Convex Relaxations of the Maximin Dispersion Problem

by

Sheena Ayla Haines

B.Sc. Hons., The University of British Columbia, 2009

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The College of Graduate Studies

(Mathematics)

THE UNIVERSITY OF BRITISH COLUMBIA

(Okanagan)

July 2011

Abstract

Recently, convex relaxations have achieved notable success in solving NP-hard optimization problems. This thesis studies semidefinite and second-order cone programming convex relaxations of the maximin dispersion problem. Providing nontrivial approximation bounds, we believe that our SDP and SOCP relaxation methods are useful in large-scale optimization.

The thesis is organized as follows. We begin by recalling some basic concepts from convex analysis, nonsmooth analysis, and optimization. We then introduce the weighted maximin dispersion optimization problem; locating point(s) in a given region $\mathcal{X} \subseteq \mathbb{R}^n$ that is/are furthest from a given set of m points. Also given are several reformulations of the original problem, including a convex relaxation problem and semidefinite and second order cone programming relaxations. Using well known results on Lipschitz functions and subdifferentials of Lipschitz functions we then derive a theoretical characterization of the optimal solutions for a given convex region $\mathcal{X}$ and equal weights. Next, we provide a proof that the weighted maximin dispersion problem is NP-hard even in the case where $\mathcal{X}$ is is a box and the weights are equal. We then propose an algorithm for finding an approximate solution using the SDP relaxation, and derive an approximation bound that depends on the subset $\mathcal{X}$. In the case that $\mathcal{X}$ is a box or a product of low-dimensional spheres, we show that the convex relaxation reduces to a second-order cone program, and provide further results on the bound provided. Lastly, we provide numerical examples using the SDP and SOCP relaxations, and suggest future work.

Table of Contents

Appendices

List of Tables

List of Figures

Acknowledgements

I would like to thank my supervisors Jason Loeppky and Shawn Wang for their infinite patience and encouragement. Their passion and enthusiasm for mathematics and statistics has both inspired me and given me the motivation to pursue my Master's degree. My thanks also go to my thesis committee members Heinz Bauschke, Yong Gao and Warren Hare for their insightful comments and suggestions which have improved the presentation of my thesis. I would also like to thank Richard Taylor for being the first professor to encourage me to pursue mathematics, and Rebecca Tyson for giving me my first opportunity to pursue mathematical research. Most importantly, I want to acknowledge the work of Paul Tseng. The work in this thesis is based on an unpublished paper that Paul was working on at the time of his disappearance. Recognizing the significance of his work, my supervisors suggested that we take Paul's ideas and carry them to their conclusion, the result of which is this thesis. Without Paul's ideas this thesis would never have existed, and I hope that I have done justice to him in this work.

Dedication

This work is dedicated to my mom, who has always believed in me, and the late (Captain) Bob MacAuley, who was the first person to ever spark my interest in mathematics, physics and statistics.

Chapter 1

Introduction

The fundamental building blocks of all studies in optimization can be found in calculus, linear algebra and analysis. This chapter collects background materials and known facts which will be used in later chapters. Sections 1.1, 1.3 and 1.4 give a brief overview of the necessary results from these fields which are used to build the basic theoretical results of optimization. Sections 1.5.1 and 1.5 then introduce concepts which are required specifically for solving non-convex optimization problems, like the one presented in this thesis.

1.1 Preliminaries

In this work the geometry of problems is mainly concerned with Euclidean space and finding solutions that are subsets of $\mathbb{R}^n$. In this section we discuss the geometry and fundamental concepts of the real vector space $\mathbb{R}^n$. We also recall a few very basic definitions from calculus which are required throughout the rest of this thesis.

We denote the set of n-dimensional real vectors by $\mathbb{R}^n$, and for any $x \in \mathbb{R}^n$ we denote the i^{th} component of x by x_i. Thus, each $x \in \mathbb{R}^n$ is a column vector

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}.$$

For any $x, y \in \mathbb{R}^n$ we define the inner product $\langle \cdot, \cdot \rangle$ by

$$\langle x, y \rangle = x^T y = \sum_{i=1}^{n} x_i y_i. \tag{1.1}$$

Definition 1.1. A norm $\|\cdot\|$ on $\mathbb{R}^n$ is a function that assigns a scalar $\|x\|$ to every $x \in \mathbb{R}^n$ with the following properties:

1. $\|x\| \geq 0$ for all $x \in \mathbb{R}^n$.

2. $\|x\| = 0$ if and only if $x = 0$.

3. $\|\alpha x\| = |\alpha| \, \|x\|$ for all $\alpha \in \mathbb{R}$ and $x \in \mathbb{R}^n$.

4. $\|x + y\| \leq \|x\| + \|y\|$ for all $x, y \in \mathbb{R}^n$. This is known as the Triangle Inequality.

Throughout this thesis we will be working exclusively with the *Euclidean* norm

$$\|x\|_2 = \left(x^T x\right)^{1/2} = \left(\sum_{i=1}^{n} x_i^2\right)^{1/2} \tag{1.2}$$

and so we will drop the subscript, so that $\|x\|$ refers to the Euclidean norm.

Based on the definition of a norm we have the notion of *distance*. In general, the distance between two vectors x and y is given by

$$d(x, y) = \|x - y\|. \tag{1.3}$$

The Euclidean distance between two vectors $x, y \in \mathbb{R}^n$ is given by:

$$\|x - y\|_2 = \sqrt{(x - y)^T (x - y)} = \left(\sum_{i=1}^{n} (x_i - y_i)^2\right)^{1/2}. \tag{1.4}$$

As this is the only distance measure used throughout this thesis, we will again drop the subscript.

A useful inequality that we will apply throughout this thesis is the Cauchy-Schwarz inequality:

Fact 1.2. *(Cauchy-Schwarz Inequality) [9] Let $x, y \in \mathbb{R}^n$. Then*

$$|\langle x, y\rangle| \leq \|x\| \, \|y\|, \tag{1.5}$$

with equality holding if x and y are collinear, i.e., x being a scalar multiple of y or vice versa.

A basic result of analysis is that

Fact 1.3. *All norms in $\mathbb{R}^n$ are equivalent. More precisely, suppose that $\|\cdot\|_a$ and $\|\cdot\|_b$ are norms in $\mathbb{R}^n$, then there exists $\alpha, \beta > 0$ such that*

$$\alpha\|x\|_a \leq \|x\|_b \leq \beta\|x\|_a \quad \forall x \in \mathbb{R}^n.$$

1.2 Existence of Maximizers and Minimizers

Definition 1.4. The domain of a function f, denoted **dom** (f), is given by

$$\mathbf{dom}\ (f) = \{x \in \mathbb{R}^n |\ f(x) < +\infty\}.$$

Example 1.5. The domain of some common functions:

- **dom** $(x^2) = \mathbb{R}$
- **dom** $(\sqrt{x}) = \{x \in \mathbb{R}\ \mid x \geq 0\}$
- **dom** $(\frac{1}{x}) = \{x \in \mathbb{R}\ \mid x \neq 0\}$

Definition 1.6. We say that the function f is proper if its domain is nonempty and $f > -\infty$.

Definition 1.7. The α-sublevel set of a function f is the set

$$\{x \in \mathbf{dom}\ f \,|f(x) \leq \alpha\} \tag{1.6}$$

and the epi-graph of f is the set

$$\mathbf{epi}\ (f) = \left\{(x, r) \in \mathbb{R}^{n+1} \,|x \in \mathbf{dom}\ f \text{ and } f(x) \leq r\right\}. \tag{1.7}$$

The indicator function associated with $S \subseteq \mathbb{R}^n$ is defined by

$$\iota_S(x) = \begin{cases} 0, & \text{if } x \in S; \\ +\infty, & \text{otherwise.} \end{cases}$$

Associated with $f : \mathbb{R}^n \to [-\infty, +\infty]$ we let

$$\text{argmin} f = \{x \in \mathbb{R}^n \mid f(x) = \min f\} \tag{1.8}$$

and

$$\text{argmax} f = \{x \in \mathbb{R}^n \mid f(x) = \max f\}. \tag{1.9}$$

Recall that

Definition 1.8. A set $S \subset \mathbb{R}^n$ is compact if it is closed and bounded.

A continuous function on a compact set always has maximizers and minimizers.

Theorem 1.9. *(Extreme Value Theorem) [9] Assume that $f : \mathbb{R}^n \to \mathbb{R}$ is continuous and $\mathcal{X} \subset \mathbb{R}^n$ is compact. Then*

$$\operatorname{argmin}(f + \iota_{\mathcal{X}}) = \left\{ x \in \mathcal{X} \;\middle|\; f(x) = \min_{\mathcal{X}} f \right\} \neq \emptyset \tag{1.10}$$

and

$$\operatorname{argmax}(f + \iota_{\mathcal{X}}) = \left\{ x \in \mathcal{X} \;\middle|\; f(x) = \max_{\mathcal{X}} f \right\} \neq \emptyset. \tag{1.11}$$

Building on the vector space $\mathbb{R}^n$, the next section focuses on real matrices; that is, the set of all $p \times q$ matrix with entries $a_{ij} \in \mathbb{R}$. Given the vector space $\mathbb{R}^n$, the Euclidean norm, and the theories of linear algebra presented in the next section, we can define many optimization problems of interest.

1.3 Matrix Algebra

For any matrix A, we denote its i, j^{th} element by $A_{i,j}$. The transpose of A, denoted A^T is defined by $A^T_{i,j} = A_{j,i}$. For an $n \times n$ matrix, we write $A \in \mathbb{R}^{n \times n}$.

Definition 1.10. Let A be a square matrix. We say that A is symmetric if

$$A = A^T. \tag{1.12}$$

The space of all $n \times n$ symmetric matrices is denoted $\mathcal{S}^n$.

Definition 1.11. The $n \times n$ identity matrix is a diagonal matrix whose diagonal elements are equal to 1, and is denoted I or sometimes I_n. That is, $I_{j,j} = 1$ for $j = 1, \ldots, n$ and $I_{i,j} = 0$ for all $i \neq j$.

Definition 1.12. The trace of an $n \times n$ square matrix A is defined to be the sum of the elements on the main diagonal. That is,

$$\operatorname{trace}(A) = \operatorname{tr}(A) = \sum_{i=1}^{n} A_{i,i}. \tag{1.13}$$

Definition 1.13. For $A, B \in \mathcal{S}^n$ we define the inner product by

$$\langle A, B \rangle = \operatorname{tr}(AB). \tag{1.14}$$

Fact 1.14. *([24] p.29 and p.45) For any* $A, B \in \mathcal{S}^n$ *we have*

$$\operatorname{tr}(AB) = \operatorname{tr}(BA) \tag{1.15}$$

and

$$\operatorname{tr}(A + B) = \operatorname{tr}(A) + \operatorname{tr}(B). \tag{1.16}$$

Definition 1.15. A matrix $A \in \mathcal{S}^n$ is said to be positive definite if

$$x^T Ax > 0 \qquad \text{for all } x \in \mathbb{R}^n \setminus \{0\} \tag{1.17}$$

and we write $A \succ 0$. The matrix $A \in \mathcal{S}^n$ is negative definite if

$$x^T Ax < 0 \qquad \text{for all } x \in \mathbb{R}^n \setminus \{0\} \tag{1.18}$$

and we write $A \prec 0$.

Definition 1.16. A matrix $A \in \mathcal{S}^n$ is said to be positive semidefinite if

$$x^T Ax \geq 0 \qquad \text{for all } x \in \mathbb{R}^n \tag{1.19}$$

and we write $A \succeq 0$. The matrix $A \in \mathcal{S}^n$ is negative semidefinite if

$$x^T Ax \leq 0 \qquad \text{for all } x \in \mathbb{R}^n \tag{1.20}$$

and we write $A \preceq 0$.

Definition 1.17. The set of a positive semidefinite symmetric matrices is denoted $\mathcal{S}^n_+$.

Definition 1.18. Let A be any $n \times n$ square matrix. A principal submatrix of A is any $m \times m$ submatrix of A obtained by deleting $n - m$ rows and corresponding columns.

Example 1.19. Let $A = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$. One example of a principal submatrix of A is obtained by deleting the second and third columns and rows:

$$\begin{bmatrix} a & d \\ m & p \end{bmatrix}. \tag{1.21}$$

Another principal submatrix might delete just the fourth column and row, or the first and third, etc.

Proposition 1.20. *If $A \in \mathcal{S}^n$ is positive semidefinite, then every principal submatrix of A is symmetric positive semidefinite.*

Proof. Let A_p be any principal submatrix of A. First, note that since we have deleted corresponding rows and columns of A, a symmetric matrix, the submatrix A_p must be symmetric.

Now, let x_p be any nonzero vector of size p. "Enlarge" x_p to a vector x of size n by inserting zeros into the positions corresponding to the rows (and columns) of A which were deleted to form A_p. Then we have that

$$x_p^T A_p x_p = x^T A x \geq 0. \tag{1.22}$$

Hence, A_p is positive semidefinite. □

Corollary 1.21. *Let $A \in \mathcal{S}^n$ be positive semidefinite. Then for each $i = 1, \ldots, n-1$ we have*

$$\begin{bmatrix} A_{i,i} & A_{i,n} \\ A_{i,n} & A_{n,n} \end{bmatrix} \succeq 0. \tag{1.23}$$

This follows directly from Proposition 1.20.

Definition 1.22. The eigenvalues of an $n \times n$ matrix A are the values $\lambda_1, \ldots, \lambda_n$ (not necessarily distinct) such that $\det(A - \lambda I) = 0$.

Fact 1.23. *([24] p.278) The determinant and trace of an $n \times n$ matrix A can be expressed in terms of the eigenvalues of A by:*

$$\det(A) = \prod_{i=1}^{n} \lambda_i \qquad \operatorname{tr}(A) = \sum_{i=1}^{n} \lambda_i. \tag{1.24}$$

Fact 1.24. *([24] p.309) An $n \times n$ matrix A is positive semidefinite if and only if all of the eigenvalues of A are nonnegative.*

Proposition 1.25. *If A is a 2×2 symmetric matrix, then A is positive semidefinite if and only if $\det(A) \geq 0$ and $\operatorname{trace}(A) \geq 0$.*

Proof. Suppose first that A is positive semidefinite. Then by Fact 1.24 we have $\lambda_1, \lambda_2 \geq 0$. Then by Fact 1.23 we have

$$\det(A) = \prod_{i=1}^{2} \lambda_i = \lambda_1 \lambda_2 \geq 0 \tag{1.25}$$

and

$$\operatorname{trace}(A) = \sum_{i=1}^{2} \lambda_i = \lambda_1 + \lambda_2 \geq 0 \tag{1.26}$$

as required.

Now, suppose that $\det(A) \geq 0$ and $\operatorname{trace}(A) \geq 0$. By Fact 1.23 we have that

$$\det(A) = \prod_{i=1}^{2} \lambda_i = \lambda_1 \lambda_2 \geq 0 \tag{1.27}$$

so that λ_1 and λ_2 must have the same parity. Furthermore, by Fact 1.23 we also have

$$\operatorname{trace}(A) = \sum_{i=1}^{2} \lambda_i = \lambda_1 + \lambda_2 \geq 0 \tag{1.28}$$

and thus we can conclude that $\lambda_1, \lambda_2 \geq 0$ and hence A is positive semidefinite. $\square$

Fact 1.26. *(The Cholesky Factorization, [18] page 154.) Let $A \in \mathcal{S}^n$ be positive definite. Then $A = L^T L$, where L is a nonsingular upper triangular matrix.*

1.4 Convex Analysis

Recent applications of optimization are focused strongly on finding optimal solutions to non-convex problems. The approach to solving non-convex problems however, often involves creating convex relaxation problems which can then be solved by existing optimization methods. The following section provides a brief overview of convex analysis.

1.4.1 Convex Sets and Convex Functions

Definition 1.27. A set $C \subseteq \mathbb{R}^n$ is affine if, for any $x_1, x_2 \in C$ and $\theta \in \mathbb{R}$ we have

$$\theta x_1 + (1-\theta) x_2 \in C. \tag{1.29}$$

That is, an affine set contains the linear combination of any two points in the set, provided the coefficients sum to one. A point of the form $\theta_1 x_1 + \cdots + \theta_k x_k$, where $\theta_1 + \cdots + \theta_k = 1$, is called an *affine combination* of the points $x_1, \ldots, x_k$.

Definition 1.28. A set $C \subseteq \mathbb{R}^n$ is convex if, for any $x_1, x_2 \in C$ and $0 \leq \theta \leq 1$ we have

$$\theta x_1 + (1-\theta) x_2 \in C. \tag{1.30}$$

Graphically, a set C is convex if the line segment between any two points in C is also contained in C, see Figure 1.1. Every affine set is convex, since if a set contains the line through any two points, it must obviously contain the line between those two points. Given $\theta_i \geq 0$ with $\theta_1 + \cdots + \theta_k = 1$ and points $x_1, \ldots, x_k \in C$, we say that the point $\theta_1 x_1 + \cdots + \theta_k x_k$ is a *convex combination* of the points $x_1, \ldots, x_k$.

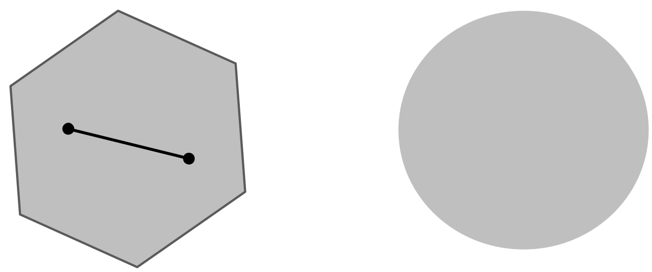

Figure 1.1: Examples of convex sets.

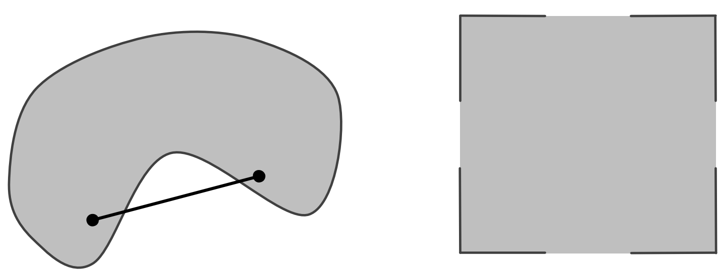

Figure 1.2: Examples of nonconvex sets.

Definition 1.29. The convex hull of a set C is the set of all convex combinations of points in C:

$$\mathbf{conv}\ (C) = \left\{ \sum_{i=1}^{k} \theta_i x_i \;\middle|\; x_i \in C,\ \theta_i \geq 0,\ i = 1, \ldots, k,\ \sum_{i=1}^{k} \theta_i = 1, k \geq 1 \right\}. \tag{1.31}$$

The convex hull of the set C is the smallest convex set that contains C. That is, if D is any convex set such that $C \subseteq D$ then $\mathbf{conv}\ (C) \subseteq D$.

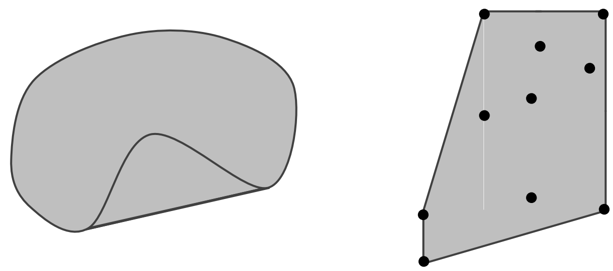

Figure 1.3: The convex hull of some nonconvex sets.

Fact 1.30. [22] *If* $C \subseteq \mathbb{R}^n$ *is compact, i.e. closed and bounded, then* $\operatorname{conv}(C)$ *is compact.*

Corollary 1.31. *Given* $x_1, x_2, \ldots, x_k$ *in* $\mathbb{R}^n$, $\operatorname{conv}\{x_1, \ldots, x_k\}$ *is compact.*

Definition 1.32. A set $K \subseteq \mathbb{R}^n$ is called a cone if, for any $x \in K$ and $\theta \geq 0$ we have $\theta x \in K$.

A set K is a *convex cone* if it is convex and a cone; that is, for any $x_1, x_2 \in K$ and $\theta_1, \theta_2 \geq 0$ we have

$$\theta_1 x_1 + \theta_2 x_2 \in K. \tag{1.32}$$

Example 1.33.

- $\mathbb{R}_+ = \{x \mid x \geq 0\}$ is a convex cone in $\mathbb{R}$.
- $\mathbb{R}^n_+ = \{(x_1, \ldots, x_n) \mid x_1 \geq 0, \ldots, x_n \geq 0\}$ is a convex cone in $\mathbb{R}^n$.
- $\mathcal{S}^n_+ = \{A \mid A \text{ is an } n \times n \text{ matrix, } A \text{ is positive semidefinite}\}$ is a convex cone.

A point of the form $\theta_1 x_1 + \cdots + \theta_k x_k$, where $\theta_i \geq 0$ for $i = 1, \ldots, k$ is called a *conic combination* of the points $x_1, \ldots, x_k$.

Definition 1.34. Let $\mathcal{K} \subseteq \mathbb{R}^n$ be a cone. The dual cone of $\mathcal{K}$, denoted $\mathcal{K}^*$ is the set

$$\mathcal{K}^* = \left\{ y \in \mathbb{R}^n \,\middle|\, y^T x \geq 0 \ \forall x \in \mathcal{K} \right\} \tag{1.33}$$

Definition 1.35. A hyperplane is a set of the form

$$\left\{ x \in \mathbb{R}^n \,\middle|\, a^T x = b \right\} \tag{1.34}$$

where $a \in \mathbb{R}^n$, $a \neq 0$ and $b \in \mathbb{R}$.

Analytically, a hyperplane is the solution set of a nontrivial linear equation. Geometrically, the hyperplane $H = \left\{ x \,\middle|\, a^T x = b \right\}$ can be interpreted as the set of points with constant inner product to a given vector a; that is, a hyperplane with normal vector a. Note that a hyperplane is affine (and thus convex).

Definition 1.36. A (closed) halfspace is a set of the form

$$\left\{ x \in \mathbb{R}^n \,\middle|\, a^T x \le b \right\} \tag{1.35}$$

where $a \neq 0$.

A hyperplane divides $\mathbb{R}^n$ into two halfspaces (see Figure 1.4). Analytically, a halfspace is the solution set of one nontrivial linear inequality. Halfspaces are convex, but not affine.

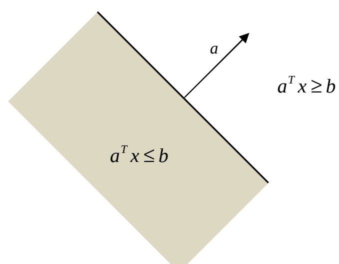

Figure 1.4: A hyperplane with normal vector a divides $\mathbb{R}^2$ into two halfspaces.

Some frequently seen convex sets are listed below:

– The *closed norm ball* in $\mathbb{R}^n$ with center x_c and radius r is given by

$$B_r[x_c] = \{x \in \mathbb{R}^n \mid \|x - x_c\| \le r\} \tag{1.36}$$

where $r > 0$ and $\|\cdot\|$ denotes any norm on $\mathbb{R}^n$.

– The *norm cone* associated with any norm $\|\cdot\|$ is given by

$$\{(x,t) \mid \|x\| \le t, x \in \mathbb{R}^n, t \in \mathbb{R}\} \subseteq \mathbb{R}^{n+1}. \tag{1.37}$$

The norm cone is a convex cone. See Figure 1.5 for an example of the norm cone associated with the Euclidean norm.

– A *polyhedron* is the solution set of a finite number of linear equalities and inequalities, and is of the form

$$\left\{x \in \mathbb{R}^n \,\middle|\, a_j^T x \le b_j,\ j = 1, \ldots, m,\ c_k^T x \le d_k,\ k = 1, \ldots, p\right\}. \tag{1.38}$$

Graphically, a polyhedron is the intersection of a finite number of halfspaces and hyperplanes.

Note that the norm cone associated with the Euclidean norm is referred to as **the second-order cone** or **Lorentz cone**.

Definition 1.37. The Lorentz cone is the cone in $\mathbb{R}^n$ defined as

$$\mathcal{L} = \left\{x \in \mathbb{R}^n \,\middle|\, x_n^2 \ge \sum_{i=1}^{n-1} x_i^2,\ x_n \ge 0\right\}. \tag{1.39}$$

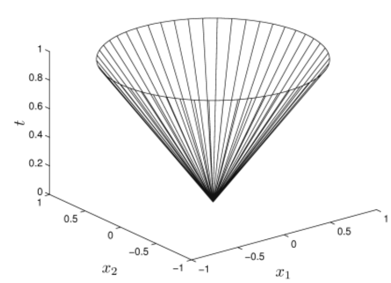

Figure 1.5: The Euclidean norm cone in $\mathbb{R}^3$.

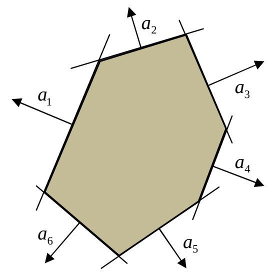

Figure 1.6: A polyhedron.

Definition 1.38. A function $f : \mathbb{R}^n \to (-\infty, \infty]$ is said to be a convex function if **dom** f is a convex set and, for all $x, y \in$ **dom** f and $0 < \theta < 1$ we have

$$f\left(\theta x + (1-\theta)\, y\right) \leq \theta f(x) + (1-\theta) f(y). \tag{1.40}$$

Graphically, a function f is convex if the line segment joining $(x, f(x))$ and $(y, f(y))$ lies above the graph of f, (see Figure 1.7). A function f is *strictly convex* if strict inequality holds in (1.40) whenever $x \neq y$ and $0 < \theta < 1$. We say that the function f is *concave* if $-f$ is convex, and *strictly concave* if $-f$ is strictly convex.

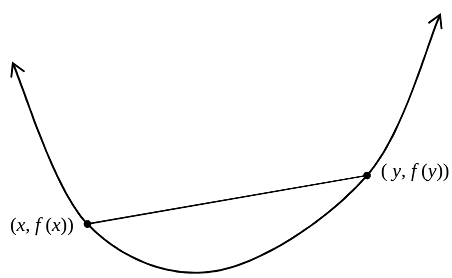

Figure 1.7: A convex function.

Remark 1.39. A function f is convex if and only if **epi** (f) is a convex set.

Fact 1.40. *Every norm is a convex function on* $\mathbb{R}^n$*.*

Proof. Let $0 \leq \theta \leq 1$, and consider

$$\begin{aligned} f(\theta x + (1-\theta)y) &= \|\theta x + (1-\theta)y\| && (1.41)\\ &\leq \|\theta x\| + \|(1-\theta)y\| && (1.42)\\ &= \theta\,\|x\| + (1-\theta)\,\|y\| && (1.43)\\ &= \theta f(x) + (1-\theta)f(y). && (1.44) \end{aligned}$$

Hence, f is convex.

Alternatively, simply note that the epigraph of the function $f(x) = \|x\|$ is given by

$$\textbf{epi}\,(f) = \{(x,t) \mid \|x\| \leq t\} \tag{1.45}$$

which is a convex set in $\mathbb{R}^{n+1}$ (the norm cone), and so by Remark 1.39 f is a convex function. □

Definition 1.41. Let $f : \mathbb{R}^n \to \mathbb{R}^m$ and $x \in \operatorname{int} \textbf{dom}\, f$. If f is differentiable at x, its *Jacobian* matrix $\nabla f(x) \in \mathbb{R}^{m\times n}$ is given by

$$\nabla f(x)_{ij} = \frac{\partial f_i(x)}{\partial x_j} \tag{1.46}$$

for all $i = 1,\ldots,m$, $j = 1,\ldots,n$.

Definition 1.42. For a real-valued function $f : \mathbb{R}^n \to \mathbb{R}$, if f is twice differentiable at x, its *Hessian* matrix $\nabla^2 f(x) \in \mathbb{R}^{n\times n}$ is given by

$$\nabla^2 f(x)_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j} \tag{1.47}$$

for all $i, j = 1,\ldots,n$.

Convexity of a differentiable function can also be established using the following conditions:

Fact 1.43. *(First order convexity condition)*
Suppose f is differentiable on an open set $O \subseteq \mathbb{R}^n$. Then f is convex on O if and only if O is convex and

$$f(y) \geq f(x) + \nabla f(x)^T(y-x) \tag{1.48}$$

for all $x, y \in O$ ([22] page 46).

Fact 1.44. *(Second order convexity condition)*
Assume f is twice differentiable on an open set $O \subseteq \mathbb{R}^n$. Then f is convex on O if and only if O is convex and $\nabla^2 f$ is positive semidefinite; i.e.,

$$\nabla^2 f(x) \succeq 0 \tag{1.49}$$

for every $x \in O$ ([22] page 46).

Fact 1.45. *The function*

$$f_i(x) = w_i \left\| x - x^i \right\|^2 \tag{1.50}$$

is a convex function on $\mathbb{R}^n$, where $w_i > 0 \in \mathbb{R}$ and $x^i \in \mathbb{R}^n$ is fixed.

This follows from the second order convexity condition, and the fact that $\nabla^2 f_i(x) = 2w_i I \succeq 0$.

Definition 1.46. A vector $v \in \mathbb{R}^n$ is said to be a subgradient of the convex function f at the point x if

$$f(y) \geq f(x) + \langle v, y - x \rangle, \qquad \forall y \in \mathbf{dom}\ f. \tag{1.51}$$

The set of all subgradients of f is called the subdifferential of f at x, and is denoted $\partial f(x)$.

A subdifferential mapping, being a surrogate for derivative, is fundamentally important in convex analysis and optimization.

Definition 1.47. A vector x^* is said to be normal to the convex set C at the point $x \in C$ if

$$\langle y - x, x^* \rangle \leq 0 \qquad \forall y \in C. \tag{1.52}$$

The set of all vectors x^* that are normal to C at the point x is called the Normal Cone to C at x, and is denoted $N_C(x)$. If $x \notin C$, then $N_C(x) = \emptyset$.

Recall that for $C \subseteq \mathbb{R}^n$, x is an interior point of C if there exists $\epsilon > 0$ such that $B_\epsilon(x) \subseteq C$. The set of interior points of C will be denoted by $\text{int}(C)$.

Example 1.48. Let $C \subseteq \mathbb{R}^n$. Immediately from the definition of $\partial \iota_C$, we have

$$\partial \iota_C = N_C.$$

Moreover, if $x \in \text{int}(C)$, then $\partial \iota_C(x) = \{0\}$, i.e. $N_C(x) = \{0\}$.

Definition 1.49. Let $f : \mathbb{R}^n \to [-\infty, \infty]$, and let x be a point such that $|f(x)| < \infty$. The one-sided directional derivative of f at x with respect to a vector y is defined to be the limit

$$f'(x; y) := \lim_{\lambda \downarrow 0} \frac{f(x + \lambda y) - f(x)}{\lambda} \tag{1.53}$$

if it exists (here $\pm\infty$ are allowed as limits).

Definition 1.50. Let $f : \mathbb{R}^n \to [-\infty, \infty]$, and let x be a point such that $|f(x)| < \infty$. We say that f is differentiable at x if and only if there exists a vector x^* with the property

$$\lim_{z \to x} \frac{f(z) - f(x) - \langle x^*, z - x \rangle}{\|z - x\|} = 0. \tag{1.54}$$

Such an x^*, if it exists, is called the gradient of f at x and is denoted $\nabla f(x)$.

Fact 1.51. *If f is differentiable at x, then $f'(x; y)$ exists and is a linear function of y for all y. In particular,*

$$f'(x; y) = \langle \nabla f(x), y \rangle \qquad \forall y \in \mathbb{R}^n. \tag{1.55}$$

Fact 1.52. [2, Proposition 2.3.2] *Let $\bar{x} \in int(C)$, $C \subset \mathbb{R}^n$. Suppose that continuous functions $g_0, g_1, \ldots, g_m : C \to \mathbb{R}$ are differentiable at $\bar{x}$, that g is the max function*

$$g(x) = \max_{i=0,\ldots,m} g_i(x) \tag{1.56}$$

and define the index set $I = \{i \,|\, g_i(\bar{x}) = g(\bar{x})\}$. Then for all directions d in $\mathbb{R}^n$, the directional derivative of g is given by

$$g'(\bar{x}; d) = \max_{i \in I} \{\langle \nabla g_i(\bar{x}), d \rangle\}. \tag{1.57}$$

It turns out that the directional derivative of a convex function can be used to compute its subdifferential.

Fact 1.53. [19, Theorem 23.2] *Let $f : \mathbb{R}^n \to (-\infty, +\infty]$ be convex and $x \in$ **dom** f. Then $f'(x; y)$ exists for every $y \in \mathbb{R}^n$. Moreover,*

$$\partial f(x) = \{v \in \mathbb{R}^n |\ f'(x; y) \geq \langle v, y \rangle\ \forall y \in \mathbb{R}^n\}.$$

1.4.2 Minimizers or Maximizers of Convex Functions

Theorem 1.54. [19, page 264] *For a convex function f, we have that*

$$x \in \mathrm{argmin} f \;\Leftrightarrow\; 0 \in \partial f(x). \tag{1.58}$$

Remark 1.55. We say that $x \in D$ is an extremal point of D if

$$x = \lambda y + (1-\lambda) z \tag{1.59}$$

with $0 < \lambda < 1$, $y, z \in D$ implies that $y = z$.

Theorem 1.56. [22, Theorem 32.3] *Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex, and $K \subseteq \mathbb{R}^n$ be compact and convex. Then*

$$\max_{x \in K} f(x) = \max_{x \in ext(K)} f(x) \tag{1.60}$$

where $ext(K)$ denotes the extremal points of K.

1.5 Nonsmooth Analysis

Since maximin problems are nonconvex in general, in Subsection 1.5.1 and 1.5.2 below we introduce the concept of a Lipschitz function and nonsmooth analysis. They are used in Section 3.2 to derive necessary optimality conditions for the solutions of nonconvex maximin optimization problem.

1.5.1 Locally Lipschitz Functions

Definition 1.57. Given a function $f : \mathbb{R}^n \to \mathbb{R}$, we say that f is locally Lipschitz continuous at a point x_0 if there exists $\delta > 0$ and $\Lambda > 0$ such that whenever $\|x - x_0\| < \delta$ and $\|y - x_0\| < \delta$ we have

$$|f(x) - f(y)| \leq \Lambda \|x - y\|. \tag{1.61}$$

Fact 1.58. *If f is locally Lipschitz at x_0, then $-f$ is locally Lipschitz at x_0.*

The above fact follows directly from

$$|-f(x) - (-f(y))| = |-(f(x) - f(y))| = |f(x) - f(y)|. \tag{1.62}$$

Fact 1.59. *If f_1, f_2 are locally Lipschitz at x_0, then*

$$\max\{f_1, f_2\} \tag{1.63}$$

and

$$\min\{f_1, f_2\} \tag{1.64}$$

are locally Lipschitz at x_0.

Proof. Since f_1 is locally Lipschitz at x_0, there exists $\delta_1 > 0$ and $\Lambda_1 > 0$ such that

$$|f_1(x) - f_1(y)| \le \Lambda_1 \|x - y\| \tag{1.65}$$

whenever $\|x - x_0\| < \delta_1$ and $\|y - x_0\| < \delta_1$. Similarly, there exists $\delta_2 > 0$ and $\Lambda_2 > 0$ such that

$$|f_2(x) - f_2(y)| \le \Lambda_2 \|x - y\| \tag{1.66}$$

whenever $\|x - x_0\| < \delta_2$ and $\|y - x_0\| < \delta_2$.

Set $\delta = \min\{\delta_1, \delta_2\}$, and $\Lambda = \max\{\Lambda_1, \Lambda_2\}$ so that whenever $\|x - x_0\| < \delta$ and $\|y - x_0\| < \delta$ we have

$$-\Lambda \|x - y\| \le f_1(x) - f_1(y) \le \Lambda \|x - y\| \tag{1.67}$$

$$-\Lambda \|x - y\| \le f_2(x) - f_2(y) \le \Lambda \|x - y\|. \tag{1.68}$$

From this we have

$$f_1(x) - \max\{f_1(y), f_2(y)\} \le \Lambda \|x - y\| \tag{1.69}$$

$$f_2(x) - \max\{f_1(y), f_2(y)\} \le \Lambda \|x - y\|. \tag{1.70}$$

Combining the above two equations we have

$$\begin{aligned} &\max\{f_1(x), f_2(x)\} - \max\{f_1(y), f_2(y)\} \\ &\qquad = \max\{f_1(x) - \max\{f_1(y), f_2(y)\}, f_2(x) - \max\{f_1(y), f_2(y)\}\} \\ &\qquad \le \Lambda \|x - y\|. \end{aligned} \tag{1.71}$$

Similarly,

$$-\Lambda \|x - y\| \le f_1(x) - f_1(y) \;\Rightarrow\; f_1(y) - f_1(x) \le \Lambda \|x - y\| \tag{1.72}$$

$$-\Lambda \|x - y\| \le f_2(x) - f_2(y) \;\Rightarrow\; f_2(y) - f_2(x) \le \Lambda \|x - y\| \tag{1.73}$$

$$\tag{1.74}$$

so that

$$f_1(y) - \max\{f_1(x), f_2(x)\} \le \Lambda \|x - y\| \tag{1.75}$$

$$f_2(y) - \max\{f_1(x), f_2(x)\} \le \Lambda \|x - y\| \tag{1.76}$$

and thus

$$\begin{aligned} &\max\{f_1(y), f_2(y)\} - \max\{f_1(x), f_2(x)\} \le \Lambda \|x - y\| \\ &\Rightarrow \max\{f_1(x), f_2(x)\} - \max\{f_1(y), f_2(y)\} \ge -\Lambda \|x - y\|. \end{aligned} \tag{1.77}$$

Hence, whenever $\|x - x_0\| < \delta$ and $\|y - x_0\| < \delta$ we have that

$$|\max\{f_1(y), f_2(y)\} - \max\{f_1(x), f_2(x)\}| \leq \Lambda \|x - y\| \tag{1.78}$$

which shows that $\max\{f_1, f_2\}$ is locally Lipschitz at x_0.

Now,

$$\min\{f_1, f_2\} = -\max\{-f_1, -f_2\}. \tag{1.79}$$

Using Fact 1.58 we know that $-f_1, -f_2$ are both locally Lipschitz at x_0. Then $\max\{-f_1, -f_2\}$ is locally Lipschitz at x_0 by the above proof. Applying Fact 1.58 once more, we have that $-\max\{-f_1, -f_2\}$ and thus $\min\{f_1, f_2\}$ is locally Lipschitz at x_0. □

Remark 1.60. Fact 1.59 can also be seen by using

$$\max\{f_1, f_2\} = \frac{f_1 + f_2}{2} + \frac{|f_1 - f_2|}{2},$$

$$\min\{f_1, f_2\} = \frac{f_1 + f_2}{2} - \frac{|f_1 - f_2|}{2}.$$

Fact 1.61. *Assume that f_i is locally Lipschitz at x_0 for $i = 1, \ldots, m$. Then*

$$\max\{f_1, \ldots, f_m\} \tag{1.80}$$

and

$$\min\{f_1, \ldots, f_m\} \tag{1.81}$$

are locally Lipschitz at x_0.

Proof. This follows directly from applying induction to the results of Fact 1.59. □

1.5.2 Subdifferentials and Critical Points

Let $O \subseteq \mathbb{R}^n$ be an open set. For a locally Lipschitz function $f : O \to \mathbb{R}$, and $x \in O$, we define its Clarke directional derivative with respect to $v \in \mathbb{R}^n$ by

$$f^o(x; v) = \limsup_{y \to x, t \downarrow 0} \frac{f(y + tv) - f(y)}{t} \tag{1.82}$$

and the Clarke subdifferential

$$\partial^o f(x) = \{x^* \in \mathbb{R}^n \mid f^o(x; v) \geq \langle x^*, v \rangle \ \forall v \in \mathbb{R}^n\}. \tag{1.83}$$

The Clarke subdifferential plays a fundamental role in optimization problems involving nonconvex functions. Some key properties are:

Theorem 1.62. *Let $f, g : O \subset \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz and $x \in O$. Then*

1. $\partial^o f(x)$ *is compact and convex ([7] page 40).*
2. $\partial^o (f+g)(x) \subseteq \partial^o f(x) + \partial^o g(x)$ *([7] Proposition 2.3.3).*
3. $\partial^o(-f)(x) = -\partial^o f(x)$ *([7] Proposition 2.3.1).*
4. *If f is a convex function on O, then $\partial^o f = \partial f$ ([7] Proposition 2.2.7).*
5. *If f is continuously differentiable on O, then $\partial^o f(x) = \{\nabla f(x)\}$ ([7] Proposition 2.2.7).*

Theorem 1.63. [23, Proposition 7.4.7] *If f is locally Lipschitz about $\bar{x}$ and $\bar{x}$ is a local minimizer or a local maximizer of f, then $0 \in \partial^o f(\bar{x})$.*

Theorem 1.64. [23, Proposition 7.4.7] *Let I be a finite set, and for all $i \in I$ let $g_i : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz continuous around $\bar{x}$. Let g be the max-function defined as*

$$g(x) = \max \{g_i(x) \mid i \in I\} \tag{1.84}$$

and define the active index set at $\bar{x}$ by

$$I(\bar{x}) = \{i \in I \mid g_i(\bar{x}) = g(\bar{x})\} . \tag{1.85}$$

Suppose that g_i is differentiable at $\bar{x}$. Then the Clarke subdifferential of g at $\bar{x}$ is given by

$$\partial^o g(\bar{x}) = conv \{\nabla g_i(\bar{x}) \mid i \in I(\bar{x})\} . \tag{1.86}$$

1.6 Some Convex Optimization Problems

In this section we give the general and standard forms of a semidefinite programming optimization problem (SDP) and a second-order cone programming problem (SOCP), which we will use in section 2.2. For a convex cone $K \subset \mathbb{R}^p$ and $x \in \mathbb{R}^p$, we write $x \succeq_K 0$ if $x \in K$.

Definition 1.65. ([4]) A conic form program (or conic program) is an optimization problem which has a linear objective function and one inequality constraint function

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Fx + G \succeq_K 0 \\ & Ax = b \end{aligned} \tag{1.87}$$

where $c, x \in \mathbb{R}^p$, $A \in \mathbb{R}^{m \times p}$, $b \in \mathbb{R}^m$, $F \in \mathcal{S}^p$, $G \in \mathbb{R}^p$ and K is a convex cone in $\mathbb{R}^p$.

Definition 1.66. When K is $\mathcal{S}^n_+$, the cone of positive semidefinite $n \times n$ matrices, the associated conic form problem is called a semidefinite program, and has the form

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & x_1 F_1 + \cdots + x_m F_m + G \succeq 0 \\ & Ax = b \end{aligned} \tag{1.88}$$

where $c, x \in \mathbb{R}^m$, $G, F_1, \ldots, F_m \in \mathcal{S}^n$ and $A \in \mathbb{R}^{p \times m}$.

A **standard SDP in inequality form** is

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & x_1 A_1 + \cdots + x_m A_m \preceq B \end{aligned} \tag{1.89}$$

where $A_1, \ldots, A_m, B \in \mathcal{S}^n$, $c, x \in \mathbb{R}^m$.
A **standard SDP** is in the following form:

$$\begin{aligned} \min \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \langle A_i, X \rangle = b_i \quad i = 1, \ldots, m \\ & X \succeq 0 \end{aligned} \tag{1.90}$$

where $C, X, A_i \in \mathcal{S}^n$ and $b_i \in \mathbb{R}$. One can show that (1.90) has a dual given by

$$\begin{aligned} \max \quad & b^T x \\ \text{s.t.} \quad & x_1 A_1 + \cdots + x_m A_m \preceq C \end{aligned} \tag{1.91}$$

which is in the form of (1.89), see [4, Example 5.11], [29].

Definition 1.67. ([4]) The second-order cone program (SOCP) is of the form

$$\begin{aligned} \min \quad & f^T x \\ \text{s.t.} \quad & \|A_i x + b_i\| \leq c_i^T x + d_i, \quad i = 1, \ldots, m \\ & Fx = g \end{aligned} \tag{1.92}$$

where $f, x \in \mathbb{R}^n$, $A_i \in \mathbb{R}^{n \times n}$ and $F \in \mathbb{R}^{p \times n}$.

Constraints of the form $\|Ax + b\| \leq c^T x + d$ are called second-order cone constraints because the affinely defined variables $u = Ax + b$ and $t = c^T x + d$ must belong to the second order cone. Note that SOCP is a special case of SDP by using

$$\|A_i x + b_i\| \leq c_i^T x + d_i \quad \Leftrightarrow \quad \begin{pmatrix} (c_i^T x + d)I & A_i x + b_i \\ (A_i x + b_i)^\intercal & (c_i^T x + d) \end{pmatrix} \succeq 0.$$

Further readings on SDP and SOCP may also be found in [32], [27], [29].

1.7 NP-Complete Problems

In this section we will give a brief overview to the concept of complexity theory and NP-completeness. In order to begin discussing complexity of problems however, we first need a few basic definitions.

Definition 1.68. [12] A *problem* is a general question to be answered, usually containing several parameters with values left unspecified. A problem is described by giving a general description of all of its parameters, and a statement of what properties the answer (or solution) is required to satisfy. An *instance* of a problem is obtained by specifying particular values for all the problem parameters.

Definition 1.69. [12] An *algorithm* is a general, step-by-step procedure for solving a problem.

For concreteness, we often think of algorithms as being computer programs written in a precise computer language. We say that an algorithm solves a problem Π if that algorithm can be applied to any instance I of Π and be guaranteed to provide a solution for that instance.

Definition 1.70. [12] The *input length* for an instance I of a problem Π is defined to be the number of symbols in the description of I obtained from the encoding scheme for Π.

Definition 1.71. [12] The *time complexity function* for an algorithm expresses its time requirement by giving, for each possible input length, the largest amount of time needed by the algorithm to solve a problem instance of that size.

Note that by this definition, this function is not well-defined until the encoding scheme to be used for determining input length and the computer model for a particular problem are fixed. However, the particular choices made for these have little effect on the broad distinctions made in the theory of NP-completeness ([12] p.6).

1.7.1 The Theory of NP-Completeness

In this section we discuss measures for the complexity of algorithms, and introduce the concept of NP-complete problems.

Definition 1.72. [12] We say that a function $f(n)$ is $O(g(n))$ whenever there exists a constant c such that

$$|f(n)| \leq c \cdot |g(n)| \tag{1.93}$$

for all values of $n \geq 0$.

Definition 1.73. [12] A *polynomial time algorithm* is defined to be one whose complexity function is $O(p(n))$ for some polynomial function p, where n is used to denote the input length. Any algorithm whose time complexity function can not be so bounded is called an *exponential time algorithm.*

Note that this definition includes certain non-polynomial time complexity functions (like $n^{\log n}$) which are not normally regarded as exponential functions.

Definition 1.74. [12] A problem is called *intractable* if no polynomial time algorithm can possibly solve it.

The idea of *polynomial time reducibility* is integral to the theory of NP-completeness, and refers to reductions for which the transformation can be executed by a polynomial time algorithm. This is significant, since if we have a polynomial time reduction from one problem to another, then this ensures that any polynomial time algorithm for the second problem can be converted into a corresponding polynomial time algorithm for the first problem.

The simplest way to explain the idea of NP-completeness is as follows. First, define two classes of problems:

1. The class of problems with solution algorithms bounded by a polynomial of fixed degree, **P** [8].

2. The class of problems with solutions that are verifiable in polynomial time, **NP**. That is, given a solution S to problem Π, a polynomial time algorithm exists to verify that S is in fact a solution of Π [31].

Fact 1.75. [8] $P \subseteq NP$.

One of the most important open questions in theoretical computer sciences is whether or not P=NP. The notation NP stands for "non-deterministic polynomial time", since the original definition of the class NP is the class of problems which can be solved by a non-deterministic Turing machine in polynomial time [31].

A problem P_i is **NP-hard** if every problem in NP is polynomially reducible to P_i. One can say that

$$\boxed{P_i \text{ is NP-hard} \Leftrightarrow \text{ If } P_i \text{ is solved in a polynomial time, then } P = NP.}$$

A problem P_i is **NP-complete** if it is both NP-hard and an element of NP. In essence, the NP-complete problems are the "hardest" problems in NP [12]. The question of whether NP-complete problems are intractable is still a major open problem in mathematics and computer science (See [31] and references therein).

How to show that a problem P_i is NP-hard? One often uses a **reduction argument**:

1. Confirm that P_i is a decision problem;
2. Take one problem P_j that is already known to be NP-hard or NP-complete;
3. Show that the problem P_j is polynomially reducible to P_i;
4. This shows that P_i is NP-hard.

This is what we will use in the thesis.

1.7.2 Some NP-Hard Problems

Based on the previous section, we see that the simplest way to prove that a problem is NP-complete is to reduce it in polynomial time to another problem which is known to be NP-complete (or NP-hard). In order for this to be possible, we require a compendium of problems which are already known to be NP-complete (or NP-hard). There are many books and papers including lists of problems that have been proven to be NP-complete or NP-hard (for example, see [12], [17], [6]). Here, we provide two examples of NP-hard problems that are integral to the work of this thesis.

Fact 1.76. [6, MP1] *It is NP-hard to determine the solvability of the integer linear program*

$$By = b, \quad y \in \{-1, 1\}^q, \tag{1.94}$$

with $B \in \mathbb{Z}^{p \times q}$, $b \in \mathbb{Z}^p$.

Given an undirected graph $G = (V, E)$ with weights $w_{ij} = w_{ji} = 1$ on the edges $(i, j) \in E$, the **Max Cut Problem** is that of finding the set of vertices $S \subseteq V$ that maximizes the weight of the edges in the cut $(S, \overline{S})$. Put $w_{ij} = 0$ if $(i, j) \notin E$. The weight of a cut $(S, \overline{S})$ is

$$w(S, \overline{S}) = \sum_{i \in S, j \notin S} w_{ij}.$$

The Max Cut problem is to

$$\max_{S \subset V} w(S, \overline{S}).$$

Assume that the graph has q nodes, i.e., $|V| = q$. Define two $q \times q$ symmetric matrices $A = (a_{ij})$ and W respectively by setting $a_{ij} = a_{ji} = w_{ij}$, and

$$W = \mathrm{Diag}(Ae) - A \in \mathbb{Z}^{q \times q}$$

where $e = (1, \ldots, 1)^\intercal \in \mathbb{R}^q$.

Definition 1.77. (See [21, page 309], [27], [14]) The MaxCut problem on a graph with q nodes may be formulated as

$$\max_{y \in \{-1,1\}^q} y^T W y, \tag{1.95}$$

where $W \in \mathbb{Z}^{q \times q}$ and $W \succeq 0$.

The Max Cut problem has long been known to be NP complete.

Fact 1.78. ([14, page.1116], [21]) *The Maxcut problem (1.95) is NP-hard.*

For further details on complexity theory and NP-hard problems, see [12].

1.7.3 The Complexity of Algorithms for Solving SDP and SOCP

To some extent, SDP is very similar to linear programming. Given any $\epsilon > 0$, SDPs can be solved within an additive error of ϵ in polynomial time (ϵ is part of the input, so the running time dependence on ϵ is polynomial in $\log(1/\epsilon)$), [14]. This is can be done through ellipsoid algorithm, polynomial algorithms for convex programming, as well as interior point methods. See [14], [15], [1]. If the feasible region is bounded, then SOCPs can also be solved to arbitrary fixed precision in polynomial time, [13]. For the current state of the art in SDP, see [32].

Currently, available software packages solving SDPs and SOCPs include SeDuMi (self-dual minimization) [25], CVX (a modeling system for disciplined convex programming) [11] and SDPT3.

Chapter 2

The Weighted Maximin Location Problem

2.1 General Problem and Motivation

Definition 2.1. The problem of finding a point x in a closed set $\mathcal{X} \subseteq \mathbb{R}^n$ ($n \geq 1$) that is furthest from a given set of points $x^1, \ldots, x^m$ in $\mathbb{R}^n$ in a weighted maximin sense is given by

$$\boxed{\upsilon_p \;:=\; \max_{x \in \mathcal{X}} f(x) \quad \text{with} \quad f(x) := \min_{i=1,\ldots,m} w_i \left\| x - x^i \right\|^2,} \tag{2.1}$$

where $w_i > 0$. This is known as the *weighted maximin dispersion problem.*

In general $\|\cdot\|$ denotes the Euclidean norm; in the case that we consider another norm we will make the distinction explicit.

In the equal weight case of $w_1 = \cdots = w_m$, (2.1) has the geometric interpretation of finding the largest Euclidean sphere with center in $\mathcal{X}$ and enclosing no given point. To illustrate, let $B_\delta(x) = \{y \mid \|y - x\| < \delta, y \in \mathbb{R}^n\}$, and $f(x) = \min_{i=1,\ldots,m} \left\| x - x^i \right\|^2$.

Example 2.2. Set $r = \sqrt{f(x)}$. Then $B_r(x) \cap \left\{x^1, \ldots, x^m\right\} = \emptyset$.

Proof. Indeed, $r^2 \leq \left\| x - x^i \right\|^2 \;\; \forall i = 1, \ldots, m$ by definition. If $x^{i_0} \in B_r(x)$ then $\left\| x - x^{i_0} \right\| < r$, so $\left\| x - x^{i_0} \right\|^2 < r^2$. Thus we have

$$r^2 \leq \left\| x - x^{i_0} \right\|^2 < r^2 \tag{2.2}$$

which is a contradiction. Hence, $B_r(x) \cap \left\{x^1, \ldots, x^m\right\} = \emptyset$. $\square$

Our interest in the minimax distance problem arises from deterministic function modeling in statistics [16]. We begin with a deterministic computer code which models a complex physical system. It is assumed that the code is very costly to run, and so our goal is to build a computationally cheap surrogate of the computer code. The statistical approach to this problem is to place a prior belief on the function space, $\mathcal{X}$, and then use the weighted maximin distance criteria to choose a set of sites in $\mathcal{X}$ at which to run the computer code, so that we may model the resulting deterministic output as a stochastic process (see [26]).

In general, the weighted maximin problem also has many diverse applications in facility location, spatial management and pattern recognition (see [10], [30] and references therein). One applied example in facility location would be the determining of the location for a new and highly polluting industry within some region $\mathcal{X}$. Suppose that there are m cities within region $\mathcal{X}$, represented by the points $x^1, \ldots, x^m$. One criteria in choosing a location for the facility could be that the amount of pollution reaching any city be minimized. Then the optimization problem (2.1) may be used in choosing the location of the facility, provided we can assume that the amount of pollutant reaching a city is monotonically decreasing function of the distance between the city and the facility.

For ease of notation throughout this thesis we define

$$f_i(x) = w_i \left\| x - x^i \right\|^2 \tag{2.3}$$

where $w_i > 0$ is a fixed constant and $x^i \in \mathbb{R}^n$. Thus, the maximin dispersion problem becomes

$$\max_{x \in \mathcal{X}} f(x) \tag{2.4}$$

where, for all $x \in \mathbb{R}^n$, $i \in \{1, \ldots, m\}$

$$f(x) = \min_{1 \leq i \leq m} f_i(x). \tag{2.5}$$

In particular, we will focus on the case that $\mathcal{X}$ in (2.1) is convex under componentwise squaring:

$$\boxed{\mathcal{X} = \left\{ x \in \mathbb{R}^n \,\middle|\, \left(x_1^2, \ldots, x_n^2, 1\right)^T \in \mathcal{K} \right\},} \tag{2.6}$$

for some closed convex cone $\mathcal{K} \subseteq \mathbb{R}^{n+1}$. We will examine two specific cases: the case where $\mathcal{X} = [-1, 1]^n$ (a box) corresponding to

$$\mathcal{K} = \left\{ y \in \mathbb{R}^{n+1} \left| y_j \leq y_{n+1}, j = 1, \ldots, n \right. \right\}, \tag{2.7}$$

and the case of a unit Euclidean ball $\mathcal{X} = \left\{ x \in \mathbb{R}^n \;\middle|\; \|x\|^2 \leq 1 \right\}$ corresponding to

$$\mathcal{K} = \left\{ y \in \mathbb{R}^{n+1} \,|y_1 + \cdots + y_n \leq y_{n+1} \right\}. \tag{2.8}$$

Even in the simple case of a box, $\mathcal{X} = [-1, 1]^n$, the function $f(x)$ can become complex quite quickly. Figure 2.1 shows $f(x)$ for two sets of points in $\mathcal{X}$; the set $S = \{(-1,-1), (-1,1), (1,-1), (1,1), (0,0)\}$ and the set T, a randomly generated set of 10 points in $\mathcal{X}$.

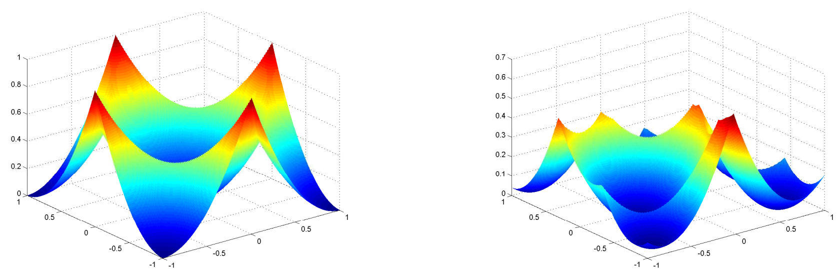

Figure 2.1: Plots of $f(x)$ for $\mathcal{X} = [-1, 1]^2$, with the set S on the left and set T on the right.

Remark 2.3. The function

$$f(x) := \min_{i=1,\ldots,m} w_i \left\| x - x^i \right\|^2 \tag{2.9}$$

in Equation (2.1) is neither convex nor concave whenever $m \geq 2$ and $x^i \neq x^j$.

This is illustrated in Figure 2.1.

Proposition 2.4. *The optimal value of the maximin dispersion problem satisfies*

$$v_p > 0.$$

Proof. Take $x \in \mathcal{X}$ with $x \neq x^i$ for $i = 1, \ldots, m$. We have $f(x) > 0$. Then $v_p \geq f(x) > 0$. □

Since the weighted maximin dispersion problem is a *nonconvex problem*, we cannot apply convex programming methods. The goal of this thesis then is to consider convex relaxations of the problem, the solutions of which can be found using known techniques. The solutions to the relaxation problems can then provide a nontrivial lower bound for the original nonconvex problem.

2.2 Convex Relaxations

In this section we give several alternative formulations of the original problem (2.1) as well as defining the convex relaxation problems. First note that

Proposition 2.5. *If we let*

$$A^i = \begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix} \in \mathcal{S}^{n+1}$$

and

$$X = \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix} \in \mathcal{S}^{n+1}$$

where $I \in \mathbb{R}^{n \times n}$*, then*

$$tr[A^i X] = \|x\|^2 - 2\left(x^i\right)^T x + \left\|x^i\right\|^2.$$

Proof. We have

$$\begin{aligned} A^i X &= \begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix} \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} xx^\intercal - x^i x^\intercal & x - x^i \\ -(x^i)^\intercal x x^\intercal + \|x^i\|^2 x^\intercal & -(x^i)^\intercal x + \|x^i\|^2 \end{bmatrix} \end{aligned}$$

so that

$$\begin{aligned} \mathrm{tr} A^i X &= \mathrm{tr}(xx^\intercal - x^i x^\intercal) - (x^i)^\intercal x + \|x^i\|^2 \\ &= \mathrm{tr}(xx^\intercal) - \mathrm{tr}(x^i x^\intercal) - (x^i)^\intercal x + \|x^i\|^2 \\ &= \mathrm{tr}(x^\intercal x) - \mathrm{tr}(x^\intercal x^i) - (x^i)^\intercal x + \|x^i\|^2 \\ &= \|x\|^2 - 2(x^i)^\intercal x + \|x^i\|^2. \end{aligned}$$

□

For any $x \in \mathbb{R}^n$ and $i \in \{1, \ldots, m\}$, by Proposition 2.5 we have

$$\left\|x - x^i\right\|^2 = \|x\|^2 - 2\left(x^i\right)^T x + \left\|x^i\right\|^2 = \left\langle A^i, \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix} \right\rangle, \qquad (2.10)$$

where

$$A^i := \begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix} \qquad (2.11)$$

and
$$\langle A, Z\rangle = \mathrm{tr}[AZ].$$

By substituting $X = \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix}$, we can reformulate the original problem (2.1) as

$$\begin{aligned} \upsilon_p := \max_{X,\zeta} \quad & \zeta \\ \text{s.t.} \quad & w_i \left\langle A^i, X\right\rangle \geq \zeta,\ i = 1, \ldots, m \\ & (X_{11}, \ldots, X_{nn}, 1)^T \in \mathcal{K} \\ & \mathcal{X}_{n+1,n+1} = 1 \\ & X \succeq 0 \\ & \mathrm{rank} X = 1, \end{aligned} \tag{2.12}$$

where "s.t." abbreviates for "subject to", $\mathcal{K}$ is given as in either (2.7) or (2.8), and A^i is given in (2.11).

Proposition 2.6. *Let $n \geq 2$. The rank function rank : $\mathbb{R}^{n\times n} \to [0, n]$: $X \mapsto \mathrm{rank}X$ is neither convex nor Lipschitz.*

Proof. It suffices to consider the case when $n = 2$. Consider matrices

$$X_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad X_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

We have

$$\mathrm{rank}\left(\frac{1}{2}X_1 + \frac{1}{2}X_2\right) = 2 > 1 = \frac{1}{2}\mathrm{rank}X_1 + \frac{1}{2}\mathrm{rank}X_2,$$

which implies that $X \mapsto \mathrm{rank}X$ is not convex.

Now let $\epsilon > 0$ and
$$Y = \begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix}.$$

We have
$$|\mathrm{rank}X_1 - \mathrm{rank}Y| = |1 - 2| = 1, \quad \|X_1 - Y\| = \epsilon$$

which gives
$$\frac{|\mathrm{rank}X_1 - \mathrm{rank}Y|}{\|X_1 - Y\|} = \frac{1}{\epsilon} \to +\infty$$

when $\epsilon \downarrow 0$. Therefore, rank is not Lipschitz at X_1. ☐

Since the rank function is neither convex nor Lipschitz, we drop the rank-1 constraint. This yields a convex programming relaxation of (2.1):

$$\begin{aligned}
\upsilon_{cp} \;:=\; \max_{X,\zeta}\;& \zeta \\
\text{s.t.}\;\;& w_i \left\langle A^i, X \right\rangle \geq \zeta,\; i = 1,\dots,m \\
& (X_{11},\dots,X_{nn},1)^T \in \mathcal{K} \\
& X_{n+1,n+1} = 1 \\
& X \succeq 0.
\end{aligned} \tag{2.13}$$

As (2.13), a convex relaxation of (2.12), is obtained by relaxing some of the constraints (2.12), all possible solutions of (2.12) are feasible for (2.13). Hence,

Proposition 2.7.

$$\upsilon_{cp} \geq \upsilon_p.$$

Our SDP and SOCP relaxations are formulated in next two subsections.

2.2.1 SDP Relaxation

We will make the mild assumption that (2.13) has an optimal solution. This holds whenever $\mathcal{X}$ is compact. If $\mathcal{X}$ is convex, then we can strengthen this relaxation by adding the constraints $(X_{1,n+1},\dots,X_{n,n+1})^T \in \mathcal{X}$. Since $\upsilon_p > 0$, $\zeta \neq 0$ at an optimal solution of (2.13). By making the substitution $Z = X/\zeta$, we can reformulate (2.13) more compactly as

$$\boxed{\begin{aligned}
\frac{1}{\upsilon_{cp}} \;=\; \min_{Z}\;& Z_{n+1,n+1} \\
\text{s.t.}\;\;& w_i \left\langle A^i, Z \right\rangle \geq 1,\; i = 1,\dots,m \\
& (Z_{11},\dots,Z_{nn},Z_{n+1,n+1})^T \in \mathcal{K} \\
& Z \succeq 0.
\end{aligned}} \tag{2.14}$$

Note that when $\mathcal{K}$ is a polyhedral set having a tractable algebraic representation, for example (2.7), then (2.14) reduces to a semidefinite program (SDP), so we will refer to this as the semidefinite-programming relaxation of (2.1).

Lemma 2.8. *Let $Z^* \in \mathcal{S}^{(n+1)\times(n+1)}$ be an optimal solution of* (2.14)*. Then* $Z^*_{n+1,n+1} > 0$

Proof. This follows from

$$\frac{1}{Z^*_{n+1,n+1}} = \upsilon_{cp} \geq \upsilon_p > 0.$$

□

Theorem 2.9. *When $\mathcal{K} = \left\{ y \in \mathbb{R}^{n+1} \mid y_j \leq y_{n+1}, j = 1, \ldots, n \right\}$, we have*

1. *X solves the convex relaxation problem (2.13) $\Leftrightarrow \frac{X}{\upsilon_{cp}}$ solves (2.14).*
2. *Z solves (2.14) $\Leftrightarrow X = Z\upsilon_{cp} = Z\frac{1}{Z_{n+1,n+1}}$ solves (2.13).*
3. *If Z^* solves* (2.14)*, then $\upsilon_{cp} = \frac{1}{Z^*_{n+1,n+1}}$.*

2.2.2 SOCP Relaxation

Lastly, we consider a further relaxation that replaces the semidefinite-cone constraint, $Z \succeq 0$, of (2.14) with the following constraints

$$\begin{bmatrix} Z_{jj} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succeq 0, \quad j = 1, \ldots, n. \tag{2.15}$$

Lemma 2.10. *The constraints of equation (2.15),*

$$\begin{bmatrix} Z_{j,j} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succeq 0 \qquad i = 1, \ldots, n \tag{2.16}$$

are equivalent to second order cone constraints.

Proof. For each $j = 1, \ldots, n$, let Z^j denote the matrix

$$Z^j = \begin{bmatrix} Z_{j,j} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix}. \tag{2.17}$$

Then by Proposition 1.25 we know that the constraints $Z^j \succeq 0$ are equivalent to the constraints $\det(Z^j) \geq 0$ and $\mathrm{trace}(Z^j) \geq 0$ for $j = 1, \ldots, n$. Now,

$$\begin{aligned} \det(Z^j) \geq 0 \quad &\Leftrightarrow \quad Z_{j,j}Z_{n+1,n+1} - (Z_{j,n+1})^2 \geq 0 \\ &\Leftrightarrow \quad Z_{j,j}Z_{n+1,n+1} \geq (Z_{j,n+1})^2 \end{aligned} \tag{2.18}$$

and

$$\mathrm{trace}(Z^j) \geq 0 \quad \Leftrightarrow \quad Z_{j,j} + Z_{n+1,n+1} \geq 0. \tag{2.19}$$

We will show that (2.18) and (2.19) are equivalent to second order cone constraints, namely

$$\left\|\begin{bmatrix} 2Z_{j,n+1} \\ Z_{j,j} - Z_{n+1,n+1} \end{bmatrix}\right\| \leq Z_{j,j} + Z_{n+1,n+1}, \tag{2.20}$$

$$0 \leq Z_{j,j} + Z_{n+1,n+1}. \tag{2.21}$$

To see that (2.18) and (2.20) are equivalent, consider the following:

$$\begin{aligned}
&\left\|\begin{bmatrix} 2Z_{j,n+1} \\ Z_{j,j} - Z_{n+1,n+1} \end{bmatrix}\right\| \leq Z_{j,j} + Z_{n+1,n+1} \\
\Leftrightarrow &\left\|\begin{bmatrix} 2Z_{j,n+1} \\ Z_{j,j} - Z_{n+1,n+1} \end{bmatrix}\right\|^2 \leq (Z_{j,j} + Z_{n+1,n+1})^2 \\
\Leftrightarrow &\, 4(Z_{j,n+1})^2 + (Z_{j,j})^2 - 2Z_{j,j}Z_{n+1,n+1} + (Z_{n+1,n+1})^2 \\
&\qquad\qquad \leq (Z_{j,j})^2 + 2Z_{j,j}Z_{n+1,n+1} + (Z_{n+1,n+1})^2 \\
\Leftrightarrow &\, 4(Z_{j,n+1})^2 \leq 4Z_{j,j}Z_{n+1,n+1} \\
\Leftrightarrow &\, (Z_{j,n+1})^2 \leq Z_{j,j}Z_{n+1,n+1}
\end{aligned} \tag{2.22}$$

Equation (2.21) is exactly equation (2.19), which is already a second order cone constraint. That is, we have

$$\|Ax + b\| \leq Z_{j,j} + Z_{n+1,n+1} \tag{2.23}$$

with $A = 0$ and $b = 0$. □

Thus, substituting (2.15) for $Z \succeq 0$ in (2.14) yields the second-order cone programming (SOCP) relaxation of (2.1):

$$\boxed{\begin{aligned}
\frac{1}{\upsilon_{socp}} &= \min_{Z} \; Z_{n+1,n+1} \\
&\quad \text{s.t.} \quad w_i \left\langle A^i, Z \right\rangle \geq 1, \; i = 1, \ldots, m \\
&\qquad\qquad (Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K} \\
&\qquad\qquad \begin{bmatrix} Z_{jj} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succeq 0, \quad j = 1, \ldots, n.
\end{aligned}} \tag{2.24}$$

Theorem 2.11. *One always has*

$$\upsilon_p \leq \upsilon_{cp} \leq \upsilon_{socp}.$$

Corollary 2.12. *Let $Z^* \in \mathcal{S}^{n+1}$ be an optimal solution of* (2.24)*. Then $Z^*_{n+1,n+1} > 0$.*

Proof. This follows from

$$\frac{1}{Z^*_{n+1,n+1}} = \upsilon_{socp} \geq \upsilon_{cp} \geq \upsilon_p > 0.$$

□

We have $\upsilon_{socp} = \frac{1}{z^*_{n+1,n+1}}$ if Z^* solves (2.24).

2.2.3 More Properties on the Constraints of Convex Relaxations

(1) Constraint $(Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K}$.
In either (2.14) or (2.24), what does the constraint

$$(Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K} \tag{2.25}$$

look like? We explicitly consider two cases:
Box case: When

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} |\ y_j \leq y_{n+1}, j = 1, \ldots, n\}$$

(2.25) transpires to

$$Z_{jj} \leq Z_{n+1,n+1} \quad \forall j = 1, \ldots, n. \tag{2.26}$$

Ball case: When

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} |\ \sum_{j=1}^{n} y_j \leq y_{n+1}\}$$

(2.25) transpires to

$$\sum_{j=1}^{n} Z_{jj} \leq Z_{n+1,n+1}. \tag{2.27}$$

In both cases, (2.25) consists of linear inequalities.

(2) Constraint $w_i \left\langle A^i, Z \right\rangle \geq 1$. In both (2.14) and (2.24), what does the constraint

$$w_i\langle A^i, Z\rangle \geq 1 \tag{2.28}$$

look like?

We can exploit the structure of A^i! For a matrix $Z \in \mathcal{S}^{n+1}$, let

$$z = (Z_{1,n+1}, \ldots, Z_{n,n+1})^T$$

and

$$Z^* = \begin{bmatrix} Z_{11} & & \\ ** & \ddots & ** \\ & & Z_{nn} \end{bmatrix}$$

so that we can write

$$Z = \begin{bmatrix} Z^* & z \\ (z)^T & Z_{n+1,n+1} \end{bmatrix}. \tag{2.29}$$

into a block form.

Now consider the trace:

$$\begin{aligned}
\langle A^i, Z\rangle &= \mathrm{Tr}\left(\begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix} Z\right) \\
&= \mathrm{Tr}\left(\begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix}\begin{bmatrix} Z^* & z \\ (z)^T & Z_{n+1,n+1} \end{bmatrix}\right) \\
&= \mathrm{Tr}\left(\begin{bmatrix} Z^* - x^i z^T & z - x^i Z_{n+1,n+1} \\ -\left(x^i\right)^T Z^* + \left\|x^i\right\|^2 z^T & -(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1} \end{bmatrix}\right) \\
&= \mathrm{Tr}\left(Z^* - x^i z^T\right) + \mathrm{Tr}\left(-(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1}\right) \\
&= \left(\sum_{j=1}^{n} Z_{jj} - (x^i)^T z\right) + \left(-(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1}\right) \\
&= \sum_{j=1}^{n} Z_{jj} - 2(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1}.
\end{aligned} \tag{2.30}$$

Hence (2.28) is equivalent to

$$w_i\left(\sum_{j=1}^{n} Z_{jj} - 2(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1}\right) \geq 1. \tag{2.31}$$

(3) The feasible region of (2.24) is strictly larger than the one in (2.14). A special case suffices. To this end, let $n = 2$, we consider the matrix

$$Z = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 1 & 0 \\ 1 & 0 & 2 \end{pmatrix}$$

We have

$$\begin{pmatrix} Z_{11} & Z_{13} \\ Z_{31} & Z_{3,3} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \in \mathcal{S}^2_+$$

and

$$\begin{pmatrix} Z_{22} & Z_{23} \\ Z_{32} & Z_{3,3} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} \in \mathcal{S}^2_+,$$

but $Z \notin \mathcal{S}^3_+$ because its principle submatrix

$$\begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \notin \mathcal{S}^2_+.$$

This shows that algthough

$$Z \succcurlyeq 0 \quad \Rightarrow \begin{bmatrix} Z_{jj} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succcurlyeq 0 \ \forall \ j = 1, \dots, n$$

always holds, the converse implication fails.

The above discussions allow us to conclude:

Theorem 2.13 (box case)**.** *Let*

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} | \ y_j \le y_{n+1}, j = 1, \dots, n\}.$$

Then

1. *The SDP relaxation* (2.14) *has an explicit form:*

$$\boxed{\begin{array}{rcl} \dfrac{1}{\upsilon_{cp}} & = & \min\limits_{Z} \ Z_{n+1,n+1} \\ & & s.t. \quad w_i \left(\sum_{j=1}^n Z_{jj} - 2(x^i)^T z + \left\| x^i \right\|^2 Z_{n+1,n+1} \right) \ge 1, \ i = 1, \dots, m \\ & & \qquad\quad Z_{jj} \le Z_{n+1,n+1} \quad j = 1, \dots, n \\ & & \qquad\quad Z \succeq 0, \end{array}} \tag{2.32}$$

where $z = (Z_{1,n+1}, \dots, Z_{n,n+1})^T$.

2. The SOCP relaxation (2.24) *has an explicit form:*

$$
\boxed{\begin{aligned}
\frac{1}{\upsilon_{socp}} = \min_{Z} \;& Z_{n+1,n+1} \\
s.t. \;& w_i\left(\textstyle\sum_{j=1}^n Z_{jj} - 2(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1} \right) \geq 1,\; i = 1,\ldots,m \\
& Z_{jj} \leq Z_{n+1,n+1} \quad j = 1,\ldots,n \\
& \begin{bmatrix} Z_{jj} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succeq 0, \quad j = 1,\ldots,n,
\end{aligned}} \tag{2.33}
$$

where $z = (Z_{1,n+1},\ldots,Z_{n,n+1})^T$.

Theorem 2.14 (ball case)**.** *Let*

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} |\; \sum_{j=1}^n y_j \leq y_{n+1}\}.$$

Then

1. The SDP relaxation (2.14) *has an explicit form:*

$$
\boxed{\begin{aligned}
\frac{1}{\upsilon_{cp}} = \min_{Z} \;& Z_{n+1,n+1} \\
s.t. \;& w_i\left(\textstyle\sum_{j=1}^n Z_{jj} - 2(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1} \right) \geq 1,\; i = 1,\ldots,m \\
& \textstyle\sum_{j=1}^n Z_{jj} \leq Z_{n+1,n+1} \\
& Z \succeq 0,
\end{aligned}} \tag{2.34}
$$

where $z = (Z_{1,n+1},\ldots,Z_{n,n+1})^T$.

2. The SOCP relaxation (2.24) *has an explicit form:*

$$
\boxed{\begin{aligned}
\frac{1}{\upsilon_{socp}} = \min_{Z} \;& Z_{n+1,n+1} \\
s.t. \;& w_i\left(\textstyle\sum_{j=1}^n Z_{jj} - 2(x^i)^T z + \left\|x^i\right\|^2 Z_{n+1,n+1} \right) \geq 1,\; i = 1,\ldots,m \\
& \textstyle\sum_{j=1}^n Z_{jj} \leq Z_{n+1,n+1} \\
& \begin{bmatrix} Z_{jj} & Z_{j,n+1} \\ Z_{j,n+1} & Z_{n+1,n+1} \end{bmatrix} \succeq 0, \quad j = 1,\ldots,n,
\end{aligned}} \tag{2.35}
$$

where $z = (Z_{1,n+1},\ldots,Z_{n,n+1})^T$.

Remark 2.15. Note that in both box case and ball case, the SOCP relaxation only relies on $(Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})$ and $z = (Z_{1,n+1}, \ldots, Z_{n,n+1})^T$.

Remark 2.16. In both box case and ball case, the feasible regions are *unbounded*. Indeed, if Z is feasible, then kZ is also feasible for every $k \geq 1$. Amazingly, in next section we show that both (2.14) and (2.24) do have optimal solutions!

2.2.4 Why Does the SDP or SOCP Relaxation Have an Optimal Solution?

For a matrix $Z \in \mathbb{R}^{n \times n}$ we define its norm by

$$\|Z\|_2 = \max\{\|Zu\| \; \|u\| \leq 1\}.$$

Fact 2.17. *For every $Z \in \mathbb{R}^{n \times n}$, we have*

$$\|Z\|_2 = \lambda_{max}(Z^\mathsf{T} Z).$$

Theorem 2.18 (box case)**.** *Let*

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} | \; y_j \leq y_{n+1}, j = 1, \ldots, n\}.$$

Then the following hold:

1. *The SDP relaxation* (2.14) *has $\upsilon_{cp} < +\infty$ and at least one optimal solution.*

2. *The SOCP relaxation* (2.24) *has $\upsilon_{socp} < +\infty$ and at least one optimal solution.*

Proof. Part 1. First we show that $\upsilon_{cp} < \infty$. Assume to the contrary that $\upsilon_{cp} = \infty$. Then there exists a sequence $Z^{(k)}_{n+1,n+1} \downarrow 0$ when $k \to \infty$. Since by (2.26),

$$0 \leq Z^{(k)}_{jj} \leq Z^{(k)}_{n+1,n+1} \quad \forall \; j = 1, \ldots, n$$

we have $\lim_{k \to \infty} Z^{(k)}_{jj} = 0$. Now the matrix $Z^{(k)} \succeq 0$ implies

$$\begin{bmatrix} Z^{(k)}_{jj} & Z^{(k)}_{j,n+1} \\ Z^{(k)}_{j,n+1} & Z^{(k)}_{n+1,n+1} \end{bmatrix} \succeq 0 \qquad j = 1, \ldots, n \tag{2.36}$$

so that

$$Z^{(k)}_{n+1,n+1} Z^{(k)}_{jj} \geq (Z^{(k)}_{j,n+1})^2 \geq 0$$

from which $\lim_{k\to\infty} Z^{(k)}_{j,n+1} = 0$. By (2.31), we obtain that

$$w_i\left(\sum_{j=1}^{n} Z^{(k)}_{jj} - 2(x^i)^T z^{(k)} + \left\|x^i\right\|^2 Z^{(k)}_{n+1,n+1}\right) \geq 1 \tag{2.37}$$

where $(z^{(k)})^\intercal = (Z^{(k)}_{1,n+1}, \ldots, Z^{(k)}_{n,n+1})$. When $k \to \infty$, (2.37) gives $0 \geq 1$, a contradiction.

Next we show that υ_{cp} is achieved. Assume that there exists a feasible sequence $(Z^{(k)})_{k=1}^{\infty}$ such that $Z^{(k)}_{n+1,n+1} \downarrow 1/\upsilon_{cp}$ when $k \to \infty$. Since $0 \leq Z^{(k)}_{jj} \leq Z^{(k)}_{n+1,n+1}$ for $j = 1, \ldots, n$, the sequence $(Z^{(k)}_{jj})_{k=1}^{\infty}$ is bounded for $j = 1, \ldots, n$. This shows that $(\mathrm{tr} Z^{(k)})_{k=1}^{\infty}$ is bounded. As the largest eigenvalue

$$0 \leq \lambda_{\max}(Z^{(k)}) \leq \mathrm{tr} Z^{(k)}$$

we see that $(\lambda_{\max}(Z^{(k)}))_{k=1}^{\infty}$ is bounded. Note that

$$\|Z^{(k)}\|_2 = \lambda_{\max}(Z^{(k)})$$

because $Z^{(k)} \in \mathcal{S}^{n+1}_{+}$. Since all norms are equivalent in finite dimensional space $\mathcal{S}^{(n+1)\times(n+1)}$, this means that $(Z^{(k)})_{k=1}^{\infty}$ is bounded. Therefore, there exists a subsequence of $(Z^{(k)})_{k=1}^{\infty}$ converges to a $Z^* \in \mathcal{S}^{n+1}_{+}$. In particular, Z^* is feasible because each $Z^{(k)}$ is feasible. Without relabeling, let us assume that $Z^{(k)} \to Z^*$. Then $Z^{(k)}_{n+1,n+1} \to Z^*_{n+1,n+1}$ and $Z^*_{n+1,n+1} = 1/\upsilon_{cp}$. Hence Z^* is an optimal solution.

Part 2. Apply the similar arguments as in Part 1. □

Theorem 2.19 (ball case)**.** *Let*

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} | \sum_{j=1}^{n} y_j \leq y_{n+1}\}.$$

Then the following hold:

1. *The SDP relaxation* (2.14) *has* $\upsilon_{cp} < +\infty$ *and at least one optimal solution.*

2. *The SOCP relaxation* (2.24) *has* $\upsilon_{socp} < +\infty$ *and at least one optimal solution.*

Proof. The proof is similar to that of Theorem 2.18. Instead of using (2.26), we use (2.27). □

Chapter 3

A Necessary Condition for Optimal Solutions

In this Chapter we prove general results about the objective functions of the maximin dispersion problem, as well as deriving a necessary condition for the optimal solutions of (2.1).

3.1 Properties of the Maximin Objective Functions

In this section, we will apply the results of Sections 1.4, 1.5 and 1.5.1 to the objective functions of the maximin dispersion problem, (2.3).

First, we show that $f(x)$ as defined in (2.1) is locally Lipschitz at any x_0 in $\mathbb{R}^n$.

Theorem 3.1. *The function f_i as defined in (2.3) is locally Lipschitz at any $x_0 \in \mathbb{R}^n$.*

Proof. Fix $x_0 \in \mathbb{R}^n$ and choose $x, y \in \mathbb{R}^n$ such that $\|x - x_0\| \leq \delta$ and $\|y - x_0\| \leq \delta$ for some $\delta > 0$. We have

$$\begin{aligned}
|f_i(x) - f_i(y)| &= \left| w_i \left\| x - x^i \right\|^2 - w_i \left\| y - x^i \right\|^2 \right| \qquad (3.1) \\
&= w_i \left| \left(\left\| x - x^i \right\| - \left\| y - x^i \right\| \right) \left(\left\| x - x^i \right\| + \left\| y - x^i \right\| \right) \right| \\
&= w_i \left| \left(\left\| x - x^i \right\| - \left\| y - x^i \right\| \right) \right| \left(\left\| x - x^i \right\| + \left\| y - x^i \right\| \right) \\
&\leq w_i \left(\left\| (x - x^i) - (y - x^i) \right\| \right) \left(\left\| x - x^i \right\| + \left\| y - x^i \right\| \right) \\
&= w_i \left(\|x - y\| \right) \left(\left\| x - x^i \right\| + \left\| y - x^i \right\| \right).
\end{aligned}$$

Now,

$$\begin{aligned}\left\|x - x^i\right\| &\leq \|x\| + \left\|x^i\right\| \\ &= \|x - x_0 + x_0\| + \left\|x^i\right\| \\ &\leq \|x - x_0\| + \|x_0\| + \left\|x^i\right\| \\ &\leq \delta + \|x_0\| + \left\|x^i\right\| \\ &= L_1 < \infty. \end{aligned} \tag{3.2}$$

Similarly, there exists $L_2 < \infty$ such that $\left\|y - x^i\right\| \leq L_2$. Then

$$\left\|x - x^i\right\| + \left\|y - x^i\right\| \leq L_1 + L_2 \tag{3.3}$$

so that by setting $M = w_i(L_1 + L_2)$ we have

$$\begin{aligned}|f_i(x) - f_i(y)| &\leq w_i \|x - y\| (L_1 + L_2) \\ &= M \|x - y\| \end{aligned} \tag{3.4}$$

whenever $\|x - x_0\| \leq \delta$ and $\|y - x_0\| \leq \delta$, and hence f_i is locally Lipschitz at x_0. □

Corollary 3.2. *The function*

$$f(x) = \min_{i=1,\ldots,m} f_i(x) = \min_{i=1,\ldots,m} w_i \left\|x - x^i\right\|^2 \tag{3.5}$$

is locally Lipschitz at any $x_0 \in \mathbb{R}^n$.

This follows directly from Theorem 3.1 and Fact 1.61.

Lastly, we characterize the gradient of $f_i(x)$ as defined in (2.3), and give a theorem for the existence of solutions.

Fact 3.3. *For each $i = 1, \ldots, m$, and every $x \in \mathcal{X}$ the function f_i is differentiable, with derivative*

$$\nabla f_i(x) = 2w_i \left(x - x^i\right). \tag{3.6}$$

Proof. This follows from that

$$f_i(x) = w_i\langle x - x^i, x - x^i\rangle = w_i\big(\langle x, x\rangle + 2\langle x^i, x\rangle + \|x^i\|^2\big).$$

□

Theorem 3.4. *Assume that $\mathcal{X}$ is a compact subset of $\mathbb{R}^n$. Then*

$$\mathrm{argmax}_{x\in\mathcal{X}} f \neq \emptyset.$$

Proof. This follows directly from Theorem 1.9. □

Remark 3.5. If $\mathcal{X}$ is not compact, then it may happen that $\mathrm{argmax} f = \emptyset$. For example, let $\mathcal{X} = \mathbb{R}^n$. Then

$$\sup_{x\in\mathbb{R}^n} f = \infty$$

so that $\mathrm{argmax}_{x\in\mathbb{R}^n} f = \emptyset$.

3.2 Characterization of Optimal Solutions

In this section we will provide a theoretical result that gives a necessary condition for $\bar{x}$ to be an optimal solution of the equally weighted maximin dispersion problem, which is (2.1) with $w_i = \cdots = w_m = w$.

To begin, first rewrite the original problem (2.1) as

$$\min_{x\in\mathcal{X}} \left(\max_{1\le i\le m} -f_i \right) \tag{3.7}$$

and note that this is equivalent to

$$\min_{x\in\mathcal{X}} \left(\max_{1\le i\le m} -f_i + \iota_{\mathcal{X}} \right). \tag{3.8}$$

Theorem 3.6. *Let $\bar{x}$ belong to the interior of $\mathcal{X}$, and let*

$$f(x) = \min_{i=1,\ldots,m} w \left\| x - x^i \right\|^2. \tag{3.9}$$

We have that if $\bar{x}$ is an optimal solution of

$$max_{x\in\mathcal{X}} f(x) \tag{3.10}$$

then

$$\bar{x} \in conv\{x^i \,|i \in I(\bar{x})\} \tag{3.11}$$

where

$$I(\bar{x}) = \left\{ i \,\middle|\, \left\|\bar{x} - x^i\right\|^2 = \min_{i=1,\ldots,m} \left\|\bar{x} - x^i\right\|^2 \right\}. \tag{3.12}$$

Proof. Since $\bar{x}$ is a critical point of f if and only if $\bar{x}$ is a critical point of $-f$, we begin by converting our min-function $f(x)$ into a max-function $-f(x)$ as follows:

$$- f(x) = - \min_{i=1,\ldots,m} w \left\| x - x^i \right\|^2 \tag{3.13}$$

$$-f(x) = \max_{i=1,\ldots,m} \left(-w \left\| x - x^i \right\|^2 \right) \tag{3.14}$$

$$-f(x) = \max_{i=1,\ldots,m} g_i(x) \tag{3.15}$$

where $g_i(x) = -w \left\| x - x^i \right\|^2 = -f_i(x)$.

Now, by Theorem 3.1, we know that $f_i(x)$ is locally Lipschitz at any $x \in \mathbb{R}^n$, and so applying Fact 1.58 we know that $g_i(x)$ is locally Lipschitz at any $x \in \mathbb{R}^n$. Furthermore, by applying Fact 1.59 we know that the max of a family of locally Lipschitz functions is locally Lipschitz, so that $-f(x) = \max_{i=1,\ldots,m} g_i(x)$ is locally Lipschitz at any $x \in \mathbb{R}^n$. Thus, by applying Theorem 1.63 we know that if $\bar{x}$ is a possible critical point of $-f + \iota_{\mathcal{X}}$, then we must have $0 \in \partial^o(-f + \iota_{\mathcal{X}})(\bar{x}) \subseteq \partial^o(-f)(\bar{x}) + \partial^o \iota_{\mathcal{X}}(\bar{x})$. Thus, in order to derive a necessary condition for $\bar{x}$ to be a critical point of $-f(x)$, we need to characterize $\partial^o(-f)(\bar{x})$. Since $-f(x)$ is locally Lipschitz at any $\bar{x} \in \mathbb{R}^n$ and g_i is differentiable for each i, by Theorem 1.64 the Clarke subgradient of $-f$ at $\bar{x}$ is given by

$$\partial(-f(\bar{x})) = \operatorname{conv} \left\{ \nabla g_i(\bar{x}) \mid i \in I(\bar{x}) \right\}. \tag{3.16}$$

Applying Fact 3.3 to the equally weighted case, we have that the derivative of the functions g_i is

$$\nabla g_i(x) = -2w(x - x^i) \tag{3.17}$$

so that

$$\partial^o(-f(\bar{x})) = \operatorname{conv} \left\{ -2w(\bar{x} - x^i) \mid i \in I(\bar{x}) \right\}. \tag{3.18}$$

Thus $0 \in \partial^o(-f)(\bar{x}) + \partial^o \iota_X(\bar{x})$ implies that

$$0 \in \operatorname{conv} \left\{ -2w(\bar{x} - x^i) \mid i \in I(\bar{x}) \right\} + \partial \iota_{\mathcal{X}}(\bar{x}).$$

Dividing both sides by w and using the fact that $\partial^o \iota_{\mathcal{X}}(\bar{x}) = N_{\mathcal{X}}(\bar{x})$ is a cone, we have

$$\Leftrightarrow 0 \in \operatorname{conv} \left\{ -(\bar{x} - x^i) \mid i \in I(\bar{x}) \right\} + \partial^o \iota_{\mathcal{X}}(\bar{x}) \tag{3.19}$$

$$\Leftrightarrow 0 \in -\bar{x} + \operatorname{conv} \left\{ x^i \mid i \in I(\bar{x}) \right\} + \partial^o \iota_{\mathcal{X}}(\bar{x}). \tag{3.20}$$

Since $\bar{x} \in \text{int}\mathcal{X}$, we have $\partial^o \iota_{\mathcal{X}}(\bar{x}) = 0$ by Example 1.48 so that

$$\bar{x} \in \text{conv}\left\{x^i \mid i \in I(\bar{x})\right\}. \tag{3.21}$$

Lastly, since $i \in I(\bar{x})$ if and only if

$$g_i(\bar{x}) = -f(\bar{x}) \tag{3.22}$$
$$\Leftrightarrow -w\left\|\bar{x} - x^i\right\|^2 = -f(\bar{x}) \tag{3.23}$$
$$\Leftrightarrow w\left\|\bar{x} - x^i\right\|^2 = f(\bar{x}) \tag{3.24}$$
$$\Leftrightarrow w\left\|\bar{x} - x^i\right\|^2 = \min_{i=1,\dots,m} w\left\|\bar{x} - x^i\right\|^2 \tag{3.25}$$
$$\Leftrightarrow \left\|\bar{x} - x^i\right\|^2 = \min_{i=1,\dots,m} \left\|\bar{x} - x^i\right\|^2 \tag{3.26}$$

we have that $\bar{x}$ is a critical point of $f(x) = \min_{i=1,\dots,m} w\left\|x - x^i\right\|^2$ if and only if

$$\bar{x} \in \text{conv}\left\{x^i \mid i \in I(\bar{x})\right\} \tag{3.27}$$

where

$$I(\bar{x}) = \left\{ i \;\middle|\; \left\|\bar{x} - x^i\right\|^2 = \min_{i=1,\dots,m} \left\|\bar{x} - x^i\right\|^2 \right\}. \tag{3.28}$$

□

Thus, we have provided a necessary condition for $\bar{x} \in \mathbb{R}^n$ to be a possible maximizer for the equally weighted maximin dispersion problem.

Remark 3.7. Let f be given as in (3.9). Then $\bar{x}$ is an optimal solution of $\max_{x\in\mathcal{X}} f(x)$ only if

$$0 \in \partial^o(-f)(\bar{x}) + N_{\mathcal{X}}(\bar{x}).$$

That is,

$$0 \in \overline{\text{conv}}\left\{-w(\bar{x} - x^i) \mid i \in I(\bar{x})\right\} + N_{\mathcal{X}}(\bar{x}) \tag{3.29}$$
$$\Leftrightarrow \bar{x} \in \text{conv}\left\{x^i \mid i \in I(\bar{x})\right\} + N_{\mathcal{X}}(\bar{x}) \tag{3.30}$$
$$\Leftrightarrow \bar{x} \in \text{conv}\left\{x^i \mid i \in I(\bar{x})\right\} + N_{\mathcal{X}}(\bar{x}). \tag{3.31}$$

See Theorem 1.30.

Chapter 4

NP-Hardness Results

Fact 4.1. *The general weighted maximin distance problem is known to be NP-hard, even in the case where the distances satisfy the triangle inequality [20].*

In this chapter we focus on a particular subset of maximin distance problems, such that $w_1 = \cdots = w_m$, $\mathcal{X} = [-1, 1]^n$ and $x^1, \ldots, x^m$ are in $\mathcal{X}$. We show that even in this reduced case of equal weighting and $\mathcal{X}$ an n-dimensional hypercube, the problem is still NP-hard (see Theorem 4.5). We also present a suggested heuristic for solving (2.1) which involves solving restricted subproblems, and show that the restricted subproblems are still NP-hard.

4.1 The Case That $\mathcal{X}$ is a Box

In this section we will show that the equally weighted maximin optimization problem given by

$$\max_{x \in [-1,1]^n} \min_{i=1,\ldots,m} w \left\| x - x^i \right\|^2 \tag{4.1}$$

is NP-hard.

In order to prove this result, we first provide some preliminary results.

Proposition 4.2. *The following are equivalent to solve:*

1. *y is a solution of the integer linear program*

$$By = b, \quad y \in \{-1, 1\}^q \,, \tag{4.2}$$

with $B \in \mathbb{Z}^{p \times q}$, $b \in \mathbb{Z}^p$.

2. *x is a solution to*

$$Cx \leq 0, \quad x \in \{-1, 1\}^{q+1} \,, \tag{4.3}$$

with $C := \begin{bmatrix} B & -b \\ -B & b \end{bmatrix}$.

Note that when we say that (4.2) and (4.3) are equivalent to solve, we mean it in the following sense:

$y \in \{-1,1\}^q$ solves (4.2) $\Rightarrow$ $(y,-1)$ solves (4.3).

$x = (x_1, \ldots, x_q, x_{q+1})$ solves (4.3) $\Rightarrow$ $\left(\dfrac{x_1}{x_{q+1}}, \cdots, \dfrac{x_q}{x_{q+1}}\right)$ solves (4.2).

Proof. Consider:

$$\begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1q} & -b_1 \\ B_{21} & B_{22} & \cdots & B_{2q} & -b_2 \\ \vdots & & & \vdots & \\ B_{p1} & B_{p2} & \cdots & B_{pq} & -b_p \\ -B_{11} & -B_{12} & \cdots & -B_{1q} & b_1 \\ \vdots & & & \vdots & \\ -B_{p1} & -B_{p2} & \cdots & -B_{pq} & b_p \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{q+1} \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{4.4}$$

$$\Leftrightarrow \begin{cases} B_{11}x_1 + B_{12}x_2 + \cdots + B_{1q}x_q - b_1 x_{q+1} \leq 0 \\ B_{21}x_1 + B_{22}x_2 + \cdots + B_{2q}x_q - b_2 x_{q+1} \leq 0 \\ \vdots \\ B_{p1}x_1 + B_{p2}x_2 + \cdots + B_{pq}x_q - b_p x_{q+1} \leq 0 \\ +B_{11}x_1 + B_{12}x_2 + \cdots + B_{1q}x_q - b_1 x_{q+1} \geq 0 \\ \vdots \\ +B_{p1}x_1 + B_{p2}x_2 + \cdots + B_{pq}x_q - b_p x_{q+1} \geq 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} b_1 x_{q+1} \leq B_{11}x_1 + B_{12}x_2 + \cdots + B_{1q}x_q \leq b_1 x_{q+1} \\ \vdots \\ b_p x_{q+1} \leq B_{p1}x_1 + B_{p2}x_2 + \cdots + B_{pq}x_q \leq b_p x_{q+1} \end{cases}$$

And thus, since $x_i = \pm 1$ for $i = 1, .., q$ we obtain

$$\begin{aligned} & x_{q+1}b \leq Bx \leq x_{q+1}b \\ \Leftrightarrow \quad & Bx = x_{q+1}b \\ \Leftrightarrow \quad & B\left(\frac{x}{x_{q+1}}\right) = b. \end{aligned} \tag{4.5}$$

Since $x_{q+1} = \pm 1$, we have $\frac{x}{x_{q+1}} \in \{-1,1\}^q$, and $\frac{x}{x_{q+1}}$ solves (4.2). $\square$

Proposition 4.3. *The following are equivalent to solve:*

1. *x is a solution to*

$$Cx \leq 0, \quad x \in \{-1, 1\}^{q+1}, \tag{4.6}$$

with $C := \begin{bmatrix} B & -b \\ -B & b \end{bmatrix} \in Z^{2p \times (q+1)}$.

2. *x is a solution to*

$$2Cx \leq \rho e, \quad x \in \{-1, 1\}^{q+1}, \tag{4.7}$$

for any $0 < \rho \leq 1$, *where* $e = (1, \ldots, 1)^T$.

In this case, when we say that (4.6) and (4.7) are equivalent to solve, we mean that

$$x \text{ solves } (4.6) \ \Leftrightarrow x \text{ solves } (4.7)\ .$$

Proof. We can rewrite the inequality (4.7) in item 2 as

$$Cx \leq \frac{\rho}{2} e, \quad x \in \{-1, 1\}^{q+1}. \tag{4.8}$$

Now, since $0 < \rho \leq 1$ we have $0 < \frac{\rho}{2} \leq \frac{1}{2}$. Furthermore, as $C \in \mathbb{Z}^{2p \times (q+1)}$ and $x \in \{-1, 1\}^{q+1}$ it is obvious to see that $Cx \in \mathbb{Z}^{2p}$ so that solving $Cx \leq \frac{1}{2} e$ is equivalent to solving $Cx \leq 0$. □

Proposition 4.4. *The integer linear program (1.94) is reducible in polynomial time to the equally weighted maximin dispersion problem*

$$\max_{x \in [-1,1]^{q+1}} \min_{i=1,\ldots,2p} \left\| x - x^i \right\|^2. \tag{4.9}$$

Moreover,

$$\max_{x \in [-1,1]^{q+1}} \min_{i=1,\ldots,2p} \left\| x - x^i \right\|^2 = \max_{x \in \{-1,1\}^{q+1}} \min_{i=1,\ldots,2p} \left\| x - x^i \right\|^2. \tag{4.10}$$

Proof. From Fact 1.76 we know that it is NP-hard to determine the solvability of

$$By = b, \quad y \in \{-1, 1\}^q, \tag{4.11}$$

with $B \in \mathbb{Z}^{p \times q}$, $b \in \mathbb{Z}^p$. By Proposition 4.2 this is equivalent to

$$Cx \leq 0, \quad x \in \{-1, 1\}^{q+1}, \tag{4.12}$$

with $C := \begin{bmatrix} B & -b \\ -B & b \end{bmatrix}$.

By Proposition 4.3, it is equivalent to solve

$$2Cx \le \rho e, \quad x \in \{-1, 1\}^{q+1}, \tag{4.13}$$

for any $0 < \rho \le 1$, where $e = (1, \dots, 1)^T$.

We assume without loss of generality that C has no zero row. Let c^i denote the ith column of C^T. Then $(c^i)^T$ is the ith row of C so that $\|c^i\| \ne 0$. Let

$$x^i := \varrho_i c^i \in \mathbb{R}^{q+1} \quad \text{with} \quad \varrho_i := \frac{\rho}{\|c^i\|^2}, \; i = 1, \dots, 2p. \tag{4.14}$$

Then $\varrho_i \rho = \frac{\rho^2}{\|c^i\|^2} = \|x^i\|^2$, since

$$\|x^i\| = \|\varrho_i c^i\| = \frac{\rho}{\|c^i\|^2} \|c^i\| = \frac{\rho}{\|c^i\|}.$$

Furthermore, we have

$$2Cx = 2\left[\left(c^1\right)^T \; \left(c^2\right)^T \cdots \left(c^{2p}\right)^T\right]^T x = \begin{bmatrix} 2\left(c^1\right)^T x \\ 2\left(c^2\right)^T x \\ \vdots \\ 2\left(c^{2p}\right)^T x \end{bmatrix} \tag{4.15}$$

and thus

$$\begin{aligned} 2Cx \le \rho e \quad & \Leftrightarrow \quad 2\left(c^i\right)^T x \le \rho \;\; \forall i = 1, \dots, 2p \\ & \Leftrightarrow \quad 2\varrho_i \left(c^i\right)^T x \le \varrho_i \rho \;\; \forall i = 1, \dots, 2p \\ & \Leftrightarrow \quad 2\left(x^i\right)^T x \le \|x^i\|^2, \;\; \forall i = 1, \dots, 2p. \end{aligned} \tag{4.16}$$

Then (4.13) is equivalent to

$$2\left(x^i\right)^T x \le \|x^i\|^2, \quad i = 1, \dots, 2p, \quad x \in \{-1, 1\}^{q+1}. \tag{4.17}$$

Now, from $\|x^i - x\|^2 = \|x^i\|^2 - 2\left(x^i\right)^T x + \|x\|^2$, we have

$$\begin{aligned} & \|x^i\|^2 \ge 2\left(x^i\right)^T x \qquad i = 1, \dots, 2p \\ \Leftrightarrow \quad & \|x^i - x\|^2 + 2\left(x^i\right)^T x - \|x\|^2 \ge 2\left(x^i\right)^T x \qquad i = 1, \dots, 2p \\ \Leftrightarrow \quad & \|x^i - x\|^2 \ge \|x\|^2 \qquad i = 1, \dots, 2p \\ \Leftrightarrow \quad & \|x^i - x\|^2 \ge q + 1 \qquad i = 1, \dots, 2p \end{aligned} \tag{4.18}$$

(since $x \in \{-1,1\}^{q+1}$ implies $\|x\|^2 = q+1$). Hence, (4.17) is in turn is equivalent to

$$q+1 \le \left\|x^i - x\right\|^2, \quad i = 1,\dots,2p, \quad x \in \{-1,1\}^{q+1}. \tag{4.19}$$

Thus (1.94) has a solution if and only if the optimal value of

$$\max_{x\in\{-1,1\}^{q+1}} \min_{i=1,\dots,2p} \left\|x^i - x\right\|^2 \tag{4.20}$$

is no less than $q+1$. That is,

$$\begin{aligned}
& \exists\, x \in \{-1,1\}^{q+1} \text{ such that } q+1 \le \left\|x^i - x\right\|^2 \quad \forall i \\
\Leftrightarrow\quad & \exists\, x \in \{-1,1\}^{q+1} \text{ such that } q+1 \le \min_{i=1,\dots,2p} \left\|x^i - x\right\|^2 \\
\Leftrightarrow\quad & q+1 \le \max_{x\in\{-1,1\}^{q+1}} \min_{i=1,\dots,2p} \left\|x^i - x\right\|^2. \qquad (4.21)
\end{aligned}$$

Now, we show that (4.20) is equivalent to

$$\max_{x\in[-1,1]^{q+1}} \min_{i=1,\dots,2p} \left\|x^i - x\right\|^2. \tag{4.22}$$

.

Indeed, since $\left\|x^i\right\| = \frac{\rho}{\|c^i\|}$, we can take

$$\rho = \min\left\{1, \min_i \left\|c^i\right\|/3\right\}, \tag{4.23}$$

so that $\left\|x^i\right\| < \frac{1}{2}$ for all i. To see this, simply note that by (4.23) we have

$$\begin{aligned}
\rho \le \min_i \frac{\left\|c^i\right\|}{3} &\Rightarrow \rho \le \frac{\left\|c^i\right\|}{3}\ \forall\, i \\
&\Rightarrow \frac{\rho}{\|c^i\|} \le \frac{1}{3}\ \forall\, i.
\end{aligned}$$

Now we show that the objective function value of (4.22) is strictly improved by setting $x = \{\pm 1\}^{q+1}$:
For any $x \in [-1,1]^{q+1}$, if $0 \le x_j < 1$ for some $j \in \{1,\dots,q+1\}$, then, for each $i \in \{1,\dots,2p\}$, either
(i) $x^i_j \le x_j$ so that

$$\left|x_j - x^i_j\right| = x_j - x^i_j < 1 - x^i_j = \left|1 - x^i_j\right|, \text{ or} \tag{4.24}$$

(ii) $x_j^i > x_j$ so that

$$\left|x_j - x_j^i\right| = x_j^i - x_j \le x_j^i < \frac{1}{2} < 1 - x_j^i = \left|1 - x_j^i\right|. \tag{4.25}$$

Then the objective function value of (4.22) is strictly improved by setting $x_j = 1$. Similarly, if $0 \ge x_j > -1$ for some $j \in \{1, \ldots, q+1\}$, then the objective function value of (4.22) is strictly improved by setting $x_j = -1$.

Lastly, it is readily seen that the size of $x^1, \ldots, x^{2p}$ (encoded in binary) is polynomial in the size of $[B\ b]$ ([12] p.9-10, p.21). Thus the solvability of (1.94) is reducible in polynomial time to the optimization problem (4.22). □

We now state the main theorem of this Chapter:

Theorem 4.5. *Let $w > 0$. The equally weighted maximin optimization problem given by*

$$\max_{x \in [-1,1]^n} \min_{i=1,\ldots,m} w \left\|x - x^i\right\|^2 \tag{4.26}$$

is NP-hard.

Proof. By Proposition 4.4, we know that the integer linear program (1.94) is reducible in polynomial time to (4.22). Since (1.94) is NP-hard (Fact 1.76), this implies that (4.22) is NP-hard. Lastly, since (4.22) is simply (4.26) with $w_i = 1$ for all i, we thus have that (4.26) is NP-hard. □

4.2 NP-Hardness of Restricted Problems

In [30], the author describes an heuristic approach for solving (2.1) based on partitioning $\mathcal{X}$ into m Voronoi cells

$$\mathcal{X}^i := \left\{x \in \mathcal{X} \,\middle|\, w_i \left\|x - x^i\right\|^2 \le w_j \left\|x - x^j\right\|^2 \ \forall j \ne i\right\} \ \ i = 1, \ldots, m, \tag{4.27}$$

and maximizing $\left\|x - x^i\right\|^2$ over $x \in \mathcal{X}^i$, for $i = 1, \ldots, m$. However, since (2.1) is NP-hard, at least one of these restricted problems (or subproblems) is NP-hard. In fact, we show below that the restricted problem is NP-hard even when $w_1 = \cdots = w_m$ and $\mathcal{X} = \mathbb{R}^n$.

Theorem 4.6. *We have*

1.

$$f(x) = \min_{1 \le i \le m} w_i \left\| x - x^i \right\|^2 = \begin{cases} w_1 \left\| x - x^1 \right\|^2 & \text{if } x \in \mathcal{X}^1 \\ w_2 \left\| x - x^2 \right\|^2 & \text{if } x \in \mathcal{X}^2 \\ \vdots \\ w_m \left\| x - x^m \right\|^2 & \text{if } x \in \mathcal{X}^m \end{cases} \tag{4.28}$$

where

$$\mathcal{X}^i = \left\{ x \in \mathcal{X} \;\middle|\; w_i \left\| x - x^i \right\|^2 \le w_j \left\| x - x^j \right\|^2 \;\; \forall \; j \neq i \right\},$$

for $i = 1, \ldots, m$.

2.

$$\max_{x \in \mathcal{X}} f(x) = \max \left\{ \max_{x \in \mathcal{X}^1} f(x), \ldots, \max_{x \in \mathcal{X}^m} f(x) \right\}.$$

3. If $w_1 = \cdots = w_m$*, then each* $\mathcal{X}^i$ *is a convex set, since*

$$\begin{aligned} &\left\{ x \in \mathcal{X} \;\middle|\; w_i \left\| x - x^i \right\|^2 \le w_j \left\| x - x^j \right\|^2, \; j \neq i \right\} \\ &\quad = \left\{ x \in \mathcal{X} \;\middle|\; \left\langle x^j - x^i, x \right\rangle \le \left\| x^j \right\|^2 - \left\| x^i \right\|^2, j \neq i \right\}. \end{aligned}$$

Note that $\max_{x \in \mathcal{X}^i} f(x)$ is:

$$\begin{aligned} \max_{x \in \mathcal{X}^i} f(x) = \; & \max_{x \in \mathcal{X}^i} \left\| x - x^i \right\|^2 \\ & \text{s.t.} \qquad \left\| x - x^i \right\|^2 \le \left\| x - x^j \right\|^2 \;\; j \neq i. \end{aligned} \tag{4.29}$$

Let $W \in \mathbb{Z}^{q \times q}$ and assume that W is positive semidefinite and symmetric. We have the following Lemma:

Lemma 4.7. *The MaxCut problem on a graph with q nodes*

$$\max_{y \in \{-1,1\}^q} y^T W y, \tag{4.30}$$

is equivalent to the following problem:

$$\max_{y \in [-1,1]^q} y^T \left(W + D \right) y, \tag{4.31}$$

where $D \in \mathbb{R}^{q \times q}$ *is any diagonal matrix chosen so that* $W + D$ *is positive definite.*

To prove Lemma 4.7 we recall Theorem 1.56, which states that if $D \subset \mathbb{R}^n$ is compact and convex and $f : \mathbb{R}^n \to \mathbb{R}$ is convex, then

$$\boxed{\max_{x \in D} f \;=\; \max_{x \in ext(D)} f} \tag{4.32}$$

where $ext(D)$ is the set of extremal points of D.

Proof. Note that the set of extremal points of $[-1,1]^q$ is $\{-1,1\}^q$. Then

$$\begin{aligned}
\max_{y \in [-1,1]^q} y^T (W + D)\, y &= \max_{y \in \{-1,1\}^q} \left\{ y^T W y + y^T D y \right\} \\
&= \max_{y \in \{-1,1\}^q} \left\{ y^T W y + \sum_{j=1}^{q} y_j^2 D_{jj} \right\} \\
&= \max_{y \in \{-1,1\}^q} \left\{ y^T W y + \sum_{j=1}^{q} D_{jj} \right\} \\
&= \max_{y \in \{-1,1\}^q} y^T W y + \sum_{j=1}^{q} D_{jj}.
\end{aligned} \tag{4.33}$$

Since $\sum_{j=1}^{q} D_{jj}$ is constant, it is equivalent to solve

$$\max_{y \in \{-1,1\}^q} y^T W y. \tag{4.34}$$

□

Now, since $W + D$ is nonsingular, positive definite, by Fact 1.26 we can decompose $W + D = L^T L$ where $L \in \mathbb{R}^{q \times q}$ is a nonsingular and upper (or lower) triangular matrix, and make the substitution $x = Ly$, which leads to the following lemma:

Lemma 4.8. *Solving (4.31) is equivalent to solving*

$$\begin{aligned}
\max_{x \in \mathbb{R}^q} \quad & \|x\|^2 \\
s.t. \quad & 2\left(u^i\right)^T x \le 1,\ -2\left(u^i\right)^T x \le 1,\ i = 1, \ldots, q,
\end{aligned} \tag{4.35}$$

where u^i denotes the ith column of $\left(L^{-1}\right)^T/2$.

Proof. This follows from

$$\left(L^{-1}\right)^T = (2u_1, \ldots, 2u_q) \Rightarrow L^{-1} = \begin{bmatrix} 2u_1^T \\ 2u_2^T \\ \vdots \\ 2u_q^T \end{bmatrix} \tag{4.36}$$

so that

$$x = Ly \Rightarrow y = L^{-1}x = \begin{bmatrix} 2u_1^T \\ 2u_2^T \\ \vdots \\ 2u_q^T \end{bmatrix} x \Rightarrow y_i = 2\left(u_i\right)^T x \quad i = 1, \ldots, q \tag{4.37}$$

and

$$|y_i| \leq 1 \ \ \forall i = 1, \ldots, q \ \Leftrightarrow \ \left|2\left(u^i\right)^T x\right| \leq 1 \ \ \forall i = 1, \ldots, q \tag{4.38}$$

Lastly, it is easily seen that

$$\|x\|^2 = \langle Ly, Ly \rangle = y^T \left(L^T L\right) y = y^T \left(W + D\right) y. \tag{4.39}$$

□

Remark 4.9. x solves (4.35) if and only if $y = L^{-1}x$ solves (4.31).

Lemma 4.10. *Problem (4.35) can be written as*

$$\begin{aligned} \max_{x \in \mathbb{R}^q} \quad & \|x\|^2 \\ s.t. \quad & \|x\|^2 \leq \left\|x - x^i\right\|^2, \ i = 1, \ldots, 2q. \end{aligned} \tag{4.40}$$

where u^i denotes the ith column of $\left(L^{-1}\right)^T/2$ and

$$x^i := \rho_i u^i, \quad x^{i+q} := -\rho_i u^i, \quad \text{with} \quad \rho_i := \frac{1}{\|u^i\|^2}, \ i = 1, \ldots, q. \tag{4.41}$$

Proof. By definition of x^i, we have $\left\|x^i\right\| = \left\|\rho_i u^i\right\| = \frac{1}{\|u^i\|^2}\left\|u^i\right\| = \frac{1}{\|u^i\|}$, so that

$$\begin{aligned} & \|x\|^2 \leq \left\|x - x^i\right\|^2 = \|x\|^2 - 2\left\langle x, x^i \right\rangle + \left\|x^i\right\|^2 \\ \Leftrightarrow \quad & 0 \leq -2\left\langle x, x^i \right\rangle + \frac{1}{\|u^i\|^2} = -2\left\langle x, x^i \right\rangle + \rho_i \\ \Leftrightarrow \quad & 2\left\langle x, x^i \right\rangle \leq \rho_i, \quad i = 1, \ldots, 2q \\ \Leftrightarrow \quad & \left|2\left\langle x, \rho_i u^i \right\rangle\right| \leq \rho_i, \quad i = 1, \ldots, q \\ \Leftrightarrow \quad & \left|2\left\langle x, u^i \right\rangle\right| \leq 1. \end{aligned} \tag{4.42}$$

□

Theorem 4.11. *The MaxCut problem on a graph with q nodes is reducible in polynomial time to the problem (4.40),*

$$\begin{array}{ll} \max\limits_{x \in \mathbb{R}^q} & \|x\|^2 \\ s.t. & \|x\|^2 \le \left\|x - x^i\right\|^2, \ i = 1, \ldots, 2q. \end{array} \tag{4.43}$$

Proof. By applying Lemmas 4.7, 4.8 and 4.10 we know that the MaxCut problem is reducible to solving (4.40). Since $W \ge 0$, $W \in \mathbb{Z}^{q \times q}$, and by choosing D appropriately, we can ensure that $x^1, \ldots, x^{2q}$ have rational entries and the size of $x^1, \ldots, x^{2q}$ (encoded in binary) is polynomial in the size of W ([12]). Thus MaxCut is reducible in polynomial time to the restricted problem (4.40). □

Corollary 4.12. *The restricted problems of the heuristic method of [30],*

$$\begin{array}{ll} \max\limits_{x \in \mathbb{R}^q} & \|x\|^2 \\ s.t. & \|x\|^2 \le \left\|x - x^i\right\|^2, \ i = 1, \ldots, 2q. \end{array} \tag{4.44}$$

are NP-hard.

Proof. By Theorem 4.11, we know that MaxCut is reducible in polynomial time to the restricted problem (4.40). Thus, since the MaxCut problem is NP-hard (Fact 1.78), we have that (4.40) must be NP-hard.

□

Corollary 4.13. *One of the restricted problems*

$$\max_{x \in \mathcal{X}^i} f(x)$$

must be NP-hard.

Proof. Apply Corollary 4.12 and Theorem 4.6 part 2. □

Chapter 5

Convex Relaxation Based Algorithms

Given that (2.1) is NP-hard, see Chapter 4, in general it is very important to explore methods for finding approximate solutions to (2.1). In this chapter, we outline an algorithm which can be used to find approximate solutions to (2.1) using our convex SDP (2.14) and SOCP (2.24) relaxations.

5.1 Auxiliary Results

Let $Z^* \in \mathcal{S}^{n+1}$ be an optimal solution of the SDP relaxation (2.14). Then $Z^* \in \mathcal{S}_+^{n+1}$ so that $Z_{jj}^* \geq 0$ for $j = 1, \ldots, n+1$. For $i = 1, \ldots, m$, let $b^i \in \mathbb{R}^n$ be given by

$$b_j^i = \sqrt{Z_{jj}^*} x_j^i, \quad j = 1, \ldots, n \tag{5.1}$$

where x^i, $i = 1, \ldots, m$ are the given set of points from (2.1).

To prove our main result, Theorem 5.4, we require the following tail estimate by Ben-Tal, Nemirovski, and Roos; see [3]. For completeness, we include the proof.

Lemma 5.1. *Let $\zeta \in \{-1, 1\}^n$ be a random vector, componentwise independent, with*

$$\Pr(\zeta_j = 1) = \Pr(\zeta_j = -1) = \frac{1}{2} \quad \forall j = 1, \ldots, n. \tag{5.2}$$

Then for any $\alpha > 0$,

$$\Pr\left(\left(b^i\right)^T \zeta \geq \sqrt{\alpha} \left\|b^i\right\|\right) \leq e^{-\alpha/2}, \quad i = 1, \ldots, m.$$

Proof. (based on proof of Lemma A.3 in [3]):

First, note that ζ is a random variable and b^i is fixed for each i, so $(b^i)^T \zeta$ is a random variable. Since $\alpha > 0$ and $\left\|b^i\right\|$ are fixed, $(b^i)^T \zeta - \sqrt{\alpha} \left\|b^i\right\|$ is

also a random variable.
Now, for any random variable y with density function f, for any $\rho \geq 0$ we have

$$\begin{aligned} E\left(e^{\rho y}\right) &= \int_{-\infty}^{0} e^{\rho y} f(y)dy + \int_{0}^{\infty} e^{\rho y} f(y)dy \\ &\geq 0 + \int_{0}^{\infty} f(y)dy \qquad \text{(since } e^{\rho y} \geq 1 \text{ for } y \geq 0\text{)} \\ &= \Pr\left(y \geq 0\right). \end{aligned} \tag{5.3}$$

Hence, substituting $y = (b^i)^T\zeta - \sqrt{\alpha}\left\|b^i\right\|$, we have:

$$\begin{aligned} \Pr\left((b^i)^T\zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) &= \Pr\left((b^i)^T\zeta - \sqrt{\alpha}\left\|b^i\right\| \geq 0\right) \\ &\leq \mathrm{E}\left(e^{\rho\left((b^i)^T\zeta - \sqrt{\alpha}\|b^i\|\right)}\right) \\ &= \mathrm{E}\left(e^{\rho\left((b^i)^T\zeta\right)}\right) e^{-\rho\sqrt{\alpha}\|b^i\|} \end{aligned} \tag{5.4}$$

and thus

$$\begin{aligned} \mathrm{E}\left(e^{\rho\left((b^i)^T\zeta\right)}\right) &= \prod_{j=1}^{n} \mathrm{E}\left(e^{\rho b^i_j \zeta_j}\right) \\ &= \prod_{j=1}^{n} \frac{1}{2}\left(e^{\rho b^i_j} + e^{-\rho b^i_j}\right) \qquad \text{(from 5.2)} \\ &= \prod_{j=1}^{n} \cosh\left(\rho b^i_j\right) \end{aligned} \tag{5.5}$$

Apply the inequality $\cosh t \leq e^{\frac{1}{2}t^2}$ (which holds by Taylor's series) to obtain

$$\begin{aligned} \mathrm{E}\left(e^{\rho\left((b^i)^T\zeta\right)}\right) &\leq \prod_{j=1}^{n} e^{\frac{1}{2}\rho^2 (b^i_j)^2} = e^{\frac{1}{2}\rho^2\left(\sum_{j=1}^{n}(b^i_j)^2\right)} \\ &= e^{\frac{1}{2}\rho^2\left\|b^i\right\|^2}. \end{aligned}$$

Hence, we have for $\rho \geq 0$

$$\begin{aligned} \Pr\left((b^i)^T\zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) &\leq \mathrm{E}\left(e^{\rho\left((b^i)^T\zeta\right)}\right) e^{-\rho\sqrt{\alpha}\|b^i\|} \\ &\leq \left(e^{\frac{1}{2}\rho^2\left\|b^i\right\|^2}\right) e^{-\rho\sqrt{\alpha}\|b^i\|} \\ &= e^{\frac{1}{2}\rho^2\left\|b^i\right\|^2 - \rho\sqrt{\alpha}\|b^i\|}. \end{aligned} \tag{5.6}$$

The right hand side of this equation is minimized at $\rho = \sqrt{\alpha}/\left\|b^i\right\|$, with

$$e^{\frac{1}{2}(\sqrt{\alpha}/\|b^i\|)^2\|b^i\|^2-(\sqrt{\alpha}/\|b^i\|)\sqrt{\alpha}\|b^i\|} = e^{-\frac{\alpha}{2}} \tag{5.7}$$

Thus, we have

$$\Pr\left(\left(b^i\right)^T\zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) \leq e^{-\frac{\alpha}{2}} \tag{5.8}$$

as required. □

5.2 SDP Relaxation Based Algorithm

In this section, we consider the case of (2.6) and propose an algorithm for constructing an approximate solution of (2.1) based on the SDP relaxation (2.14), with an approximation bound of

$$\frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_{cp}. \tag{5.9}$$

We will see in Section 6.1 that, in the case of $\mathcal{X} = [-1,1]^n$, this yields an approximation bound that is uniformly bounded away from zero as $n \to \infty$. First we give some elementary properties of optimal solutions to the SDP relaxation.

Lemma 5.2 (box case)**.** *Let*

$$\mathcal{K} = \left\{y \in \mathbb{R}^{n+1}\,|y_j \leq y_{n+1},\ j = 1,\ldots,n\right\}. \tag{5.10}$$

If the SDP

$$\begin{array}{rcl}\dfrac{1}{\upsilon_{cp}} & = & \min\limits_{Z}\ Z_{n+1,n+1} \\ & & s.t. \quad w_i\left\langle A^i, Z\right\rangle \geq 1,\ i=1,\ldots,m \\ & & \qquad\ \ (Z_{11},\ldots,Z_{nn},Z_{n+1,n+1})^T \in \mathcal{K} \\ & & \qquad\ \ Z \succeq 0.\end{array}$$

has a solution, then it has a solution $Z^* \in \mathcal{S}^{n+1}$ such that

$$Z^*_{jj} = Z^*_{n+1,n+1} \quad \forall\ j = 1,\ldots,n.$$

In particular, $Z^*_{jj} > 0$ for $j = 1,\ldots,n$.

Proof. The constraint $(Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K}$ gives

$$Z_{jj} \leq Z_{n+1,n+1} \quad \forall\, j = 1, \ldots, n. \tag{5.11}$$

The constraint $w_i \left\langle A^i, Z \right\rangle \geq 1$ gives

$$w_i \left(\sum_{j=1}^{n} Z_{jj} + \left\| x^i \right\|^2 Z_{n+1,n+1} - 2 \left(x^i\right)^T z^* \right) \geq 1. \tag{5.12}$$

If Z is a feasible solution, by (5.11)

$$\mathrm{Diag}(Z_{n+1,n+1} - Z_{11}, \ldots, Z_{n+1,n+1} - Z_{nn}, 0) \succeq 0.$$

Then

$$\tilde{Z} := Z + \mathrm{Diag}(Z_{n+1,n+1} - Z_{11}, \ldots, Z_{n+1,n+1} - Z_{nn}, 0) \succeq 0$$

and also verifies (5.11) and (5.12), so $\tilde{Z}$ is feasible. Moreover,

$$\tilde{Z}_{jj} = Z_{n+1,n+1} \quad \forall\, j = 1, \ldots, n.$$

Hence any optimal solution Z^* to the SDP can be modified so that

$$Z^*_{jj} = Z^*_{n+1,n+1} \quad \forall\, j = 1, \ldots, n.$$

The proof is complete by using Lemma 2.8. □

Lemma 5.3 (ball case)**.** *Let*

$$\mathcal{K} = \{y \in \mathbb{R}^{n+1} | \sum_{j=1}^{n} y_j \leq y_{n+1}\} \tag{5.13}$$

If the SDP

$$\begin{aligned} \frac{1}{\upsilon_{cp}} \;=\; & \min_{Z} \; Z_{n+1,n+1} \\ & s.t. \quad w_i \left\langle A^i, Z \right\rangle \geq 1,\; i = 1, \ldots, m \\ & \qquad\;\; (Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K} \\ & \qquad\;\; Z \succeq 0. \end{aligned}$$

has a solution, then it has a solution $Z^* \in \mathcal{S}^{n+1}$ such that

$$\sum_{j=1}^{n} Z^*_{jj} = Z^*_{n+1,n+1}.$$

In particular, $\sum_{j=1}^{n} Z^*_{jj} > 0$.

Proof. The constraint $(Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K}$ gives

$$\sum_{j=1}^{n} Z_{jj} \leq Z_{n+1,n+1}. \tag{5.14}$$

The constraint $w_i \left\langle A^i, Z \right\rangle \geq 1$ gives

$$w_i \left(\sum_{j=1}^{n} Z_{jj} + \left\| x^i \right\|^2 Z_{n+1,n+1} - 2 \left(x^i \right)^T z^* \right) \geq 1. \tag{5.15}$$

If Z is a feasible solution, by (5.14)

$$\mathrm{Diag}(Z_{n+1,n+1} - \sum_{j=1}^{n} Z_{jj}, 0, \ldots, 0) \succeq 0.$$

Then

$$\tilde{Z} := Z + \mathrm{Diag}(Z_{n+1,n+1} - \sum_{j=1}^{n} Z_{jj}, 0, \ldots, 0) \succeq 0$$

and also verifies (5.14) and (5.15), so $\tilde{Z}$ is feasible. Moreover,

$$\sum_{j=1}^{n} \tilde{Z}_{jj} = Z_{n+1,n+1}.$$

Hence any optimal solution Z^* to the SDP can be modified so that

$$\sum_{j=1}^{n} Z^*_{jj} = Z^*_{n+1,n+1}.$$

The proof is complete by using Lemma 2.8. $\square$

Theorem 5.4. *Let $Z^* \in \mathcal{S}^{n+1}$ be an optimal solution of the SDP-relaxation problem (2.14)*

$$\begin{array}{rcll} \dfrac{1}{\upsilon_{cp}} & = & \min\limits_{Z} & Z_{n+1,n+1} \\ & & s.t. & w_i \left\langle A^i, Z \right\rangle \geq 1, \ i = 1, \ldots, m \\ & & & (Z_{11}, \ldots, Z_{nn}, Z_{n+1,n+1})^T \in \mathcal{K} \\ & & & Z \succeq 0 \end{array}$$

and define

$$\gamma^* := \frac{\max_{j=1,\ldots,n} Z^*_{jj}}{\sum_{j=1}^{n} Z^*_{jj}}.$$

Then there exists a feasible solution $\widetilde{x}$ for the original optimization problem (2.1) such that

1.
$$\tilde{x} = \left(\frac{\sqrt{Z_{11}^*}\zeta_1}{\sqrt{Z_{n+1,n+1}^*}}, \frac{\sqrt{Z_{22}^*}\zeta_2}{\sqrt{Z_{n+1,n+1}^*}}, \cdots, \frac{\sqrt{Z_{nn}^*}\zeta_n}{\sqrt{Z_{n+1,n+1}^*}} \right) \tag{5.16}$$

where $\zeta = (\zeta_1, \ldots, \zeta_n)$ *satisfies* $\zeta \in \{-1,1\}^n$ *and* $(b^i)^T\zeta < \sqrt{\alpha}\,\|b^i\|$, *and* $\alpha = 2\ln(m/\rho)$.

2.
$$f(\tilde{x}) \;=\; \min_{i=1,\ldots,m} w_i \left\|\tilde{x} - x^i\right\|^2 \;\geq\; \frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_{cp}. \tag{5.17}$$

3. $f(\tilde{x}) \geq \frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_p$.

Before proving Theorem 5.4 we provide some Propositions and definitions. Recall that in Section 5.1 we let $Z^* \in \mathcal{S}^{n+1}$ be an optimal solution of the SDP relaxation (2.14), and for $i = 1, \ldots, m$, we let $b^i \in \mathbb{R}^n$ be given by

$$b^i = \left(\sqrt{Z_{11}^*}x_1^i, \ldots, \sqrt{Z_{nn}^*}x_n^i\right). \tag{5.18}$$

Proposition 5.5. *Fix any* $0 < \rho < 1$, *and set* $\alpha = 2\ln(m/\rho)$. *Then*

$$\Pr\left(\left(b^i\right)^T\zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) \leq \frac{\rho}{m}, \quad i = 1, \ldots, m. \tag{5.19}$$

Proof. Indeed, by Lemma 5.1 we have

$$\begin{aligned}
\Pr\left(\left(b^i\right)^T\zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) &\leq e^{-\left(2\ln\frac{m}{\rho}\right)/2} \\
&= e^{-\ln\frac{m}{\rho}} \\
&= e^{\ln\frac{\rho}{m}} \\
&= \frac{\rho}{m}.
\end{aligned} \tag{5.20}$$

□

Proposition 5.6. *Fix any* $0 < \rho < 1$, *and set* $\alpha = 2\ln(m/\rho)$. *We have*

$$\Pr\left(\left(b^i\right)^T\zeta < \sqrt{\alpha}\left\|b^i\right\|,\; i = 1, \ldots, m\right) \geq 1 - \rho > 0, \tag{5.21}$$

so that there exists a $\zeta \in \{-1,1\}^n$ *satisfying*

$$\left(b^i\right)^T\zeta < \sqrt{\alpha}\left\|b^i\right\|, \quad i = 1, \ldots, m. \tag{5.22}$$

Proof. Consider:

$$\begin{aligned}
&\Pr\left(\left(b^i\right)^T \zeta < \sqrt{\alpha}\left\|b^i\right\|,\ i=1,\ldots,m\right) \\
&\quad = 1-\Pr\left(\left(b^i\right)^T \zeta \geq \sqrt{\alpha}\left\|b^i\right\| \quad \text{for some} \quad i \in \{1,\ldots,m\}\right) \\
&\quad \geq 1-\sum_{i=1}^{m} \Pr\left(\left(b^i\right)^T \zeta \geq \sqrt{\alpha}\left\|b^i\right\|\right) \\
&\quad \geq 1-\sum_{i=1}^{m} \frac{\rho}{m} \qquad \text{(Apply Proposition 5.5)} \\
&\quad = 1-\rho > 0. \qquad (5.23)
\end{aligned}$$

□

How do we find the random vector ξ verifying (5.22)**?** Such a ζ can be found by repeated random sampling, and the probability of finding such a ζ after N samples is at least $1-\rho^N$. That is, the probability that we do *not* find such a ζ in each sample is less than or equal to ρ, so the probability of finding such a ζ in N samples is greater than or equal to $1-\rho^N$.

Now, set $\tilde{z} \in \mathbb{R}^n$ and $\tilde{z}_{n+1}$ according to

$$\tilde{z}_j = \sqrt{Z^*_{jj}}\zeta_j,\ j=1,\ldots,n, \quad \tilde{z}_{n+1} = \sqrt{Z^*_{n+1,n+1}}. \qquad (5.24)$$

Then

$$\begin{aligned}
\tilde{z} &= (\sqrt{Z^*_{11}}\zeta_1,\ldots,\sqrt{Z^*_{nn}}\zeta_n), \qquad (5.25)\\
\left(\tilde{z}_1^2,\ldots,\tilde{z}_n^2,\tilde{z}_{n+1}^2\right)^T &= \left(Z^*_{11},\ldots,Z^*_{nn},Z^*_{n+1,n+1}\right)^T \in \mathcal{K}.
\end{aligned}$$

We will show that $\left\|\tilde{z}-x^i\tilde{z}_{n+1}\right\|^2$ is not too small for all i, yielding a non-trivial approximation bound.

Proposition 5.7. *With $\tilde{z}$ given in* (5.24)*, we have*

$$\left\|\tilde{z}-x^i\tilde{z}_{n+1}\right\|^2 = \sum_{j=1}^{n} Z^*_{jj} - 2\left(b^i\right)^T \zeta\sqrt{Z^*_{n+1,n+1}} + \left\|x^i\right\|^2 Z^*_{n+1,n+1} \qquad (5.26)$$

For $i=1,\ldots,m$.

Proof. First,

$$\left\|\tilde{z}-x^i\tilde{z}_{n+1}\right\|^2 = \|\tilde{z}\|^2 - 2\left(x^i\right)^T \tilde{z}\tilde{z}_{n+1} + \left\|x^i\right\|^2 \tilde{z}_{n+1}^2. \qquad (5.27)$$

Then apply (5.1):

$$b_j^i = \sqrt{Z_{jj}^*} x_j^i \;\Rightarrow\; x_j^i = \frac{b_j^i}{\sqrt{Z_{jj}^*}} \tag{5.28}$$

and

$$(x^i)^T \tilde{z} = \sum_{j=1}^n x_j^i \tilde{z}_j = \sum_{j=1}^n \frac{b_j^i}{\sqrt{Z_{jj}^*}} \sqrt{Z_{jj}^*} \zeta_j = (b^i)^T \zeta \tag{5.29}$$

to get the desired result. □

Now, define

$$\gamma^* := \frac{\max_{j=1,\ldots,n} Z_{jj}^*}{\sum_{j=1}^n Z_{jj}^*}. \tag{5.30}$$

Proposition 5.8. *We have*

$$\begin{aligned} & \sum_{j=1}^n Z_{jj}^* - 2\left(b^i\right)^T \zeta \sqrt{Z_{n+1,n+1}^*} + \left\|x^i\right\|^2 Z_{n+1,n+1}^* \qquad (5.31) \\ \geq \; & \sum_{j=1}^n Z_{jj}^* - 2\sqrt{\alpha\gamma^* \sum_{j=1}^n Z_{jj}^*} \left\|x^i\right\| \sqrt{Z_{n+1,n+1}^*} + \left\|x^i\right\|^2 Z_{n+1,n+1}^*. \end{aligned}$$

Proof. First, since ζ satisfies $\left(b^i\right)^T \zeta < \sqrt{\alpha}\left\|b^i\right\|, \;\; i = 1, \ldots, m$, we have that

$$\begin{aligned} & \sum_{j=1}^n Z_{jj}^* - 2\left(b^i\right)^T \zeta \sqrt{Z_{n+1,n+1}^*} + \left\|x^i\right\|^2 Z_{n+1,n+1}^* \\ \geq \; & \sum_{j=1}^n Z_{jj}^* - 2\sqrt{\alpha}\left\|b^i\right\| \sqrt{Z_{n+1,n+1}^*} + \left\|x^i\right\|^2 Z_{n+1,n+1}^*. \qquad (5.32) \end{aligned}$$

Furthermore, we have

$$\begin{aligned} \left\|b^i\right\|^2 &= \sum_{j=1}^n Z_{jj}^* \left(x_j^i\right)^2 \\ &\leq \left(\max_{j=1,\ldots,n} Z_{jj}^*\right) \sum_{j=1}^n (x_j^i)^2 \\ &= \left(\max_{j=1,\ldots,n} Z_{jj}^*\right) \left\|x^i\right\|^2 \\ &= \left(\gamma^* \sum_{j=1}^n Z_{jj}^*\right) \left\|x^i\right\|^2 \qquad (5.33) \end{aligned}$$

by (5.28) and (5.30). Then

$$\begin{aligned} & \sum_{j=1}^{n} Z_{jj}^{*} - 2\sqrt{\alpha}\left\|b^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*} \\ \geq & \sum_{j=1}^{n} Z_{jj}^{*} - 2\sqrt{\alpha\gamma^{*}\sum_{j=1}^{n} Z_{jj}^{*}}\left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*}. \end{aligned} \tag{5.34}$$

Combining equations (5.32) and (5.33) gives the desired result. □

Proposition 5.9. *We have*

$$\begin{aligned} \sum_{j=1}^{n} Z_{jj}^{*} \quad - \quad & 2\sqrt{\alpha\gamma^{*}\sum_{j=1}^{n} Z_{jj}^{*}}\left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*} \\ \geq \quad & \left(1-\sqrt{\alpha\gamma^{*}}\right)\left(\sum_{j=1}^{n} Z_{jj}^{*} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*}\right). \end{aligned} \tag{5.35}$$

Proof. First,

$$\begin{aligned} & \sum_{j=1}^{n} Z_{jj}^{*} - 2\sqrt{\alpha\gamma^{*}\sum_{j=1}^{n} Z_{jj}^{*}}\left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*} \\ = & \left(1-\sqrt{\alpha\gamma^{*}}\right)\left(\sum_{j=1}^{n} Z_{jj}^{*} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*}\right) + \sqrt{\alpha\gamma^{*}}\left(\sqrt{\sum_{j=1}^{n} Z_{jj}^{*}} - \left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}}\right)^{2} \end{aligned} \tag{5.36}$$

which can be seen by simply expanding and grouping terms:

$$\begin{aligned} & \left(1-\sqrt{\alpha\gamma^{*}}\right)\left(\sum_{j=1}^{n} Z_{jj}^{*} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*}\right) + \sqrt{\alpha\gamma^{*}}\left(\sqrt{\sum_{j=1}^{n} Z_{jj}^{*}} - \left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}}\right)^{2} \\ = & \ (1-\sqrt{\alpha\gamma^{*}})\sum_{j=1}^{n} Z_{jj}^{*} + (1-\sqrt{\alpha\gamma^{*}})\left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*} \\ + & \sqrt{\alpha\gamma^{*}}\left(\sum_{j=1}^{n} Z_{jj}^{*} - 2\sqrt{\sum_{j=1}^{n} Z_{jj}^{*}}\left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*}\right) \\ = & \sum_{j=1}^{n} Z_{jj}^{*} + \left\|x^{i}\right\|^{2} Z_{n+1,n+1}^{*} - 2\sqrt{\alpha\gamma^{*}\sum_{j=1}^{n} Z_{jj}^{*}}\left\|x^{i}\right\|\sqrt{Z_{n+1,n+1}^{*}}. \end{aligned} \tag{5.37}$$

Then simply note that

$$\sqrt{\alpha\gamma^*}\left(\sqrt{\sum_{j=1}^{n} Z_{jj}^*} - \left\|x^i\right\| \sqrt{Z_{n+1,n+1}^*}\right)^2 \geq 0. \tag{5.38}$$

and hence

$$\begin{aligned}
& \left(1-\sqrt{\alpha\gamma^*}\right)\left(\sum_{j=1}^{n} Z_{jj}^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^*\right) + \sqrt{\alpha\gamma^*}\left(\sqrt{\sum_{j=1}^{n} Z_{jj}^*} - \left\|x^i\right\| \sqrt{Z_{n+1,n+1}^*}\right)^2 \\
\geq \quad & \left(1-\sqrt{\alpha\gamma^*}\right)\left(\sum_{j=1}^{n} Z_{jj}^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^*\right). \qquad (5.39)
\end{aligned}$$

□

Next, we provide results that will help to establish our lower bound.

Proposition 5.10. *For each* $i = 1, \ldots, m$,

$$\frac{1}{w_i} \leq 2\left(\sum_{j=1}^{n} Z_{jj}^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^*\right). \tag{5.40}$$

Proof. Recall that in (2.14) our first constraint is that $w_i \left\langle A^i, Z\right\rangle \geq 1$ or $\frac{1}{w_i} \leq \left\langle A^i, Z\right\rangle$ for $i = 1, \ldots, m$. Consider the following sequence of inequalities, which we will justify later:

$$\begin{aligned}
\frac{1}{w_i} &\leq \left\langle A^i, Z^*\right\rangle \\
&= \left\langle \begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix}, Z^* \right\rangle \\
&= \sum_{j=1}^{n} Z_{jj}^* - 2\left(x^i\right)^T z^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^* \qquad (5.41) \\
&\leq 2\left(\sum_{j=1}^{n} Z_{jj}^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^*\right). \qquad (5.42)
\end{aligned}$$

For (5.41), let

$$z^* = \left(Z_{1,n+1}^*, \ldots, Z_{n,n+1}^*\right)^T$$

and

$$Z^{**} = \begin{bmatrix} Z^*_{11} & & \\ ** & \ddots & ** \\ & & Z^*_{nn} \end{bmatrix}$$

so that we can write

$$Z^* = \begin{bmatrix} Z^{**} & z^* \\ (z^*)^T & Z^*_{n+1,n+1} \end{bmatrix}. \tag{5.43}$$

Now consider the trace:

$$\begin{aligned}
&Tr\left(\begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix} Z^*\right) \\
&\quad Tr\left(\begin{bmatrix} I & -x^i \\ -\left(x^i\right)^T & \left\|x^i\right\|^2 \end{bmatrix}\begin{bmatrix} Z^{**} & z^* \\ (z^*)^T & Z^*_{n+1,n+1} \end{bmatrix}\right) \\
&\quad = Tr\left(\begin{bmatrix} Z^{**} - x^i(z^*)^T & z^* - x^i Z^*_{n+1,n+1} \\ -\left(x^i\right)^T Z^{**} + \left\|x^i\right\|^2 (z^*)^T & -(x^i)^T z^* + \left\|x^i\right\|^2 Z^*_{n+1,n+1} \end{bmatrix}\right) \\
&\quad = Tr\left(Z^{**} - x^i(z^*)^T\right) + Tr\left(-(x^i)^T z^* + \left\|x^i\right\|^2 Z^*_{n+1,n+1}\right) \\
&\quad = \left(\sum_{j=1}^{n} Z^*_{jj} - (x^i)^T z^*\right) + \left(-(x^i)^T z^* + \left\|x^i\right\|^2 Z^*_{n+1,n+1}\right) \\
&\quad = \sum_{j=1}^{n} Z^*_{jj} - 2(x^i)^T z^* + \left\|x^i\right\|^2 Z^*_{n+1,n+1}. \qquad (5.44)
\end{aligned}$$

For (5.42), we have that $Z^* \succeq 0$ so that

$$\begin{bmatrix} Z^*_{jj} & Z^*_{j,n+1} \\ Z^*_{j,n+1} & Z^*_{n+1,n+1} \end{bmatrix} \succeq 0, \quad j = 1, \ldots, n. \tag{5.45}$$

Then

$$\left(Z^*_{jj}\right)\left(Z^*_{n+1,n+1}\right) - \left(Z^*_{j,n+1}\right)^2 \geq 0 \quad \text{for } j = 1, \ldots, n \tag{5.46}$$

and thus

$$\begin{aligned}
2\left|\left(x^i\right)^T z^*\right| &\leq 2\left\|x^i\right\|\|z^*\| && \text{(Cauchy-Schwarz)} \\
&= 2\left\|x^i\right\|\sqrt{\sum_{j=1}^{n}\left(Z^*_{j,n+1}\right)^2} && (5.47) \\
&\leq 2\left\|x^i\right\|\sqrt{\sum_{j=1}^{n} Z^*_{jj} Z^*_{n+1,n+1}} && \text{(see (5.46))} \\
&= 2\left(\left\|x^i\right\|\sqrt{Z^*_{n+1,n+1}}\right)\left(\sqrt{\sum_{j=1}^{n} Z^*_{jj}}\right) \\
&\leq \left\|x^i\right\|^2 Z^*_{n+1,n+1} + \sum_{j=1}^{n} Z^*_{jj} && (2ab \leq a^2 + b^2).
\end{aligned}$$

Hence,

$$\begin{aligned}
&\sum_{j=1}^{n} Z^*_{jj} - 2\left(x^i\right)^T z^* + \left\|x^i\right\|^2 Z^*_{n+1,n+1} \\
&\qquad \leq \sum_{j=1}^{n} Z^*_{jj} + \left\|x^i\right\|^2 Z^*_{n+1,n+1} + \sum_{j=1}^{n} Z^*_{jj} + \left\|x^i\right\|^2 Z^*_{n+1,n+1} \\
&\qquad = 2\left(\sum_{j=1}^{n} Z^*_{jj} + \left\|x^i\right\|^2 Z^*_{n+1,n+1}\right). \qquad (5.48)
\end{aligned}$$

□

5.3 Proof of Theorem 5.4: SDP Relaxation Based Algorithm

Using the results of the previous section (5.2) we now provide the proof of Theorem 5.4.

Proof. First, by combining Propositions 5.7, 5.8 and 5.9, we have

$$\left\|\tilde{z} - x^i \tilde{z}_{n+1}\right\|^2 \geq \left(1 - \sqrt{\alpha\gamma^*}\right)\left(\sum_{j=1}^{n} Z^*_{jj} + \left\|x^i\right\|^2 Z^*_{n+1,n+1}\right). \qquad (5.49)$$

Combining the equation (5.49) with Proposition 5.10 we have that

$$\begin{aligned}\min_{i=1,\ldots,m} w_i \left\|\tilde{z} - x^i \tilde{z}_{n+1}\right\|^2 &\geq \frac{(1-\sqrt{\alpha\gamma^*})\left(\sum_{j=1}^n Z_{jj}^* + \left\|x^i\right\|^2 Z_{n+1,n+1}^*\right)}{2\left(\sum_{j=1}^n Z_{jj}^* + \|x^i\|^2 Z_{n+1,n+1}^*\right)} \\ &= \frac{1-\sqrt{\alpha\gamma^*}}{2}. \end{aligned} \tag{5.50}$$

Thus $\tilde{x} \in \mathbb{R}^n$ given by

$$\boxed{\tilde{x} := \frac{\tilde{z}}{\tilde{z}_{n+1}},} \tag{5.51}$$

belongs to $\mathcal{X}$ given by (2.6), since, by (5.24)

$$\left(\tilde{x}_1^2, \ldots, \tilde{x}_n^2\right) = \left(\frac{Z_{11}^*}{Z_{n+1,n+1}^*}, \ldots, \frac{Z_{nn}^*}{Z_{n+1,n+1}^*}\right), \tag{5.52}$$

and $\left(Z_{11}^*, \ldots, Z_{nn}^*, Z_{n+1,n+1}^*\right) \in \mathcal{K}$ by feasibility. Since $\mathcal{K}$ is a convex cone,

$$\frac{1}{Z_{n+1,n+1}^*}\left(Z_{11}^*, \ldots, Z_{nn}^*, Z_{n+1,n+1}^*\right) \in \mathcal{K}, \tag{5.53}$$

so that $\left(\tilde{x}_1^2, \ldots, \tilde{x}_n^2, 1\right) \in \mathcal{K}$ and thus $\tilde{x} \in \mathcal{X}$. This establishes part 1.

Furthermore, $\tilde{x}$ satisfies

$$f(\tilde{x}) = \min_{i=1,\ldots,m} w_i \left\|\tilde{x} - x^i\right\|^2 \geq \frac{1-\sqrt{\alpha\gamma^*}}{2}\frac{1}{\tilde{z}_{n+1}^2} = \frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_{cp}, \tag{5.54}$$

since, by (5.49),

$$\begin{aligned}\min_{i=1,\ldots,m} w_i \left\|\tilde{x} - x^i\right\|^2 &= \min_{i=1,\ldots,m} w_i \left\|\frac{\tilde{z}}{\tilde{z}_{n+1}} - x^i\right\|^2 \\ &= \min_{i=1,\ldots,m} \frac{w_i}{\tilde{z}_{n+1}^2}\left\|\tilde{z} - x^i\tilde{z}_{n+1}\right\|^2 \\ &\geq \frac{1-\sqrt{\alpha\gamma^*}}{2}\frac{1}{\tilde{z}_{n+1}^2}. \end{aligned} \tag{5.55}$$

Now, Z^* is an optimal solution to (2.14) and $\tilde{z}_{n+1}^2 = Z_{n+1,n+1}^*$, so that by (2.14)

$$\frac{1}{\tilde{z}_{n+1}^2} = \frac{1}{Z_{n+1,n+1}^*} = \upsilon_{cp}.$$

Thus, we have shown that there exists a feasible solution $\tilde{x}$ of the original problem (2.1) such that

$$f(\tilde{x}) \ \geq \ \frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_{cp} \tag{5.56}$$

as required in part 2.

Finally, part 3 follows by combining part 2 and Theorem 2.11. □

Remark 5.11. Notice that we do not need $Z^* \succeq 0$ but only (5.45) to establish (5.17). Since (5.45) is equivalent to second-order cone constraints, see section 2.2.2 or [3, 25], it suffices to solve a second-order program (SOCP) instead of an SDP, which is computationally much more expensive.

To summarize, the key point is to find $\left(Z^*_{11}, \ldots, Z^*_{n+1,n+1}\right)$ using the SDP or SOCP relaxations. Upon obtaining $\left(Z^*_{11}, \ldots, Z^*_{n+1,n+1}\right)$, we can construct a feasible solution $\tilde{x}$ of (2.1) as follows:

1. Find $\left(Z^*_{11}, \ldots, Z^*_{n+1,n+1}\right)$, which is obtained from a solution Z^* to the SDP (2.14) or SOCP (2.24) relaxation.

2. Generate $\zeta \in \{-1,1\}^n$ as in Proposition 5.6.

3. Set

$$\tilde{x}(\zeta) = \frac{\tilde{z}}{\tilde{z}_{n+1}} = \frac{1}{\sqrt{Z^*_{n+1,n+1}}}\left(\sqrt{Z^*_{11}}\zeta_1, \ldots, \sqrt{Z^*_{nn}}\zeta_n\right). \tag{5.57}$$

5.4 Examples on the Lower Bound γ^*

We would like γ^* to be small for the bound (5.17) to be tight. In this section we provide concrete examples of calculating the lower bound γ^*.

Example 5.12. Let $\mathcal{X} = \{-1,1\}^n$, corresponding to

$$\mathcal{K} = \left\{y \in \mathbb{R}^{n+1} \,|y_j = y_{n+1}, \ j = 1, \ldots, n\right\}. \tag{5.58}$$

As $\left(Y^*_{11}, \ldots, Y^*_{nn}, Y^*_{n+1,n+1}\right)^T \in \mathcal{K}$, in particular, $Y^* \succcurlyeq 0$, we have

$$Y^*_{jj} = Y^*_{n+1,n+1}, \qquad j = 1, \ldots, n, \tag{5.59}$$

so that

$$\gamma^* = \frac{\max_{j=1,\ldots,n} Y^*_{jj}}{\sum_{j=1}^n Y^*_{jj}} = \frac{Y^*_{n+1,n+1}}{nY^*_{n+1,n+1}} = \frac{1}{n}. \tag{5.60}$$

Example 5.13. Suppose n is even, and let

$$\mathcal{X} = \left\{x \in \mathbb{R}^n \left| x_{j-1}^2 + x_j^2 = 1,\ j = 2, 4, \ldots, n \right.\right\}.$$

This corresponds to

$$\mathcal{K} = \left\{y \in \mathbb{R}^{n+1} \left| y_{j-1} + y_j = y_{n+1},\ j = 2, 4, \ldots, n \right.\right\}. \tag{5.61}$$

As $\left(Y_{11}^*, \ldots, Y_{nn}^*, Y_{n+1,n+1}^*\right)^T \in \mathcal{K}$, in particular $Y^* \succcurlyeq 0$, we have

$$\begin{aligned}\sum_{j=1}^{n} Y_{jj}^* &= (Y_{11}^* + Y_{22}^*) + \cdots + \left(Y_{n-1,n-1}^* + Y_{nn}^*\right) \\ &= \frac{n}{2} Y_{n+1,n+1}^*\end{aligned} \tag{5.62}$$

and

$$Y_{jj}^* \leq Y_{n+1,n+1}^*, \qquad j = 1, \ldots, n. \tag{5.63}$$

Thus,

$$\gamma^* = \frac{\max_{j=1,\ldots,n} Y_{jj}^*}{\sum_{j=1}^{n} Y_{jj}^*} \leq \frac{Y_{n+1,n+1}^*}{\frac{n}{2} Y_{n+1,n+1}^*} = \frac{2}{n}. \tag{5.64}$$

More examples of γ^* will be given in Chapter 6.

Chapter 6

Numerics and Examples

In this chapter we begin by considering two specific cases of the weighted maximin problem, and use the results of section 5.2 to derive approximation bounds. In the last section, we numerically compute the SDP and SOCP approximation bounds for a few specific examples, and (where applicable) compare these results to an alternate approximate solution based on a grid search method.

6.1 Approximation Bounds: Box Case

Lemma 6.1. *If $\mathcal{X} = [-1, 1]^n$ so that $\mathcal{K}$ is given by (2.7), i.e.*

$$\mathcal{K} = \left\{ y \in \mathbb{R}^{n+1} \mid y_j \leq y_{n+1},\ j = 1, \dots, n \right\}$$

then the relaxation problem (2.14) given by

$$\begin{array}{rcl} \dfrac{1}{\upsilon_{cp}} & = & \min\limits_{Z}\ Z_{n+1,n+1} \\ & & s.t.\quad w_i \left\langle A^i, Z \right\rangle \geq 1,\ i = 1, \dots, m \\ & & \qquad\ \left(Z_{11}, \dots, Z_{nn}, Z_{n+1,n+1}\right)^T \in \mathcal{K} \\ & & \qquad\quad Z \succeq 0. \end{array} \tag{6.1}$$

has at least one optimal solution Z^* with $Z^*_{jj} = Z^*_{n+1,n+1}$ for $j = 1, \dots, n$, corresponding to which $\gamma^* = 1/n$.

Proof. Since $\mathcal{X} = [-1, 1]^n$ which is compact, (2.14) has an optimal solution Z^*. Suppose that $Z^*_{\bar{j}\bar{j}} < Z^*_{n+1,n+1}$ for some $\bar{j} \in \{1, \dots, n\}$. Then, for $i = 1, \dots, m$, by (5.44) we have

$$\left\langle A^i, Z^* \right\rangle = \sum_{j=1}^{n} Z^*_{jj} + \left\| x^i \right\|^2 Z^*_{n+1,n+1} - 2 \left(x^i \right)^T z^*,$$

which is increasing with $Z^*_{\bar{j}\bar{j}}$. Thus, increasing $Z^*_{\bar{j}\bar{j}}$ to $Z^*_{n+1,n+1}$ will maintain all constraints in (2.14) to be satisfied while not changing the objective function value. Hence, $Z^*_{jj} = Z^*_{n+1,n+1}$ $\forall j = 1, \dots, n$, and so $\gamma^* = \frac{1}{n}$ as in Example 5.12. □

It follows from $\alpha = 2\ln(m/\rho)$, (5.17) and Lemma 6.1 that the feasible solution $\tilde{x}$ of (2.1) found by the SDP-based algorithm in Section 5.2 satisfies

$$f(\tilde{x}) \geq \frac{1 - \sqrt{2\ln(m/\rho)/n}}{2}\, \upsilon_{cp}. \tag{6.2}$$

Thus, $1 \geq f(\tilde{x})/\upsilon_{cp} \geq \frac{1-\sqrt{\beta}}{2}$ whenever $2\ln(m/\rho)/n \leq \beta < 1$. As we noted at the end of Section 5.2, the approximation bound (6.2) still holds when we further relax the SDP constraint $Z \succeq 0$ in (2.14) to the SOCP constraints. To our knowledge, this is the first nontrivial approximation bound for an NP-hard problem based on SOCP relaxation.

Theorem 6.2. *If $\mathcal{K}$ is given by (2.7), then when $n \to \infty$,*

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{cp}} \geq \frac{1}{2}. \tag{6.3}$$

Moreover,

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{p}} \geq \frac{1}{2}.$$

Proof. This follows from Lemma 6.1 and the fact that as $n \to \infty$, $(2\ln(m/\rho)/n) \to 0$ so that

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{cp}} = \liminf_{n\to\infty} \frac{1 - \sqrt{2\ln(m/\rho)/n}}{2} \geq \frac{1}{2}. \tag{6.4}$$

As $\upsilon_p \leq \upsilon_{cp}$ and $f(\tilde{x}) > 0$, we have

$$\frac{f(\tilde{x})}{\upsilon_p} \geq \frac{f(\tilde{x})}{\upsilon_{cp}}$$

so that

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_p} \geq \liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{cp}} \geq \frac{1}{2}.$$

□

6.2 Approximation Bounds: Ball Case

In this section we consider the case of

$$\mathcal{X} = \{x \in \mathbb{R}^n \,|\, \|x\| \leq 1\},$$

and derive analogous results for an approximation bound as we did in section 6.1.

As noted in section 5.2, we want γ^* to be small for the bound to be tight, so we begin by placing a lower bound on γ^*.

Lemma 6.3. *If $\mathcal{X} = \{x \in \mathbb{R}^n \,|\|x\| \leq 1\}$ is a Euclidean ball so that*

$$\mathcal{K} = \left\{y \in \mathbb{R}^{n+1} \,|y_1 + y_2 + \cdots + y_n \leq y_{n+1}\right\}$$

as in (2.8), then $\gamma^* \geq 1/n$. Furthermore, $\gamma^* = 1/n$ when $Z^*_{jj} = \frac{1}{n} Z^*_{n+1,n+1}$ for $j = 1, \ldots, n$.

Proof. In order to minimize $\gamma^* = \frac{\max_{j=1,\ldots,n} Z^*_{jj}}{\sum_{j=1}^n Z^*_{jj}}$ we want to maximize $\sum_{j=1}^n Z^*_{jj}$ while minimizing $\max_{j=1,\ldots,n} Z^*_{jj}$. From the definition of $\mathcal{K}$ it obvious that $\sum_{j=1}^n Z^*_{jj}$ is maximized at $Z^*_{n+1,n+1}$. Now, setting $Z^*_{jj} = \frac{1}{n} Z^*_{n+1,n+1}$ for $j = 1, \ldots, n$ yields $\sum_{j=1}^n Z^*_{jj} = Z^*_{n+1,n+1}$, and

$$\gamma^* = \frac{\max_{j=1,\ldots,n} Z^*_{jj}}{\sum_{j=1}^n Z^*_{jj}} = \frac{\frac{1}{n} Z^*_{n+1,n+1}}{Z^*_{n+1,n+1}} = \frac{1}{n}. \tag{6.5}$$

To show that this solution minimizes γ^*, suppose that $Z^*_{jj} = \frac{1}{\lambda_j} Z^*_{n+1,n+1}$ for $j = 1, \ldots, n$ so that $\sum_{j=1}^n Z^*_{jj} = Z^*_{n+1,n+1}$, but assume that $\frac{1}{\lambda_j} < \frac{1}{n}$ for some $j \in \{1, 2, \ldots, n\}$. Then we must have $\frac{1}{\lambda_k} > \frac{1}{n}$ for some $k \neq j$ in $1, \ldots, n$ (in order that $\sum_{j=1}^n Z^*_{jj} = Z^*_{n+1,n+1}$ is still satisfied). Thus,

$$\gamma^* = \frac{\max_{j=1,\ldots,n} Z^*_{jj}}{\sum_{j=1}^n Z^*_{jj}} = \frac{Z^*_{n+1,n+1} \max_{j=1,\ldots,n} \frac{1}{\lambda_j}}{Z^*_{n+1,n+1}} > \frac{\frac{1}{n} Z^*_{n+1,n+1}}{Z^*_{n+1,n+1}} > \frac{1}{n}. \tag{6.6}$$

□

Then we have an analogous result to Lemma 6.1 for the Euclidean ball bounded case:

Lemma 6.4. *If $\mathcal{K}$ is given by (2.8), then (2.14) has an optimal solution Z^* with $Z^*_{jj} = \frac{1}{n} Z^*_{n+1,n+1}$ for $j = 1, \ldots, n$ corresponding to which $\gamma^* = 1/n$.*

Proof. Since $\mathcal{X}$ is compact, (2.14) has an optimal solution, Z^*. We know that

$$\left\langle A^i, Z^* \right\rangle = \sum_{j=1}^n Z^*_{jj} + \left\|x^i\right\|^2 Z^*_{n+1,n+1} - 2\left(x^i\right)^T z^* \tag{6.7}$$

is increasing with $\sum_{j=1}^{n} Z_{jj}^*$. Then setting $Z_{jj}^* = \frac{1}{n} Z_{n+1,n+1}^*$ maximizes $\sum_{j=1}^{n} Z_{jj}^*$ at $Z_{n+1,n+1}^*$, which maintains all constraints of (2.14) while not changing the objective function value. Thus, applying Lemma 6.3, we have $\gamma^* = 1/n$. □

Then, as in section 6.1, we can set $\alpha = 2\ln(m/\rho)$, and apply (5.17) and Lemma 6.4 so that the feasible solution $\hat{x}$ of (2.1) found by the SDP-based algorithm in Section 5.2 satisfies

$$f(\hat{x}) \geq \frac{1 - \sqrt{2\ln(m/\rho)/n}}{2}\, \upsilon_{cp}. \tag{6.8}$$

Theorem 6.5. *If $\mathcal{K}$ is given by (2.8), then when $n \to \infty$,*

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{cp}} \geq \frac{1}{2}. \tag{6.9}$$

Moreover,

$$\liminf_{n\to\infty} \frac{f(\tilde{x})}{\upsilon_{p}} \geq \frac{1}{2}.$$

The proof is analogous to the proof of Theorem 6.2.

6.3 Where the SDP Relaxation Fails

In this section we consider the approximation bound (6.2) for the case that n is small and m is large. The following example shows that the SDP relaxation (2.14) can be a poor approximation of (2.1) in such a case, so (6.2) cannot be significantly improved upon.

Example 6.6. Suppose $n = 1$ (i.e. in $\mathbb{R}$), $w_i = 1$, and $x^i = \frac{2(i-1)}{m} - 1$, for $i = 1, \ldots, m+1$. Since $x^1, \ldots, x^{(m+1)}$ form a grid of equal spacing on $\mathcal{X} = [-1, 1]$, we will have an optimal point (call it x^*), at the halfway point between any two x^i. For convenience, choose $x^1 = -1$ and $x^2 = -1 + \frac{2}{m}$, so that $x^* = -1 + \frac{1}{m}$. Then the optimal solution is given by

$$\begin{aligned} \upsilon_p &= \left|x^* - x^1\right|^2 \\ &= \left|-1 + \frac{1}{m} - (-1)\right|^2 \\ &= \left|\frac{-1}{m}\right|^2 \\ &= \frac{1}{m^2}. \end{aligned}$$

On the other hand, $Z = I$ is a feasible solution of (2.14) with objective function value of 1.
To see that $Z = I_{2\times 2}$ is feasible, need to show the following:

- $\left\langle A^i, Z \right\rangle \geq 1$. Recall that $\left\langle A^i, Z \right\rangle = \text{trace}[A^i Z]$. Then we have

$$A^i Z = A^i I = A^i = \begin{bmatrix} 1 & -\left(\frac{2(i-1)}{m} - 1\right) \\ -\left(\frac{2(i-1)}{m} - 1\right) & \left(\frac{2(i-1)}{m} - 1\right)^2 \end{bmatrix}$$

$$\Rightarrow \left\langle A^i, Z \right\rangle = 1 + \left(\frac{2(i-1)}{m} - 1\right)^2 \geq 1$$

- Here $\mathcal{X} = [-1, 1]$ so that $\mathcal{K} = \left\{ y \in \mathbb{R}^2 \,| y_1 \leq y_2 \right\}$. Then $(Z_{11}, Z_{22}) = (1, 1) \in \mathcal{K}$ is obvious.

- $Z = I \succeq 0$.

Thus we have that $\frac{1}{\upsilon_{cp}} = \min_Z Z_{n+1,n+1} \leq 1$ so that $\upsilon_{cp} \geq 1$.

Hence, $\upsilon_{cp} \geq 1$ is significantly larger than $\upsilon_p = \frac{1}{m^2}$. In fact,

$$\upsilon_{cp} \geq m^2 \upsilon_p$$

which shows that in the case of n small and m large, υ_{cp} is a poor approximation for υ_p.

6.3.1 Necessary Condition for Non-trivial Bounds

In order for the approximation bounds to be useful, we need to ensure that they yield non-trivial results (bounds > 0). In this section we give conditions on the values of m and n for the case that $\mathcal{X}$ is a box or a ball and thus $\gamma^* = 1/n$ which will ensure that the approximation bounds provided by the SDP and SOCP relaxation problems are bounded away from 0.

To obtain a non-trivial lower bound, we require that

$$\frac{1 - \sqrt{2\ln(m/\rho)/n}}{2} \upsilon_{cp} > 0. \tag{6.10}$$

Since $\upsilon_{cp} > 0$ (since $\upsilon_{cp} \geq \upsilon_p$), we need only verify that

$$\frac{1-\sqrt{2\ln(m/\rho)/n}}{2} > 0 \tag{6.11}$$

$$\Leftrightarrow 1 > \sqrt{2\ln(m/\rho)/n} \tag{6.12}$$

$$\Leftrightarrow \frac{n}{2} > \ln(\frac{m}{\rho}) \tag{6.13}$$

$$\Leftrightarrow \frac{m}{\rho} < \exp\left(\frac{n}{2}\right) \quad \Rightarrow m < \exp\left(\frac{n}{2}\right). \tag{6.14}$$

Thus, we can guarantee that the approximation bounds provided by the SDP and SOCP based algorithms will yield a non-trivial lower bound whenever n and m satisfy $m/\rho < \exp(\frac{n}{2})$.

Table 6.1 shows the largest value of m that will yield a non-trivial bound for a given value of n (setting $\rho = 1$): $m = \lfloor \exp \frac{n}{2} \rfloor$, i.e $m =$ the largest integer $\leq \exp \frac{n}{2}$.

n	1	2	3	4	5	6	7	8	9	10
m	1	2	4	7	12	20	33	54	90	148

Table 6.1: Maximal value of m yielding a non-trivial approximation.

Note that this means that we can *not* guarantee a non-trivial bound for cases that $m \geq \exp(\frac{n}{2})$.

Remark 6.7. Consider the function

$$g : (0,1] \to \mathbb{R} : \rho \mapsto \frac{1-\sqrt{2/n\ln(m/\rho)}}{2}.$$

Since $\rho \mapsto \ln(m/\rho)$ is a decreasing function, g is an increasing function, therefore $g(1) = \max_{\rho\in(0,1]} g(\rho)$.

6.4 An Alternate Approach

Of course, SDP relaxation is not the only way to construct an approximate solution of (2.1). For example, consider the following simple procedure for constructing an approximate solution that is not based on SDP.

Definition 6.8. The ceiling, or least-integer function, is a function from the real numbers $\mathbb{R}$ to the integers $\mathbb{Z}$, that maps any real number x to the the smallest integer y such that $y \geq x$. It is denoted by $\lceil x \rceil$. For example, we have $\lceil 1.5 \rceil = 2$, $\lceil -3.2 \rceil = -3$ and $\lceil 4 \rceil = 4$.

Example 6.9. Let $N := \lceil (m+1)^{1/n} \rceil$. We partition $\mathcal{X} = [-1,1]^n$ into N^n boxes of length $2/N$ in each dimension. Since this box contains the Euclidean ball centered at $\hat{x}$ of radius $1/N$, we have $\|\hat{x} - x^i\| \geq 1/N$ for all i. Hence

$$f(\hat{x}) \;=\; \min_{i=1,\ldots,m} w_i \left\|\hat{x} - x^i\right\|^2 \;\geq\; \frac{\min_i w_i}{N} \;=\; \frac{\min_i w_i}{\lceil (m+1)^{1/n} \rceil}.$$

Also $\hat{x} \in \mathcal{X}$. Assuming that $x^1, \ldots, x^m$ are in $\mathcal{X}$ so that $\|x - x^i\|^2 \leq 4n$ for all $x \in \mathcal{X}$, then $\upsilon_p \leq 4n \min_i w_i$. Therefore

$$f(\hat{x}) \geq \frac{1}{4n \lceil (m+1)^{1/n} \rceil} \; \upsilon_p.$$

The approximate solution $\hat{x}$ is good when $x^1, \ldots, x^m$ are distributed "uniformly" over $\mathcal{X}$ but can be arbitrarily bad when $x^1, \ldots, x^m$ are clustered near each other. In particular, unlike (6.2), the above approximation bound tends to zero as $n \to \infty$; namely

$$\lim_{n\to\infty} \frac{1}{4n\lceil (m+1)^{1/n} \rceil} = 0. \tag{6.15}$$

6.5 Numerical Results

In this section, we use `cvx` [11], an optimization package for Matlab, to numerically compare the SDP and SOCP approximation bounds.

`cvx` is a modeling system for disciplined convex prgramming (DCP). DCP's are convex programming problems that are described using a limited set of construction rules. `cvx` can solve standard problems such as linear programs (LPs), quadratic programs (QPs), second-order cone programs (SOCPs) and semidefinite programs (SDPs), as well as many other more complex convex optimization problems.

Since the original optimization problems in the examples do not have a simple closed form solution, a Matlab program was written to find the approximate optimal point, yielding an approximate solution to the original weighted maximin dispersion problem (see Appendix C). Note that this code depends upon the full factorial (`fullfact`) function in Matlab, so that for high dimensional problems or very large grid sizes the computational time is very high, and thus this program is not meant to take the place of more complex solvers.

Example 6.10. Suppose that $\mathcal{X} = [-1,1]^5$, $w_i = 1$ for $i = 1, \ldots, 3$ and

$$x^1 = (1,0,0,0,0) \tag{6.16}$$
$$x^2 = (0,0,1,0,0) \tag{6.17}$$
$$x^3 = (0,0,0,0,1) \tag{6.18}$$

so that we have the following equally-weighted maximin dispersion problem:

$$\max_{x\in[-1,1]^5} \min_{i=1,2,3} \left\| x - x^i \right\|^2 . \tag{6.19}$$

Remark 6.11. We can easily derive the exact optimal solution for (6.19). Set $x = (x_1, x_2, x_3, x_4, x_5)$. Then

$$\begin{aligned}
&\min_{i=1,2,3} \left\| x - x^i \right\|^2 \\
&\quad = \min\left\{(x_1-1)^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2, x_1^2 + x_2^2 + (x_3-1)^2 + x_4^2 + x_5^2,\right. \\
&\qquad \left. x_1^2 + x_2^2 + x_3^2 + x_4^2 + (x_5-1)^2\right\} \\
&\quad = \min\left\{(x_1-1)^2 + x_3^2 + x_5^2, x_1^2 + (x_3-1)^2 + x_5^2, x_1^2 + x_3^2 + (x_5-1)^2\right\} \\
&\qquad + x_2^2 + x_4^2.
\end{aligned}$$

Thus, the optimal solution requires that $x_1 = x_3 = x_5 = -1$. Then we can set $x_1 = x_3 = x_5 = -1$ and $x_2 = \pm 1$ and $x_4 = \pm 1$ to obtain an optimal point, corresponding to the optimal solution of

$$\upsilon_p = 4 + 2 + 2 = 8.$$

First, we obtain an approximate solution of (6.19) on a grid using the Matlab code from Appendix A:
Using grid sizes of 5, 9 and 11, we have an approximate solution of

$$\upsilon_p = 8 \tag{6.20}$$

in each case, with the optimal value being achieved at one of four points:

$$\begin{aligned}
x_1^* &= (-1,-1,-1,-1,-1) \\
x_2^* &= (-1,1,-1,-1,-1) \\
x_3^* &= (-1,-1,-1,1,-1) \\
x_4^* &= (-1,1,-1,1,-1) \,.
\end{aligned} \tag{6.21}$$

Now, we want to solve the SDP and SOCP relaxation problems using `cvx` and Matlab to compare the approximation bounds. Since $\mathcal{X} = [-1,1]^5$ is a polyhedral set having a tractable algebraic representation, equation (2.14) yields the following semidefinite program:

$$\begin{aligned} \frac{1}{\upsilon_{cp}} &= \min_{Z} \; Z_{66} \\ &\quad \text{s.t.} \quad \left\langle A^i, Z \right\rangle \geq 1, \; i = 1,2,3 \\ &\qquad\quad (Z_{11}, \ldots, Z_{66})^T \in \mathcal{K} \\ &\qquad\quad Z \succeq 0. \end{aligned} \tag{6.22}$$

while equation (2.24) yields the following second-order cone program:

$$\begin{aligned} \frac{1}{\upsilon_{socp}} &= \min_{Z} \; Z_{66} \\ &\quad \text{s.t.} \quad \left\langle A^i, Z \right\rangle \geq 1, \; i = 1,2,3 \\ &\qquad\quad (Z_{11}, \ldots, Z_{66})^T \in \mathcal{K} \\ &\qquad\quad \begin{bmatrix} Z_{jj} & Z_{j,6} \\ Z_{j,6} & Z_{6,6} \end{bmatrix} \succeq 0, \quad j = 1, \ldots, 5. \end{aligned} \tag{6.23}$$

where $\mathcal{K} = \left\{ y \in \mathbb{R}^6 \, | y_j \leq y_6 \;\; \text{for } j = 1, \ldots, 5 \right\}$.

Solving both the SDP and SOCP relaxation problems using `cvx` yield an optimal solution value of

$$\frac{1}{\upsilon_{cp}} = \frac{1}{\upsilon_{socp}} = 0.125. \tag{6.24}$$

See Appendix B. Interestingly, $\frac{1}{\upsilon_{cp}} = 0.125 \;\Rightarrow \upsilon_{cp} = 8$, so that in this case, $\upsilon_{cp} = \upsilon_p$. However, this is not the case in general, and from our results in this paper we can only say with certainty that

$$\upsilon_p \geq \frac{1 - \sqrt{2\ln(3/\rho)/5}}{2} \upsilon_{cp}.$$

Substituting $\upsilon_{cp} = 8$ and choosing $\rho = 1$ (recall that $0 < \rho \leq 1$), we have the following lower bound on the optimal solution:

$$\upsilon_p \geq \frac{1 - \sqrt{2\ln(3)/5}}{2} 8 \approx 1.348.$$

Since this is a case where n and m are both quite small, υ_{cp} is not a very good approximation of the true solution υ_p. However, in cases where $\mathcal{X}$ is a box and n and m are small, it is certainly feasible to use the Matlab code provided in Appendix A to obtain an approximate solution to (2.1). We begin by examining the case where $n = 5$, which is still feasible to solve using the code in Appendix A to yield an approximation bound and we compare the results to those produced using `cvx`.

Example 6.12. Suppose that $\mathcal{X} = [-1,1]^5$, $w_i = 1$ for $i = 1, \ldots, 3$ and we have $m = 3, 4, \ldots, 12$ points where each point x^j is randomly selected from $\mathcal{X}$. Using a grid size of 11 the following table compares the lower bound based on the optimal values υ_{cp} and υ_{socp} resulting from solving the SDP and SOCP relaxation problems with `cvx` to the approximate optimal solution υ_g found using the grid solver. Table 6.2 summarizes the results of using the grid solver and CVX (setting $\rho = 1$). Since n is small we expect the bound to not be very good, and this expectation is confirmed form the reults in Table 6.2. However, for small values of m the bound will significantly help in narrowing the search space for finding the optimal solution.

m	$\frac{1-\sqrt{2\ln(m)/5}}{2}\upsilon_{cp}$	$\frac{1-\sqrt{2\ln(m)/5}}{2}\upsilon_{socp}$	υ_gGrid
3	1.2100	1.2100	6.9436
4	0.8238	0.8238	5.7219
5	0.7549	0.7549	6.1027
6	0.5184	0.5184	5.3748
7	0.5189	0.5189	5.8145
8	0.3272	0.3272	5.4333
9	0.2482	0.2482	5.7362
10	0.1341	0.1341	5.9963
11	0.0868	0.0868	4.9257
12	0.0119	0.0119	5.0090

Table 6.2: Numerical Results for Large $n = 5$.

What we are more interested in is cases where n and m are large, so that applying the grid program becomes impossible. In particular, we examine a few cases where n is "large" ($n > 5$) and $m < \exp\left(\frac{n}{2}\right)$ (setting $\rho = 1$), which guarantees that the SDP and SOCP based algorithms will yield a non-trivial lower bound.

Example 6.13. Let $\mathcal{X} = [-1,1]^n$, $w_i = 1$ for $i = 1, \ldots, m$, and $x^1, \ldots, x^m$ are a set of randomly generated points in $\mathcal{X}$. The following table gives the lower bound to the original problem 2.1 based on the approximate solutions υ_{cp} and υ_{socp} resulting from using `cvx` to solve the relaxation problems (2.14) and (2.24):

From this example it is once again apparent that the optimal solution to the simpler SOCP relaxation problem is identical to the optimal value of

n	m	$\frac{1-\sqrt{2\ln(m)/n}}{2}\upsilon_{cp}$	$\frac{1-\sqrt{2\ln(m)/n}}{2}\upsilon_{socp}$
6	20	0.0025	0.0025
7	33	0.0027	0.0027
8	54	0.0061	0.0061
9	90	0.0001	0.0001
10	148	0.0016	0.0016
11	244	0.0016	0.0016
12	403	0.0006	0.0006
13	665	0.0001	0.0001

Table 6.3: Numerical Results for Large n and m.

the SDP relaxation. Since the original problem is NP-hard, and the values of n and m are large enough that applying the grid solution is not feasible, we do not have a way to compare these results to the actual solution of the problem. However, since all of the bounds given are strictly greater than zero, Table 6.3 at least confirms that the lower bound for (2.1) resulting from the SDP and SOCP relaxation problems is a non-trivial bound.

Chapter 7

Conclusions and Future Work

7.1 Results

The purpose of this thesis has been to develop SDP and SOCP convex relaxations for tackling the weighted maximin dispersion problem, which is NP-hard.

We began by describing the general weighted maximin dispersion problem in Section 2.1, and defining the convex relaxation problems in Section 2.2. Next, in Section 3.2, we used theoretical results from nonsmooth analysis to derive a necessary condition for the optimal solutions of the equally weighted maximin problem. Provided that the set $\mathcal{X}$ is convex, we showed that $\bar{x} \in \operatorname{int}(\mathcal{X})$ is a possible solution of

$$\max_{x \in \mathcal{X}} \min_{i=1,\ldots,m} w \left\| x - x^i \right\|^2 \tag{7.1}$$

only if $\bar{x} \in \operatorname{conv}\left\{x^i \,|i \in I(\bar{x})\right\}$.

Chapter 4 was dedicated to the theory of NP-completeness. While it is well known that the general weighted maximin distance problem is NP-hard, we have provided a proof to show that even in the case of equally weighting and $\mathcal{X}$ is a box, the problem is NP-hard. That is, the problem given by

$$f(x) = \max_{x \in [-1,1]^n} \min_{i=1,\ldots,m} w \left\| x - x^i \right\|^2 \tag{7.2}$$

is NP-hard. Also in this Chapter, we investigated an heuristic approach based on partitioning the space $\mathcal{X}$ into Voronoi cells and considering subproblems based on restricting x to be in one of m Voronoi cells. Again, we showed that even in the case of equal weighting and $\mathcal{X} = \mathbb{R}^n$, the restricted subproblems are NP-hard.

Finally, in Chapter 5, we used the convex relaxations problems given in Section 2.2 to develop an algorithm to construct an approximate solution to the general weighted maximin dispersion problem. Pulling results from statistics, convex analysis and matrix algebra, we showed that there exists a feasible solution $\tilde{x}$ to the original problem

$$\max_{x\in\mathcal{X}} \min_{i=1,\ldots,m} w_i \left\|x - x^i\right\|^2 \tag{7.3}$$

such that

$$f(\tilde{x}) \geq \frac{1-\sqrt{\alpha\gamma^*}}{2}\upsilon_{cp} \tag{7.4}$$

where $1/\upsilon_{cp}$ is the optimal value of the SDP relaxation problem (2.14), and

$$\gamma^* = \frac{\max_{j=1,\ldots,n} Z^*_{jj}}{\sum_{j=1}^n Z^*_{jj}},\ \alpha = 2\ln\left(\frac{m}{\rho}\right). \tag{7.5}$$

Chapter 6 began by focusing on two specific examples - the case that $\mathcal{X}$ is a box, and the case that $\mathcal{X}$ is a ball. We showed that in either case, there exists an optimal solution to problem (2.10) with $Z^*_{jj} = Z^*_{n+1,n+1}$ for $j = 1,\ldots,n$ and $\gamma^* = 1/n$. Finally, Chapter 6 finished with a few numerical examples to apply the SDP relaxation based algorithm developed in Chapter 5. Matlab code was provided for implementing the algorithm developed in Chapter 5, as well as Matlab code that was developed for finding approximate solutions over a grid.

7.2 Future Work

Possible directions for further investigation into the weighted maximin dispersion problem and the approximation bounds derived in this paper are:

1. Can we extend Lemma 6.1, or Lemma 6.4 to other $\mathcal{X}$ of the form (2.6)? Or more generally, when $\mathcal{X}$ is invariant under permutation.

2. Is (2.1) NP-hard for other choices of $\mathcal{X}$ besides the box (e.g., $\mathcal{X}$ is the Euclidean ball)?

3. Can the above results be extended to locate multiple points in $\mathcal{X}$? And maxsum dispersion problems?

4. Can we improve the approximation bound by using the rank reduction scheme of Goemans-Williamson [14] (also see [28]) or its variant proposed by Bertsimas and Ye [5]? In particular, instead of generating ζ

uniformly on $\{-1,1\}^n$, generate $\xi \sim N(0, Z^*)$, the real-valued normal distribution, and set

$$\zeta_j = \begin{cases} 1 & \text{if } \xi_j > 0; \\ -1 & \text{else,} \end{cases} \quad j = 1, \ldots, n.$$

Note that this requires $Z^* \succeq 0$. Analyzing this will likely require new large deviation estimates.

5. Create specialized algorithms to solve the SOCP relaxation more efficiently when n and m are large (and large number of points to locate).

6. Find the lower bound for the performance of the SOCP relaxation.

7. One always has $\upsilon_{cp} \geq \upsilon_p$. Under what conditions do the convex relaxation problems (2.14) or (2.24) yield the exact solution. That is, under what conditions will we have $\upsilon_{cp} = \upsilon_p$?

8. Do some comparisons of our SDP and SOCP relaxations to existing relaxation methods, such as [16, 30, 33].

9. In the thesis, we are considering the Maximin problem only for m points in $\mathbb{R}^n$. What happen if we consider the Maximin problem for m convex sets in $\mathbb{R}^n$?

Bibliography

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):13–51, 1995. → pages 24

[2] Jonathan M. Borwein and Adrain S. Lewis. *Convex Analysis and Nonlinear Optimization, Theory and Examples.* Springer, 2000. → pages 15

[3] A. Ben-Tal, A. Nemirovski, and C. Roos. Robust solutions of uncertain quadratic and conic-quadratic problems. *SIAM Journal on Optimization*, 13(2):535–560, 2002. → pages 54, 67

[4] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004. → pages 19, 20

[5] D. Bertsimas and Y. Ye. Semidefinite relaxations, multivariate normal distribution, and order statistics. In D.Z. Du and P.M. Pardalos, editors, *Handbook of Combinatorial Optimization*, volume 3, pages 1–19. Kluwer Academic Publishers, 1998. → pages 81

[6] Pierluigi Crescenzi and Viggo Kann. A compendium of np optimization problems, 1998. → pages 23

[7] Frank H. Clarke. *Optimization and Nonsmooth Analysis.* Society for Industrial and Applied Mathematics, 1990. → pages 19

[8] Stephen Cook. The p vs np problem, 2000. → pages 22

[9] Kenneth R. Davidson and Allan P. Donsig. *Real Analysis and Applications.* Springer, 2010. → pages 2, 4

[10] B. Dasarathy and Lee J. White. A maxmin location problem. *Operations Research*, 28(6):1385–1401, 1980. → pages 26

[11] Micheal Grant and Stephen Boyd. `cvx` user's guide; for `cvx` version 1.21. User's Guide, September 2010. → pages 24, 75

[12] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Bell Telephone Laboratories, Incorporated, 1979. → pages 21, 22, 23, 24, 49, 53

[13] M. Grotschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, Berlin, 1998. → pages 24

[14] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995. → pages 24, 81

[15] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization*, 6(2):342–361, 1996. → pages 24

[16] M. E. Johnson, L. M. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26(2):131 – 148, 1990. → pages 26, 82

[17] R.M. Karp. Reducibility among combinatorial problems. In R.E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, New York, 1972. → pages 23

[18] Carl D. Meyer. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics, 2000. → pages 7

[19] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, New Jersey, 1970. → pages 15, 16

[20] S. S. Ravi, D. J. Rosenkrantz, and G. K. Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, 1994. → pages 44

[21] F. Rendl, G. Rinaldi, and A. Wiegele. Solving max-cut to optimality by intersecting semidefinite and polyhedral relaxations. *Mathematical Programming*, 121(2):307–335, 2010. → pages 24

[22] R.T. Rockafellar and Roger J-B. Wets. *Variational Analysis*. Springer-Verlag, 1998. → pages 9, 13, 14, 16

[23] Winfried Schirotzek. *Nonsmooth Analysis*. Springer-Verlag, Berlin, 2007. → pages 19

[24] Shayle R. Seale. *Matrix Algebra Useful for Statistics*. John Wiley and Sons Inc., 1982. → pages 5, 6

[25] J.F. Sturm. Using sedumi 1.02, a matlab* toolbox for optimization over symmetric cones (updated for version 1.05). Technical report, Department of Econometrics, Tilburg University, Tilburg, The Netherlands, August 1998 - October 2001. → pages 24, 67

[26] Jerome Sacks, William J. Welch, Toby J. Mitchell, and Henry P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):pp. 409–423, 1989. → pages 26

[27] M.J. Todd. Semidefinite optimization. *Acta Numerica*, 10:515–560, 2001. → pages 21, 24

[28] Paul Tseng. Further results on approximating nonconvex quadratic optimization by semidefinite programming relaxation. *SIAM Journal on Optimization*, 14(1):268–283, 2003. → pages 81

[29] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, 1996. → pages 20, 21

[30] D.J. White. A heuristic approach to a weighted maxmin dispersion problem. *IMA Journal of Management Mathematics*, 7(3):219–231, 1996. → pages 26, 49, 53, 82

[31] Wikipedia. Complexity class. → pages 22, 23

[32] H. Wolkowicz, R. Saigal, and L. Vandenberghe. *Handbook of Semidefinite Programming. Theory, Algorithms, and Applications.* Kluwer Academic Publishers, Boston, MA, 2000. → pages 21, 24

[33] Shuzhong Zhang. Quadratic maximization and semidefinite relaxation. *Mathematical Programming*, 87(3):453–465, 2000. → pages 82

Appendix A

Matlab Code

The following matlab code was used to produce 3-dimensional pictures of the minimum distance function. Note that this function calls the `MinNorm` function, the code for which is also given.

```
% Plot the minimum distance function over [-1,1]^2

% define the range over which to calculate the minimum distances
x1=(-1:0.01:1)';
x2=(-1:0.01:1)';

xmat=[kron(x1,ones(length(x2),1)) kron(ones(length(x1),1),x2)];

[x y]=meshgrid(x1,x2);

% create two sets (S and T) of 10 points
% in the 2-D unit square
% S is a "nice" set of points
S=[[-1,1];[1,-1];[1,1];[-1,-1];[0,0]];
% T is a random set of points
t=randi([-10,10],[10,2]);
T=t/10;

% get the function values for S and T
fminS=MinNorm(xmat,S);
fminT=MinNorm(xmat,T);

zS=reshape(fminS,length(x1),length(x2));
zT=reshape(fminT,length(x1),length(x2));

figure(1)
surf(x,y,zS,'edgecolor', 'none')
print('-f1', 'MinDistExS.eps','-depsc2')
```

```
figure(2)
surf(x,y,zT,'edgecolor', 'none')
print('-f2', 'MinDistExT.eps','-depsc2')
```

The following code was written to estimate the optimal solution to (2.1) when $\mathcal{X} = [-1,1]^n$ and $w_i = 1$ for all i. Using a grid of size $q+1$, we find the point in $x^* \in \mathcal{X}$ that is furthest away from the given set of points S, using the maximin distance criteria.

```
function [Mx, P]=MaximinGrid(q,S)
%This function takes a set of points S and finds the
% furthest point in [-1,1]^{n} based on the maximin distance
% criteria, using a q+1 grid
% S is an m by n array, where n is the number of dimensions
% and m is the number of points (each ROW is a point)

[m n]=size(S);

% create the q+1 grid on [-1,1]^n:
x=((q+1)*ones([n,1]));
% a vector of length n with entries q+1
X=fullfact(x);
% a grid of size (q+1)^{2} by n
% scale the grid to be in [-1,1]
[r c]=size(X);

Y=-1+2*((X-1)/q);

d=999*ones([r,1]);
% a vector that stores the mindist from each point
% on the grid to the closest point in S

for ii=1:r
% for each point on the grid
   k=0;
   % condition for the while loop
   while k==0
   % enter loop
        for jj=1:m
```

```
        % for each point in S
            D = sum( (Y(ii,:)-S(jj,:)).^2);
            % calculate the distance
            if D==0
            % if the grid point is in S
               d(ii)=D;
               % so it won't have the max value of 1!
               k=1;
               % skip this point by exiting the while loop
               break;
            elseif D < d(ii)
            % if the new distance is smaller but not equal 0
               d(ii) = D;
               % store the new mindist
            end
        end
        k=1;
   end
end

Mx=max(d);
% get the maximin distance
ind=(d==Mx);
P=Y(ind,:);

Mx=max(d);

end
```

Appendix B

Matlab Output

In this section we provide the Matlab output from Example 6.10.

Solve Example 6.10 using the MaximinGrid function with different grid sizes $q1 = 4$, $q2 = 8$ and $q3 = 10$:

```
% Find the approximate optimal point for Numerics Example 1
% Do a couple of trials with different grid sizes

% Define the given set of points S
x1 = [1,0,0,0,0];
x2 = [0,0,1,0,0];
x3 = [0,0,0,0,1];
S = [x1; x2; x3];

% Choose the different mesh sizes for the grid
q1=4;
q2=8;
q3=10;

% Find the optimal points using the different grids
[d1 p1]=MaximinGrid(q1,S);
d1 =  8
p1 =
    -1    -1    -1    -1    -1
    -1     1    -1    -1    -1
    -1    -1    -1     1    -1
    -1     1    -1     1    -1

[d2 p2]=MaximinGrid(q3,S);
d2 =  8
[d3 p3]=MaximinGrid(q3,S);
d3 =  8
```

Output from solving the SDP relaxation for Example 6.10:

```
Calling sedumi: 29 variables, 8 equality constraints
------------------------------------------------------------
SeDuMi 1.21 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500
eqs m = 8, order n = 15, dim = 45, blocks = 2
nnz(A) = 39 + 0, nnz(ADA) = 64, nnz(L) = 36
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            4.82E+000 0.000
  1 : -9.09E-002 1.99E+000 0.000 0.4133 0.9000 0.9000   2.47  1  1  2.7E+000
  2 :  1.32E-001 8.58E-001 0.000 0.4308 0.9000 0.9000   4.31  1  1  4.4E-001
  3 :  1.26E-001 2.41E-001 0.000 0.2805 0.9000 0.9000   1.57  1  1  1.0E-001
  4 :  1.25E-001 1.18E-002 0.000 0.0489 0.9900 0.9900   1.22  1  1  4.8E-003
  5 :  1.25E-001 3.71E-005 0.000 0.0032 0.9990 0.9990   1.01  1  1  1.5E-005
  6 :  1.25E-001 7.28E-006 0.000 0.1960 0.9000 0.8959   1.00  1  1  2.9E-006
  7 :  1.25E-001 5.39E-007 0.423 0.0741 0.9900 0.9438   1.00  1  1  2.2E-007
  8 :  1.25E-001 6.57E-009 0.000 0.0122 0.9902 0.9900   1.00  1  1  3.3E-009

iter seconds digits       c*x               b*y
  8      0.8   8.1  1.2500000099e-001  1.2499999993e-001
|Ax-b| =  2.7e-009, [Ay-c]_+ =  3.7E-010, |x|= 5.3e-001, |y|= 3.5e-001

Detailed timing (sec)
   Pre          IPM          Post
4.836E-001    7.800E-001    1.560E-001
Max-norms: ||b||=1, ||c|| = 1,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 316.544.
------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +0.125

Z =

    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.0000    0.1250   -0.0000   -0.0000   -0.0000    0.0000
    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.0000   -0.0000   -0.0000    0.1250   -0.0000    0.0000
    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.1250    0.0000   -0.1250    0.0000   -0.1250    0.1250
```

Output from solving the SOCP relaxation for Example 6.10:

```
Calling sedumi: 23 variables, 12 equality constraints
------------------------------------------------------------
SeDuMi 1.21 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500
eqs m = 12, order n = 19, dim = 29, blocks = 6
nnz(A) = 47 + 0, nnz(ADA) = 144, nnz(L) = 78
 it :     b*y       gap    delta  rate   t/tP*  t/tD*   feas cg cg  prec
  0 :            3.80E+000 0.000
  1 : -1.05E-001 1.62E+000 0.000 0.4249 0.9000 0.9000   2.47  1  1  2.8E+000
  2 :  1.35E-001 6.38E-001 0.000 0.3950 0.9000 0.9000   4.86  1  1  3.3E-001
  3 :  1.24E-001 5.51E-002 0.000 0.0863 0.9900 0.9900   1.45  1  1  2.4E-002
  4 :  1.25E-001 5.14E-004 0.358 0.0093 0.9945 0.9945   1.07  1  1  2.3E-004
  5 :  1.25E-001 1.23E-005 0.000 0.0240 0.9900 0.9900   1.00  1  1  5.9E-006
  6 :  1.25E-001 6.14E-007 0.082 0.0498 0.9900 0.9356   1.00  1  1  3.1E-007
  7 :  1.25E-001 1.11E-008 0.000 0.0182 0.9905 0.9900   1.00  1  1  8.7E-009

iter seconds digits       c*x               b*y
  7      0.1   7.5  1.2500000462e-001  1.2500000087e-001
|Ax-b| =  3.6e-009, [Ay-c]_+ =  2.7E-009, |x|= 5.0e-001, |y|= 3.5e-001

Detailed timing (sec)
   Pre          IPM          Post
6.240E-002    6.240E-002    3.120E-002
Max-norms: ||b||=1, ||c|| = 1,
Cholesky |add|=1, |skip| = 0, ||L.L|| = 108.52.
------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +0.125

Z =

    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.0000    0.1250   -0.0000   -0.0000   -0.0000    0.0000
    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.0000   -0.0000   -0.0000    0.1250   -0.0000    0.0000
    0.1250   -0.0000    0.1250   -0.0000    0.1250   -0.1250
   -0.1250    0.0000   -0.1250    0.0000   -0.1250    0.1250
```