defined for the setting is given as:

$$L(z, \lambda) = \|x - z\|^2 + \lambda(zXz^H) \tag{7.66}$$

From $\nabla_z L(z, \lambda)|_{\lambda}^{*} = 0$,

$$z^* = x(I + \lambda^* X)^{-1} \tag{7.67}$$

by the assumption of $I + \lambda^* X$ is invertible, since X is indefinite, the invertibility is not guaranteed for $\lambda^* \geq 0$. Together with feasibility($I + \lambda^* X \geq 0$ is brought by feasibility), invertibility assumption results $I + \lambda^* X > 0$. If $x \in C_2$, i.e. x in the feasible region and satisfies xXx^H, the optimal primal value $z^* = x \leftrightarrow \lambda^* = 0$. If $z \notin C_2$,then $\lambda > 0$ and complementary slackness of the KKT conditions brings:

$$\nabla_\lambda L(z, \lambda)|_{z^*} = z^* X z^{*H} = 0 \tag{7.68}$$

Putting (89) into (90) becomes:

$$x(I + \lambda^* X)^{-1} X (I + \lambda^* X)^{-1} x^H = 0$$

Following the same procedure as we did for the projection onto ellipsoids setting 2, i.e. applying eigenvalue decomposition(is not invertible for this case) yields:

$$y p_\lambda^*(\Omega) y^H = 0 \tag{7.69}$$

where

$$p_\lambda^*(\Omega) = (I + \lambda^* \Omega)^{-1} \Omega (I + \lambda^* \Omega)^{-1} = (I + \lambda^* \Omega)^{-2} \Omega \tag{7.70}$$

and

$$y = xQ \tag{7.71}$$

For $(I + \lambda^* X) > 0$ and invertibility assumption guarantess $1 + \lambda^* \lambda_i > 0$. Hence,

$$\sum_1^n \left(\frac{\lambda_i |y_i|^2}{1 + \lambda^* \lambda_i}\right) = 0 \tag{7.72}$$

Above equation can be solved by iteration that λ^* satisfies $I + \lambda^* X > 0$. Otherwise, λ^* could be $\frac{-1}{\lambda_{min}}$ or $\frac{-1}{\lambda_{max}}$ which satisfies $I + \lambda^* X \geq 0$ (see Park Boyd Appendix B).

Remark: For X_2 given as (7), $\begin{pmatrix} Q^H_{YY^H} & 0_{n\times m} \end{pmatrix}$ is a set of eigenvectors with corresponding set of eigenvalues $\Omega - \sigma_n^2 L_D I$. Moreover, for the case, it can be shown that $m - n$ eigenvalues are exactly 1. Knowing $det(X_2)$ and eigenvalues of YY^H enable us to calculate the other set of eigenvalues that overall process can be simplified.