# SOME ALGORITHMS FOR LARGE-SCALE LINEAR AND CONVEX MINIMIZATION IN RELATIVE SCALE

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Peter Richtárik

August 2007

# SOME ALGORITHMS FOR LARGE-SCALE LINEAR AND CONVEX MINIMIZATION IN RELATIVE SCALE

Peter Richtárik, Ph.D.

Cornell University 2007

This thesis is concerned with the study of algorithms for approximately solving large-scale linear and nonsmooth convex minimization problems within a prescribed relative error $\delta$ of the optimum. The methods we propose converge in $O(1/\delta^2)$ or $O(1/\delta)$ iterations of a first-order type. While the theoretical lower iteration bound for approximately solving (in the absolute sense) nonsmooth convex minimization problems in the black-box computational model of complexity is $O(1/\epsilon^2)$, the algorithms developed in this thesis are able to perform better by effectively utilizing the information about the *structure* of the problems.

Chapter 1 contains a brief account of the relevant part of complexity theory for convex optimization problems. This is done in order to be able to better communicate the proper setting of our work within the current literature. We finish with concise synopses of the following chapters.

In Chapter 2 we study the general problem of unconstrained convex minimization in relative scale. Algorithms of this type are hard to find in the literature and are known perhaps only for a narrow class of specialized transportation problems. It was recently suggested by Nesterov [23], [22] that this problem can be efficiently treated via a conic reformulation and by utilizing the information gained from the computation of a pair of John ellipsoids for the subdifferential of the objective

function evaluated at the origin. Our main contribution is the improvement of the theoretical performance of the algorithms in the cited papers by incorporating a simple bisection idea. We also show that it is possible to design potentially more practical "nonrestarting" versions of these methods at no or only negligible cost in their theoretical guarantees.

In Chapter 3 we consider the geometric problem of finding the intersection of a line and a centrally symmetric convex body $\mathcal{Q}$ given as the convex hull of a collection of points. Our algorithms produce a sequence of ellipsoids inscribed in $\mathcal{Q}$, "converging" towards the intersection points. It turns out that in doing so we simultaneously solve a number of closely related problems such as the problem of finding the minimum $\ell_1$ norm solution of a full rank underdetermined linear system, minimizing the maximum of absolute values of linear functions, or linear optimization over the polytope polar to $\mathcal{Q}$. We finish the discussion by describing applications to truss topology design and optimal design of statistical experiments.

## BIOGRAPHICAL SKETCH

Peter was born at midnight in mid-September of 1977 in Nitra, the historical center Slovakia. In 1996 he started his studies at Comenius University in Bratislava, simultaneously at the Faculty of Management and Faculty of Mathematics, Physics and Informatics. He graduated in 2001 with two bachelor's and one master's degree, all summa cum laude. In the Fall of 2002, he was fortunate to begin his Ph.D. studies at the School of Operations Research and Industrial Engineering at Cornell University located in gorgeous Ithaca. In the Summer of 2003 Peter married Marianna Ivanová and three years later, Amália was born. These were the greatest moment of his personal life. He received his Doctor of Philosophy degree in Operations Research in August 2007 and will be joining the Department of Mathematical Engineering (INMA) of Catholic University of Louvain, Belgium, as a postdoctoral researcher starting in September 2007.

To my wife Marianna,

daughter Amália,

and everybody I love.

## ACKNOWLEDGEMENTS

This thesis would not be possible without the constant support, limitless kindness and inspiration pouring from the heart and mind of my advisor, Professor Michael J. Todd. His patience and optimism in times of slow progress always managed to recharge my batteries. Mike has been a terrific person and a true role model. Although I was not aware of it during the process, I can see clearly now that under his guidance I have learned to think creatively about problems and see light by asking the right questions. For all of this I am enormously indebted to him.

I wish to express my thanks to the members of my special committee, Professors Adrian S. Lewis, Stephen A. Vavasis and Leonard Gross. I have truly enjoyed Adrian's "Nonlinear Programming" and "Convex Analysis" courses. His way of exposition seems to have settled in me as an ideal for my own carrier. I must express my gratitude to Professor Charles Van Loan who, during his sabbatical, agreed to serve as a proxy for Stephen Vavasis at my defense. His "Matrix Computations" course, a rare blend of intuition and computation, was both illuminating and entertaining.

There a number of other Cornell faculty who influenced me greatly in various ways. For example, Professor James Renegar introduced me to the field of mathematical programming in my first year, and I happened to fall in love with it. Jim also seemed to truly enjoy my budding photography skills. Alexander Bendikov's and Leslie Trotter's teaching style for some reason reminded me of home. Although I am tempted to go on and tell a little story about every faculty I had the privilege to spend time with and learn from, this would surely take several pages. Let me therefore at least mention some names. Tom Coleman, Shane Henderson, Sid

Resnick, Genna Samorodnitsky, David Shmoys, Éva Tardos, Huseyin Topaloglu and David Williamson — all of these have in one way or other made the academic part of my stay at Cornell exciting.

I have to mention at least some of the great number of friends whose company I have enormously enjoyed. Thanks to Stefan Wild and Pascal Tomecek, I became involved with playing ice hockey in the final three years at Cornell. It was fantastic! Being a part of the "CS Megahurtz" team, chasing the puck around the ice in shoes with sharp knives attached to them, participating in the Cornell intramurals and winning "silver" in my final year in Ithaca and being able to form a Slovak offensive trio with Miloš Hašan and Alexander Erdélyi every now and then was simply unforgettable. Yurii Zinchenko has been an incredible person and friend. He is responsible for teaching me my first table tennis skill — the top spin — and for the many hours I had henceforth spent in the ORIE PhD lounge with my office mates Chandrashekhar Nagarajan and Bikramjit Das and with Frans Schalekamp, drilling our spin.

The list of friends and stories could develop into a chapter of its own. Allow me therefore to resort only to listing the names of those who first come to mind at the moment: Damla Ahipasaoglu, Tuncay Alparslan, Aaron Archer, Dhruv Bhargava, Nikolai Bliznioukk, Broňa Brejová, Greg Bronevetsky, Tim Carnes, Millie Chu, Martin Dindoš, Nikolas Diener, Sam Ehrlichman, Ruženka Hostinská, Martina Gančárová, Souvik Ghosh, Minbok Kim, Dmitriy Levchenkov, Retsef Levi, Dennis Leventhal, Baldur Magnusson, Martin Pál, Chris Provan, Filip Radlinski, Ranjith Rajagopalan, Bharath Rangarajan, Parthanil Roy, Spyridon Schismenos, Deniz Sezer, Van Anh Truong, Tomáš Vinař, Tuohua Wu and Anke van Zuylen.

This research was partially supported by NSF through grants DMS-0209457

and DMS-0513337 and by ONR through grant N00014-02-0057. I would not have been able to complete the work without this support.

## TABLE OF CONTENTS

## LIST OF FIGURES

# Chapter 1

# Introduction

## 1.1 Optimization

Continuous optimization is the study of maximization or minimization of a continuous real-valued function $\varphi$ (*objective function*) over some set $\mathcal{Q}$ (*feasible region*). It is customary in the literature to focus on minimization problems since maximization can always be treated as the minimization of $-\varphi$. For simplicity, let us assume that $\mathcal{Q} \subseteq \mathbf{R}^n$. Our general optimization problem therefore takes on the following form:

$$\begin{aligned} \varphi^* \quad \leftarrow \quad & \text{minimize} \quad \varphi(x) \\ & \text{subject to} \quad x \in \mathcal{Q}. \end{aligned} \tag{OP}$$

A feasible point $x^*$ is *globally optimal* (a global minimizer) if $\varphi(x^*) \leq \varphi(x)$ for all $x \in \mathcal{Q}$. It is *locally optimal* (a local minimizer) if $\varphi(x^*) \leq \varphi(x)$ for all feasible points $x$ from some neighborhood of $x^*$. If $\varphi(x) \leq \varphi^* + \epsilon$ for some $x \in \mathcal{Q}$, then $x$ is an $\epsilon$-approximate minimizer in *absolute scale*. If $\varphi^* > 0$, then $x \in \mathcal{Q}$ is a $\delta$-approximate minimizer in *relative scale* if $\varphi(x) \leq (1+\delta)\varphi^*$.

Optimization problems can be naturally classified as follows:

- Unconstrained problems (if $\mathcal{Q} = \mathbf{R}^n$).
- Constrained problems (if $\mathcal{Q} \subsetneq \mathbf{R}^n$).
- Smooth problems (if $\varphi$ is smooth).
- Nonsmooth problems (if $\varphi$ is not smooth).

*Linear programming* is the case with the objective function $\varphi$ being linear and $\mathcal{Q}$ being a polyhedron. *Convex optimization* is the case with $\varphi$ and $\mathcal{Q}$ being convex. In this thesis we will deal predominantly with solving nonsmooth convex optimization problems in relative scale.

## 1.2 Complexity of optimization problems

As a rule, optimization problems with simpler functions and/or simple constraints are more tractable than more general problems. An intractable problem, loosely speaking, is one which requires enormous computational effort if we desire to solve either an instance of high dimension, or if we wish to obtain a solution (or approximate solution) of high accuracy, or both. It turns out that in some specific rigorous sense, *most* optimization problems are simply intractable. The following subsection will illustrate this on the problem class of minimizing a Lipschitz continuous function on the unit box.

### 1.2.1 Minimizing a Lipschitz function on the unit box with a zero-order oracle

Nesterov [19] gives the following example:

**Example 1.2.1.** Consider the class of $\gamma$-Lipschitz continuous functions, with respect to the $\ell_\infty$ norm, on the unit box in $\mathbf{R}^n$. That is, we assume that $|\varphi(x) - \varphi(y)| \leq \gamma \|x-y\|_\infty$ for all $x, y \in \mathcal{Q} := \{x \in \mathbf{R}^n \ : \ 0 \leq x^{(i)} \leq 1, \ i = 1, 2, \ldots, n\}$, for all functions $\varphi$ of this class. Our goal is to find a global minimizer on $\mathcal{Q}$, within absolute error $\epsilon$, of a function from this class.

What computational effort do we have to be prepared to spend to solve an

instance from this problem class? It turns out that the most straightforward approach, called the *uniform grid method*, is *optimal* under the *zero-order black-box* model. In this model we assume that the only information that a method can gather about the problem instance at hand comes from the answers of a zero-order oracle. That is, at every iteration we ask about a point $x \in \mathcal{Q}$ and the oracle answers $\varphi(x)$.

The uniform grid method proceeds as follows. We divide the feasible region into a fine uniform and then ask the oracle about each of the grid points, one-by-one. The output of the method is the best point found. It can be shown easily that this method requires at most

$$\left(\left\lfloor \frac{\gamma}{2\epsilon} \right\rfloor + 2\right)^n \tag{1.1}$$

calls of the oracle (i.e. iterations) to output an $\epsilon$-minimizer. Using the concept of a *resisting oracle*, it can be also proved that no less than

$$\left\lfloor \frac{\gamma}{2\epsilon} \right\rfloor^n \tag{1.2}$$

calls of the oracle can guarantee the desired accuracy (if $\epsilon < \frac{\gamma}{2}$) for every member function of this class, *whatever method we use*, as long as we get our information from a zero-order oracle. Notice that if $\frac{\gamma}{\epsilon}$ is large enough, at least some constant fraction of $n$, then the bound (1.1) is at most a constant multiple of (1.2). The uniform grid method can therefore be deemed to be optimal for the problem class considered.

Note that in spite of the optimality of this method, this problem class is computationally hopeless. The number of iterations grows so rapidly with increasing dimension and/or accuracy requirements that, in fact, a simple problem from this class, with parameters $\gamma = 2$ and $n = 10$, could require as much as 312.5 billion

| | |
|---|---|
| Lower bound $\left(\frac{\gamma}{2\epsilon}\right)^n$ | $10^{30}$ calls of the oracle |
| Arithmetic operations per iteration | $n = 10$ |
| Total complexity | $10^{31}$ arithmetic operations |
| TRIPS processor (expected in 2010) | $10^{12}$ arithmetic operations per second |
| Total time spent in seconds | $10^{19}$ seconds |
| One year | less than $3.2 \times 10^7$ seconds |
| Total time needed in years | **312.5 billion years** |

Figure 1.1: Optimization problems are generally intractable.

years to solve within $\epsilon = 0.001$ of the global minimum on a futuristic supercomputer[1] planned to be built no sooner than in 2010. This is more than 22 times the age of our universe! We have taken this example from [19] and boosted the level of dramatization a bit (see Figure 1.1).

Now imagine we want to solve the above problem with a smaller accuracy, say $\epsilon = 0.1$. Then we need to perform only $10^{11}$ arithmetic operations, which can be done in a tenth of a second using the TRIPS processor. Note that the time needed to solve the problem grows much more dramatically with the dimension.

The above example serves the purpose of illustrating several points:

- We cannot hope to find tractable methods if we deal with a prohibitively large class of optimization problems. It is therefore desirable to concentrate on well-defined narrow classes of problems with properties allowing for faster methods (for example, convexity).

[1] "IBM and the University of Texas at Austin plan to collaborate on building a processor capable of churning out more than 1 trillion calculations per second– faster than many of today's top supercomputers. A chip capable of performing 1 trillion operations, a tera-op, won't emerge from the project until 2010" (ZDNet News, August 27, 2003)

- Solving large-scale problems with high accuracy may be too much to ask for. There might be room for finding methods which work well in either high dimensions and low accuracies or vice versa.

- The oracle model may be too restrictive. Can we design better methods by using more information than the oracle can give us?

### 1.2.2 Complexity of convex optimization problems in the first-order black-box model

A first-order oracle outputs, apart from the value of the objective function at the point of interest, also some first-order information. In the case of a smooth convex function, this is the gradient, and in the case of a nonsmooth convex problem, a (convex) subgradient.

#### Smooth convex problems

Consider the problem class with smooth convex objective functions with Lipschitz continuous gradient with constant $\gamma$ and convex feasibility set $\mathcal{Q}$. Our goal is to find a $\epsilon$-approximate (in an absolute sense) global minimizer using a method adhering to the first-order black box model. It has been well known since the 80's that the lower complexity bound of this problem class is

$$O\left(\sqrt{\frac{\gamma}{\epsilon}}\right).$$

Optimal methods matching this bound have been developed in [18], see also [19].

**Nonsmooth convex problems**

In the case of nonsmooth convex problems, which is the focus of this thesis, the first methods proposed were the *subgradient methods*. They were studied intensively in the sixties and seventies of the twentieth century by a number of researchers, among them Y.M. Ermoliev, B.T. Polyak and N.Z. Shor. For a historical account see, for example, Shor's book [29].

A subgradient algorithm at every iteration takes a step in the direction of the negative subgradient (provided by the oracle). The size of the subgradient, unlike in the smooth setting where it points in a downhill direction, is not informative and cannot drive the algorithm. To see this, think of the function $\varphi(x) = \max\{-x, 1000x\}$ and consider taking a step in the direction of the negative subgradient of $\varphi$ at the current iterate, say $x = 0.001$. The subgradient is $g = 1000$ and its size has no relation with the distance to the minimizer — the origin. This simple example suggests that the goal of achieving a certain guaranteed decrease in the objective value at *every iteration* is out of the reach of subgradient methods. Instead, these schemes exploit the fact that the direction of the negative subgradient forms an acute (or at worst right) angle with the direction pointing from the current iterate towards a minimizer. To ensure convergence, the step lengths cannot drop too rapidly (they have to add up to infinity) and are usually chosen to decrease at the rate $1/\sqrt{k}$, where $k$ is the iteration counter. However, if one wants to run the method for a *fixed* number of steps, it turns out that it is theoretically optimal to choose steps of equal lengths.

Subgradient methods require

$$O\left(\frac{1}{\epsilon^2}\right) \tag{1.3}$$

iterations to converge to an approximate minimizer [9], [19]. It is known that a simple subgradient scheme is optimal for its problem class in the first-order black-box model, uniformly in the dimension of the problem [17] (the number of iterations does not depend on the dimension of the variables). In this sense, subgradient algorithms are likely to be useful in situations with huge dimensions and low accuracy needs.

### 1.2.3 An intrinsic problem of the black-box assumption

As recently pointed out by Nesterov [21], [20], [24], there is a certain paradox with the black box assumption for convex problems. If we want to be able to apply the subgradient method to a convex problem, we need to know that is *is* indeed convex. However, convexity is a very strong global property that is often verified *by inspection of the structure* of the problem in a process similar to verifying differentiability — there is a convex calculus. For example, the following operations preserve convexity:

- Maximum of any number of convex functions.
- Nonnegative linear (conic) combination of convex functions.
- Composition of a convex function with a linear function.
- Post-composition with an increasing convex function.

For a more exhaustive list we refer the reader to Part IV.2 of [13].

Since we can apply a convex method to a problem only if we have knowledge about its convexity and because, in turn, this knowledge comes from the structure of the problem, we actually know something about the problem class that we are

willingly forgoing. Is it the case that by strict adherence to the black-box concept we are not in the position to utilize this potential information to possibly improve on the lower complexity bound (1.3)? The answer to this question is positive. In the papers cited above Nesterov consider convex problems with objective function having an explicit structure and shows how to construct a *smooth uniform* $\epsilon$-*approximation* of this function with Lipschitz continuous gradient with a reasonably small Lipschitz constant of size $\gamma = O(\frac{1}{\epsilon})$. He then shows that after we apply an optimal smooth method to the smooth approximation, we can recover an $\epsilon$-approximate minimizer of the original nonsmooth convex problem in

$$O\left(\sqrt{\frac{\gamma}{\epsilon}}\right) = O\left(\sqrt{\frac{O(1/\epsilon)}{\epsilon}}\right) = O\left(\frac{1}{\epsilon}\right) \tag{1.4}$$

iterations of a first-order type. This is an improvement of one order of magnitude over the classical bound (1.3).

### 1.2.4 Structural optimization with second-order information

Let us note that unlike in the case of the first-order methods, the usefulness of structure was fully recognized in the theory of second-order methods already a long time ago in the seminal work on interior-point methods by Nesterov and Nemirovski [25]. For a more concise account we refer the reader to [27] and [19].

Since we do not develop any second-order methods in this thesis, we will only mention the complexity results and give a one-sentence outline of the underlying idea. The basic strategy is to transform the complexity of the convex objective function into the feasible set and then instead consider a linear objective function. The problem can then be equipped, at least theoretically, with a self-concordant

barrier function capturing its structure. The information in this barrier function is then utilized to drive the methods.

Since interior-point algorithms rely on second-order information, they are able to converge in much fewer iterations. Their theoretical complexity is

$$O\left(\sqrt{\nu}\ln\frac{1}{\varepsilon}\right),$$

where $\nu$ is a *parameter of the self-concordant barrier*, often representing the dimension of the problem. This dependence on the accuracy parameter is called linear convergence. These methods tend to converge faster in practice than in theory, in terms of their dependence on the goal accuracy, which makes them very attractive for applications where small error is crucial. One of the disadvantages is the increased computational cost per iteration. It is generally one order of magnitude higher, in the dimension of the problem, than in the case of first-order type methods. In this sense, first-order methods, and especially those with the improved guarantee (1.4), are very attractive for large-scale applications where there is need only for medium accuracy, perhaps $\epsilon \in [10^{-1}, 10^{-4}]$.

## 1.3 A brief overview of the thesis

In this thesis we develop first-order algorithms for solving large-scale nonsmooth convex problems in *relative scale*, utilizing their structure. We develop methods converging in $O(1/\delta^2)$ or $O(1/\delta)$ iterations — $\delta$ corresponds to the desired relative accuracy. While we have not improved further the dependence on $\delta$, some of our methods are less sensitive to other parameters which at this point remain hidden by the $O$-notation.

**Synopsis of Chapter 2**

In this chapter we improve the algorithms of Nesterov [23], [22] for solving unconstrained nonsmooth convex minimization problems within a prescribed error $\delta$ in relative scale.

We develop algorithms based on a subgradient subroutine and on Nesterov's smoothing technique [21]. This class of algorithms depends on the availability of an *ellipsoidal rounding* of the subdifferential of the objective function at the origin. Our main improvement is based on a simple bisection idea. We also show how to modify these methods, at no or only negligible cost in the theoretical complexity, to allow for perhaps desirable "nonrestarting" behavior. In the final section we attempt to combine the rounding and optimization phases of the algorithms based on the subgradient subroutine.

**Synopsis of Chapter 3**

Our main goal in this part of the thesis is to find the intersection point of a centrally symmetric convex set $\mathcal{Q}$ and a line passing through the origin.

This problem can be treated with the methods of the previous chapter, as will become apparent from the discussion. The proposed approach involves constructing a sequence of ellipsoids inscribed in $\mathcal{Q}$, greedily "converging" towards the intersection points. The more efficient of our algorithms can be viewed as nontrivial modifications of Khachiyan's ellipsoidal rounding algorithm to our problem. While the generic structure of an iteration is identical to that of Khachiyan, we employ a different strategy for choosing the update vector and work with a different line search objective function. One aspect of our contribution is therefore showing that modifications of this type can produce meaningful sequences of ellip-

soids. Our algorithms can also be interpreted as performing Frank-Wolfe steps on the unit simplex.

At the same time we consider several other closely related problems. We show that our methods simultaneously solve all of them in $O(1/\delta)$ iterations of a first-order type. One of these problems is the problem of minimizing the maximum of absolute values of the linear functionals over a hyperplane. Another is the problem of finding the smallest $\ell_1$ norm solution of a full-rank underdetermined linear system. We also consider maximization of a linear functional over a centrally symmetric polytope, the *polar* of $\mathcal{Q}$.

Our analysis is similar to that of [15] and [33]. For related work we refer the reader also to [16], [32] and [1].

## 1.4 The setting and some notation

The general setting of this thesis is a finite-dimensional real vector space $\mathbf{E}$. We follow a coordinate-free approach by not fixing any basis. Since we also do not wish to assume the existence of a pre-existing geometry (inner product), we instead characterize linear functionals on $\mathbf{E}$ in the functional-analytic spirit through the use of the dual space $\mathbf{E}^*$ — the space of all linear functionals on $\mathbf{E}$. By $\langle g, x\rangle$ we mean the action of the linear functional $g \in \mathbf{E}^*$ on $x \in \mathbf{E}$. By $n$ we denote the dimension of $\mathbf{E}$ (and hence of $\mathbf{E}^*$).

A linear operator $U\colon \mathbf{E} \to \mathbf{E}^*$ is *positive semidefinite* (we write $U \succeq 0$) if $\langle Ux, x\rangle \geq 0$ for all $x \in \mathbf{E}$. If the inequality is strict for $x \neq 0$, it is *positive definite* ($U \succ 0$). It is *self-adjoint* if $\langle Ux_2, x_1\rangle = \langle Ux_1, x_2\rangle$ for all $x_1, x_2 \in \mathbf{E}$.

By $gg^*\colon \mathbf{E} \to \mathbf{R}$, for $g \in \mathbf{E}^*$, we mean the (rank-one) operator defined by $gg^*x := \langle g, x\rangle g$.

**Coordinates.** Sometimes it is convenient to identify both $\mathbf{E}$ and $\mathbf{E}^*$ with $\mathbf{R}^n$ and to treat the vectors of these spaces as column vectors. A linear operator from $\mathbf{E}$ to $\mathbf{E}^*$ is then treated as a $n \times n$ real matrix, and $\langle \cdot, \cdot \rangle$ means the standard inner product in $\mathbf{R}^n$.

Let us briefly mention what we mean by this. We fix a pair of *dual bases* in $\mathbf{E}$ (say $x'_1, \ldots, x'_n$) and in $\mathbf{E}^*$ (say $g'_1, \ldots, g'_n$). That is, $\langle g'_i, x'_j \rangle$ is equal to 1 if $i = j$ and 0 otherwise. If $\hat{x}$ (resp. $\hat{g}$) denotes the column vector of coordinates of vector $x$ (resp. $\hat{g}$) relative to basis $\{x'_i\}$ (resp. $\{g'_i\}$), then

$$\langle g, x \rangle = \langle \sum \hat{g}_i g'_i, \sum \hat{x}_i x'_i \rangle = \sum \hat{g}_i \hat{x}_i = \langle \hat{g}, \hat{x} \rangle,$$

where the last expression now denotes the standard inner product in $\mathbf{R}^n$. Hence by identifying $x$ with $\hat{x}$ and $g$ with $\hat{g}$, the expression $\langle g, x \rangle$ takes on the form of a standard inner product of two vectors in $\mathbf{R}^n$.

The use of coordinates in $\mathbf{E}$ and $\mathbf{E}^*$ follows this general rule: the theorems are stated coordinate-free while some proofs may require fixing a pair of bases in the way described above.

**A pair of primal spaces and their conjugates.** In Section 2.3 of Chapter 2 we work with a pair of finite-dimensional real vector spaces $\mathbf{E}_1$ and $\mathbf{E}_2$ (possibly of different dimension) and their duals $\mathbf{E}_1^*$ and $\mathbf{E}_2^*$. If $A \colon \mathbf{E}_1 \to \mathbf{E}_2^*$ is linear then its adjoint is the linear operator $A^* \colon \mathbf{E}_2 \to \mathbf{E}_1^*$ defined via

$$\langle Ax, y \rangle = \langle A^* y, x \rangle \qquad (x \in \mathbf{E}_1,\ y \in \mathbf{E}_2).$$

**More notation.** Lower-case Greek letters such as $\alpha, \beta, \gamma, \delta, \kappa, \tau$ and $\epsilon$ denote scalars or real-valued functions; lower-case Roman letters (mainly from the beginning of the alphabet) such as $a, b, c, d, g$ are elements of $\mathbf{E}^*$, while $x, y, z$, all possibly

with subscripts, are elements of $\mathbf{E}$. While this is the general rule, we allow for local inconsistencies when it seems more natural to choose different notation.

The $m$-dimensional unit simplex is denoted by $\Delta_m := \{w \in \mathbf{R}^m_+ \,:\, \sum_i w_i = 1\}$. For a vector $w \in \mathbf{R}^m$ we will use the notation $|w| = (|w_1|, \ldots, |w_m|)^T$, $\|w\|_1 = \sum_i |w_i|$ and $\|w\|_\infty = \max_i |w_i|$. Section 2.3 of Chapter 2 is an exception and by $\|\cdot\|_1$ and $\|\cdot\|_2$ we mean two fixed norms defined on the spaces $\mathbf{E}_1$ and $\mathbf{E}_2$, respectively. By $\mathrm{sign}(\cdot)$ we denote the sign function on the reals.

# Chapter 2

# Improved algorithms for unconstrained nonsmooth convex minimization in relative scale

## 2.1 Introduction

The theory of modern convex optimization almost uniformly assumes *boundedness* of the feasible set. This assumption is usually artificially enforced even for naturally unconstrained problems via the so-called "big M" method. A clear advantage of dealing with bounded sets is the availability of a *scale* in which one can measure the absolute accuracy of a solution. However, there always seems to be the issue of keeping a balance between the size of the artificially imposed bounds (large feasible sets tend to slow algorithms down) and the possibility of exclusion of minimizers from the feasible sets in so doing. Since there is no natural absolute scale for measuring the solutions of an unconstrained problem, it seems to be reasonable to be looking for solutions that are approximately optimal in *relative scale*. Results of this type, however, are very rare in the convex optimization literature. This contrasts with the literature on combinatorial optimization where approximation algorithms are studied extensively.

Nesterov [23] recently showed that the above obstacles can be overcome for the problem class of minimizing a convex *homogeneous* function over an affine subspace. The essence of his approach involves the computation of an *ellipsoidal rounding* of the subdifferential of the objective function (at the origin) by uti-

lizing the knowledge about the *structure* of the problem. This family of problems encompasses essentially all unconstrained convex minimization problems via a dimension-lifting procedure. However, certain assumptions about the ellipsoidal rounding effectively limit the class of problems that can be treated.

In this chapter we improve the algorithms of Nesterov [23], [22] for solving unconstrained nonsmooth convex minimization problems within a prescribed error $\delta$ in relative scale. Our central idea was independently used by Chudak and Eleutério [6] to obtain the same theoretical improvement in the context of concrete combinatorial applications. The methods we propose converge in $O(1/\delta^2)$ or $O(1/\delta)$ iterations of a first-order type.

The text is organized as follows. In the introductory section we formally describe the problem, briefly describe the dimension-lifting procedure and prove essential inequalities coming from an ellipsoidal rounding of the subdifferential of the objective function evaluated at the origin. In Section 2.2 we develop algorithms based on a subgradient subroutine. We first describe Nesterov's results and then improve them by incorporating a simple bisection idea. We also show how to modify our methods, at no or only negligible cost in the theoretical complexity, to allow for a perhaps desirable "nonrestarting" behavior. Section 2.3 is devoted to the development of improved algorithms based on Nesterov's smoothing technique. The methods of this part are considerably faster then those based on the subgradient routine. After this, in Section 2.4, we apply our results to several specific choices of the objective function. One of those applications, for example, comes from game theory. The final section contains a collection of results related to the idea of combining the rounding and optimization phases of the algorithms from Section 2.2.

### 2.1.1 Constrained sublinear minimization

The central problem of this chapter is

$$\boxed{\varphi^* := \min_{x \in \mathcal{L}} \varphi(x),} \tag{P}$$

where $\mathcal{L}$ is an affine subspace of a finite-dimensional real vector space $\mathbf{E}$ not containing the origin and $\varphi\colon \mathbf{E} \to R$ is a *sublinear* function — convex and (positively) homogeneous of degree one. The last property means that the function is linear on every ray emanating from the origin: $\varphi(\tau x) = \tau\varphi(x)$ for all $\tau \geq 0$ and $x \in \mathbf{E}$. Note that convexity and homogeneity imply subadditivity. By $\mathbf{E}^*$ we denote the dual of $\mathbf{E}$, the space of linear functionals on $\mathbf{E}$. Let us define $n := \dim \mathbf{E} = \dim \mathbf{E}^*$.

We will further make the assumption that the zero vector lies in the interior of the (convex) subdifferential[1] of $\varphi$ evaluated at the origin:

$$0 \in \operatorname{int} \partial\varphi(0). \tag{2.1}$$

Given the properties of $\varphi$, condition (2.1) essentially amounts to requiring that the origin is the *unique* global minimizer of $\varphi$. The above assumptions imply that $\partial\varphi(0)$ is a full-dimensional compact and convex subset of $\mathbf{E}^*$ and that we can write[2]

$$\varphi(x) = \max\{\langle g, x\rangle \ : \ g \in \partial\varphi(0).\} \tag{2.2}$$

[1]For $x \in \mathbf{E}$ the set $\partial\varphi(x)$, called the (convex) *subdifferential* of $\varphi$ at $x$, is the subset of $\mathbf{E}^*$ defined by

$$g \in \partial\varphi(x) \quad \Leftrightarrow \quad \varphi(y) \geq \varphi(x) + \langle g, y - x\rangle, \quad \forall y \in \mathbf{E}.$$

Elements of $\partial\varphi(x)$ are called *subgradients*.

[2]There is a one-to-one correspondence between finite sublinear functions and nonempty compact convex sets via the relation $\varphi(x) = \max\{\langle g, x\rangle \mid g \in \mathcal{G}\}$ (this is the *support function* of $\mathcal{G}$). It then follows from the definition of the subdifferential that $\mathcal{G} = \partial\varphi(0)$. We refer the reader to Rockafellar's book [28], a classic in the convex analysis literature. An detailed account of the properties of sublinear functions and subdifferentials of convex functions can be found in Chapters IV and V of Hiriart-Urruty and Lemaréchal [13]. For a more compact and up-to-date treatment see Borwein and Lewis [5] (Corollary 4.2.3).

That is, $\varphi$ is the *support function* of its subdifferential at the origin. For geometric understanding of the situation implied by the assumptions it is helpful to note that the epigraph of $\varphi$ is a convex cone in $\mathbf{E} \times \mathbf{R}_+$ whose only intersection with $\mathbf{E} \times \{0\}$ is the origin (see Figure 2.1.1).

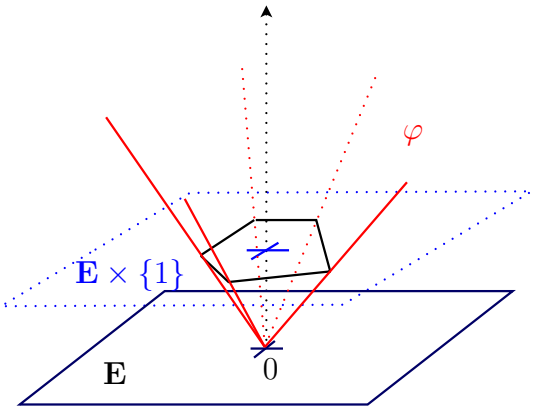



Figure 2.1: Epigraph of a sublinear function.

### Approximate solution

Our aim is to find an approximate solution of $(P)$, within relative error $\delta$. Let us formalize this concept:

**Definition 2.1.1.** Point $x \in \mathcal{L}$ is a $\delta$-*approximate solution* to $(P)$ if

$$\varphi(x) \leq (1 + \delta)\varphi^*.$$

In proving theorems we will often use the following equivalent characterization:

$$\varphi(x) - \varphi^* \leq \frac{\delta}{1 + \delta}\varphi(x).$$

**Treating unconstrained convex minimization**

We have claimed in the introduction that the general unconstrained convex minimization problem can be reformulated as a constrained sublinear problem. Let us briefly describe the construction. If $\phi\colon \mathbf{E} \to \mathbf{R}$ is a convex function, its *perspective* is the function $\varphi\colon \mathbf{E} \times \mathbf{R}_{++} \to \mathbf{R}$ defined by

$$\varphi(x) := \varphi(y, \tau) = \tau\phi\left(\frac{y}{\tau}\right).$$

This function is clearly linear on every feasible ray leaving from the origin. In fact, it can be shown that $\varphi$ is convex on its domain (see, for example, Proposition 2.2.1 in [13]). It is not in general possible to extend $\varphi$ onto the entire space $\mathbf{E} \times \mathbf{R}$ if we want to preserve both convexity and finiteness. However, there are at least some important classes of functions for which this can be done. Consider the following example:

**Example 2.1.2** (Example 1, [23])**.** Let

$$\phi(y) = \max\{|\langle a_i, y\rangle + b^{(i)}| \ : \ i = 1, 2, \ldots, m\}$$

with $y \in \mathbf{E}$, $a_1, \ldots, a_m \in \mathbf{E}^*$ and $b \in \mathbf{R}^m$. If we let $x = (y, \tau)$ and $a_i' = (a_i, b^{(i)})$ for $i = 1, 2, \ldots, m$ then for $\tau > 0$ we get

$$\begin{aligned}\varphi(x) = \varphi(y, \tau) = \tau\phi\left(\frac{y}{\tau}\right) &= \tau \max_{1\le i\le m} |\langle a_i, y/\tau\rangle + b^{(i)}| \\ &= \max_{1\le i\le m} |\langle a_i, y\rangle + b^{(i)}\tau| \\ &= \max_{1\le i\le m} |\langle a_i', x\rangle|,\end{aligned}$$

where the last equality defines a new inner product on $\mathbf{E} \times \mathbf{R}$. Clearly, $\varphi$ can be extended to a sublinear function defined on the entire space. Assumption (2.1) will be satisfied if $0 \in \operatorname{int} \partial\varphi(0) = \operatorname{conv}\{\pm a_i' \ : \ i = 1, 2, \ldots, m\}$.

### 2.1.2 Ellipsoidal rounding and key inequalities

#### John ellipsoids

As a pre-processing phase, we first find a positive definite operator $U\colon \mathbf{E} \to \mathbf{E}^*$ giving rise to a pair of central ellipsoids in $\mathbf{E}^*$, one being contained in $\partial\varphi(0)$ and the other containing it. This can be done, for example, using Khachiyan's algorithm [15] which we describe in Subsection 2.5.1. Until then we will simply assume the availability of radii $0 < \gamma_0 \leq \gamma_1$ such that

$$\mathcal{B}(U, \gamma_0) \subseteq \partial\varphi(0) \subseteq \mathcal{B}(U, \gamma_1), \tag{2.3}$$

where

$$\mathcal{B}(U, \gamma) := \{g \in \mathbf{E}^* \ : \ \sqrt{\langle g, U^{-1}g\rangle} \leq \gamma\}$$

defines an ellipsoid in $\mathbf{E}^*$ with radius $\gamma$.

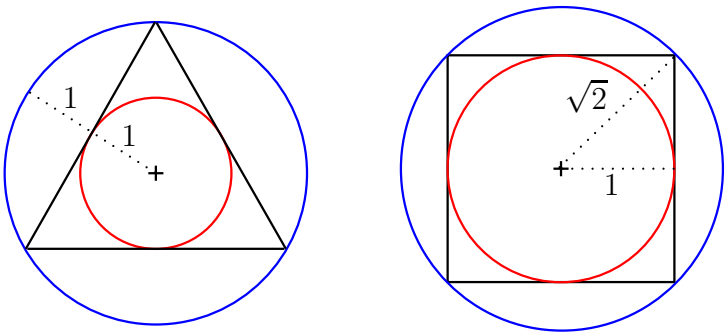
The theoretical guarantees of the algorithms presented in this chapter depend on the quantity $\alpha := \frac{\gamma_0}{\gamma_1}$, which characterizes the quality of the ellipsoidal rounding (2.3). It is clearly always the case that $0 < \alpha \leq 1$, with bigger $\alpha$ corresponding to a tighter rounding and, as we will see, faster algorithms. The following result, a celebrated theorem of John [14], gives lower bounds on the quality of rounding admitted by full-dimensional convex sets:

**Proposition 2.1.3** (John [14])**.** *Any convex body $\mathcal{Q} \subset \mathbf{E}^*$ admits a rounding by concentric ellipsoids with $\frac{1}{\alpha} \leq \dim \mathbf{E}^*$. If $\mathcal{Q}$ is centrally symmetric, then there exists a rounding with $\frac{1}{\alpha} \leq \sqrt{\dim \mathbf{E}^*}$.*

**Example 2.1.4.** To see that the above result gives tight bounds, consider the following example (see Figure 2.2 for a conveniently scaled picture for $n = 2$). The rounding obtained by the inscribed and circumscribed balls of

1. a regular $n$-simplex has quality $\frac{1}{\alpha} = n$,

2. the $n$-cube (a centrally symmetric body) has quality $\frac{1}{\alpha} = \sqrt{n}$.




a regular 2-simplex 2-cube

Figure 2.2: Tight examples of rounding convex sets by ellipsoids.

### Geometry induced by rounding

The rounding operator $U$ defines an inner product on $\mathbf{E}$ via $\langle x, y\rangle_U := \langle Ux, y\rangle$, which in turn induces the norm $\|x\|_U := \sqrt{\langle x, x\rangle_U}$. The dual space $\mathbf{E}^*$ can be equipped with the dual norm $\|g\|_U^* := \sqrt{\langle g, U^{-1}g\rangle}$. Notice that these norms are themselves sublinear functions and as such admit a representation similar to (2.2):

$$\|x\|_U = \max\{\langle g, x\rangle \ : \ \|g\|_U^* \leq 1\} \tag{2.4}$$

with $\partial\|\cdot\|_U(0) = \{g \in \mathbf{E}^* \ : \ \|g\|_U^* \leq 1\}$ and

$$\|g\|_U^* = \max\{\langle g, x\rangle \ : \ \|x\|_U \leq 1\} \tag{2.5}$$

with $\partial\|\cdot\|_U^*(0) = \{x \in \mathbf{E} \ : \ \|x\|_U \leq 1\}$. Also observe that the first and last sets in (2.3) are balls in $\mathbf{E}^*$, with respect to the dual norm induced by $U$, of radii $\gamma_0$ and $\gamma_1$, respectively.

**Subgradients in the primal space**

By defining

$$\partial_U \varphi(x) := \{h \in \mathbf{E} \ : \ \varphi(y) \geq \varphi(x) + \langle h, x \rangle_U, \quad \forall y \in \mathbf{E}\},$$

the subgradients of $\varphi$ can be thought of as being elements of $\mathbf{E}$ as opposed to elements of $\mathbf{E}^*$. This will enable us to talk about taking steps in $\mathbf{E}$ in the "direction" of a negative subgradient. Note that there is a one-to-one correspondence linking the two concepts:

$$\partial_U \varphi(x) = U^{-1}[\partial \varphi(x)]. \tag{2.6}$$

**Inequalities**

In view of (2.2) and (2.4), taking the maximum of the linear functional $\langle \cdot, x \rangle$ over the sets in (2.3) gives

$$\gamma_0 \|x\|_U \leq \varphi(x) \leq \gamma_1 \|x\|_U \quad \text{for all} \quad x \in \mathbf{E}, \tag{2.7}$$

which together with subadditivity of $\varphi$ implies that $\varphi$ is $\gamma_1$-Lipschitz:

$$\varphi(x + h) \leq \varphi(x) + \varphi(h) \leq \varphi(x) + \gamma_1 \|h\|_U.$$

From now on let us adopt the following notation. By $x^*$ we denote an arbitrary optimal solution of $(P)$ and by $x_0$ we denote the minimum norm element of the feasible region – the projection of the origin onto $\mathcal{L}$. From (2.7) we then obtain

$$\alpha \varphi(x_0) \leq \gamma_0 \|x_0\|_U \leq \gamma_0 \|x^*\|_U \leq \varphi^* \leq \varphi(x_0) \leq \gamma_1 \|x_0\|_U. \tag{2.8}$$

Dividing by $\gamma_0$ we get

$$\frac{\varphi(x_0)}{\gamma_1} \leq \|x_0\|_U \leq \|x^*\|_U \leq \frac{\varphi^*}{\gamma_0} \leq \frac{\varphi(x_0)}{\gamma_0}. \tag{2.9}$$

Because $\|x^* - x_0\|_U = \sqrt{\|x^*\|_U^2 - \|x_0\|_U^2}$ by the Pythagoras theorem and since $x_0 \neq 0$ due to the assumption that $\mathcal{L}$ does not pass through the origin, we also obtain

$$\|x^* - x_0\|_U < \|x^*\|_U \leq \frac{\varphi^*}{\gamma_0} \leq \frac{\varphi(x_0)}{\gamma_0}. \tag{2.10}$$

## 2.2 Algorithms based on a subgradient subroutine

Subgradient algorithms were studied intensively in the sixties and seventies of the twentieth century by a number of researchers, among them Y.M. Ermoliev, B.T. Polyak and N.Z. Shor. See, for example, Shor's book [29] and Goffin's paper on convergence rates [9]. For our purposes we will only need a result about the performance of a standard constant step-length subgradient algorithm applied to a convex Lipschitz function. This algorithm, together with a simple proof, can be found, for example, in Section 3.2.3 of Nesterov's book [19].

In the first subsection we start by briefly discussing the constant step-length subgradient method and its performance guarantee.

### 2.2.1 A constant step-length subgradient algorithm

The subgradient algorithm we are going to describe works in a more general setting than that of problem $(P)$. For the sake of this subsection only, consider the problem of minimizing a convex Lipschitz continuous function $\varphi\colon \mathbf{E} \to \mathbf{R}$ with Lipschitz constant $\gamma$ over a *simple* closed convex set $\mathcal{Q}_1$:

$$\boxed{\varphi^* := \min\{\varphi(x) \ : \ x \in \mathcal{Q}_1\}.} \tag{$P_{subgrad}$}$$

By simple set we mean one allowing for easy computation of projections onto it. In this setting $\mathbf{E}$ is assumed to be equipped with an inner product. Problem $(P)$

is a special case of $(P_{subgrad})$ with

- $\varphi$ having additional properties,
- $\gamma = \gamma_1$ and $\mathcal{Q}_1 = \mathcal{L}$, and
- $\mathbf{E}$ made Euclidean by the introduction of the inner product induced by $U$.

**Proposition 2.2.1.** *If $\|x^* - x_0\| \leq R$ for some $x_0 \in \mathbf{E}$, minimizer $x^*$ of $(P_{subgrad})$ and $R > 0$, then the output*

$$x = \mathbf{Subgrad}(\varphi, \mathcal{Q}_1, x_0, R, N)$$

*of Algorithm 1 run on an instance of problem $(P_{subgrad})$ satisfies:*

$$\varphi(x) - \varphi^* \leq \frac{\gamma R}{\sqrt{N+1}}. \tag{2.11}$$

*Proof.* Follows directly from Theorem 3.2.2 in [19]. □

**Algorithm 1 (Subgrad)** Constant step-length subgradient scheme

1: **Input:** $\varphi, \mathcal{Q}_1, x_0, R, N$;
2: $\kappa = R/\sqrt{N+1}$;
3: **for** $k = 0$ **to** $N - 1$
4: pick $g \in \partial\varphi(x_k)$; **if** $g = 0$ **then** $x_k$ is optimal and **exit**;
5: $x_{k+1} = \mathrm{proj}_{\mathcal{Q}_1}\left(x_k - \kappa \frac{g}{\|g\|}\right)$;
6: **end for**
7: **Output:** $x_k$ with best objective value

**Remark 2.2.2.** *For Proposition 2.2.1 it suffices to require that $\varphi$ be Lipschitz on the ball around $x^*$ with radius $R$.*

### 2.2.2 Basic algorithmic ideas

As the previous subsection indicates, the basic idea for solving $(P)$ will be that of using the subgradient method (Algorithm 1). The main issue with this algorithm, apart from the fact that it is slow (it requires $O(1/\epsilon^2)$ to output an $\epsilon$-optimal solution in the additive sense), is the need to supply an initial point $x_0$ and bound $R$ satisfying $\|x^* - x_0\| \leq R$.

The particular choice of $x_0$ as the projection of the origin onto the feasible set of $(P)$ makes sense from at least two reasons. First, notice that if the ellipsoidal rounding of $\partial\varphi(0)$ is perfectly tight ($\alpha = 1$), then by (2.7) we have $\varphi(x) \equiv \|x\|_U$ and therefore $x_0$ *is* the optimal solution of $(P)$. In fact, notice that (2.9) implies

$$\varphi(x_0) \leq \frac{\varphi^*}{\alpha}, \tag{2.12}$$

and hence $x_0$ is a $(\frac{1}{\alpha} - 1)$-approximate solution of $(P)$. The better the rounding, the better the guarantee. Second, (2.10) gives us the readily available upper bound $R = \varphi(x_0)/\gamma_0$. Of course, $\varphi^*/\gamma_0$ would be better, but we do not know it.

#### Good but unavailable upper bound

Let us formally apply Algorithm 1 to $(P)$ with $R = \varphi^*/\gamma_0$. To achieve the required relative accuracy, it then suffices to run it for $N = \lfloor \alpha^{-2}\delta^{-2} \rfloor$ iterations because by Proposition 2.2.1

$$\varphi(x) - \varphi^* \leq \frac{\gamma_1 R}{\sqrt{N+1}} \leq \frac{\varphi^*}{\alpha\sqrt{\frac{1}{\alpha^2\delta^2}}} = \delta\varphi^*.$$

#### Available but bad upper bound

Since the previous upper bound is unknown, let us use the worse (but available) bound $R = \varphi(x_0)/\gamma_0$. To guarantee a solution within relative error $\delta$, we need to

use $N = \lfloor \alpha^{-4}\delta^{-2} \rfloor$ iterations. The argument is exactly the same as in the case above except we start by replacing $\varphi(x_0)$ with $\varphi^*/\alpha$ in view of (2.12).

### Iteratively updated upper bound

To move towards the better of the two extremes, Nesterov [23] proposed a scheme (Algorithm 2) which uses the subgradient method as a subroutine and which iteratively decreases the known upper bound. His algorithm starts by running the subgradient method for $O(\alpha^{-2}\delta^{-2})$ iterations with the available upper bound $\varphi(x_0)/\gamma_0$. In case the subgradient subroutine is doing well and manages to decrease the objective value by a constant fraction, then the previously available upper bound also decreases by the same fraction. This improved bound is then used to run the next subgradient subroutine, again starting from $x_0$.

**Algorithm 2 (SubSearch)** Subgradient search scheme.

**Input:** $\varphi, \mathcal{L}, x_0, \gamma_0, \gamma_1, \beta > 0, \delta$;

$\hat{x}_0 = x_0,\ \alpha = \gamma_0/\gamma_1,\ c = e^{\beta},\ k = 1$;

$N = \left\lfloor \frac{c^2}{\alpha^2}\left(1 + \frac{1}{\delta}\right)^2 \right\rfloor$;

$\hat{x}_k = \mathbf{Subgrad}(\varphi, \mathcal{L}, x_0, \varphi(\hat{x}_{k-1})/\gamma_0, N)$;

**while** $\varphi(\hat{x}_k) < \frac{1}{c}\varphi(\hat{x}_{k-1})$ **do**

    $k = k + 1$;

    $\hat{x}_k = \mathbf{Subgrad}(\varphi, \mathcal{L}, x_0, \varphi(\hat{x}_{k-1})/\gamma_0, N)$;

**end while**

**Output:** $\hat{x}_k$

The performance of Algorithm 2 is substantially better than the naive one-time application of the subgradient method with the bad but available upper bound. Of course, it underperforms the one-time application of the subgradient method

with the good but unknown upper bound – by a factor of $O(\ln \frac{1}{\alpha})$.

**Proposition 2.2.3** (Nesterov [23], Theorem 3)**.** *Algorithm 2 returns a $\delta$-approximate solution of $(P)$ and takes at most*

$$\frac{e^{2\beta}}{\alpha^2}\left(1+\frac{1}{\delta}\right)^2\left(1+\frac{1}{\beta}\ln\frac{1}{\alpha}\right)$$

*steps of the subgradient method. If $\beta$ is chosen to be a constant, then the number of steps is*

$$O\left(\frac{1}{\alpha^2\delta^2}\ln\frac{1}{\alpha}\right). \tag{2.13}$$

*Proof.* Assume that the algorithm stops at iteration $k$, failing to satisfy the while clause at Step 5. In view of (2.8) we have

$$\alpha\varphi(x_0) \le \varphi^* \le \varphi(\hat{x}_{k-1}) < \left(\frac{1}{c}\right)^{k-1}\varphi(x_0),$$

and by comparing the first and the last term in this chain of inequalities we conclude that the number of calls of the subgradient subroutine is at most $1+\frac{1}{\beta}\ln\frac{1}{\alpha}$. The bound on the number of lower level steps is obtained by multiplying this by $N$ from Step 3 of the algorithm. It remains to show that the output is as specified. Indeed, using the termination rule from Step 5 and applying Proposition 2.2.1 to the last call of the subgradient subroutine we get

$$\varphi(\hat{x}_k)-\varphi^* \le \frac{\gamma_1\frac{\varphi(\hat{x}_{k-1})}{\gamma_0}}{\sqrt{N+1}} \le \frac{\frac{e^\beta}{\alpha}\varphi(\hat{x}_k)}{\sqrt{N+1}} \le \frac{\delta}{1+\delta}\varphi(\hat{x}_k).$$

□

### 2.2.3 Bisection improvement

Each outer iteration of Algorithm 2, possibly except the last one, produces a *guaranteed* upper bound on the distance of $x_0$ from the set of minimizers of $(P)$ —

better by a constant factor than the one available before. Loosely speaking, we will show that by allowing for *guesswork* it is possible to get a theoretical and practical improvement in the performance of this algorithm (the same improvement was independently obtained by Chudak and Eleutério [6] in the context of combinatorial applications). The key observation is formulated in the following lemma.

**Lemma 2.2.4.** *If $\varphi^*/\gamma_0 \leq R$ and $N = \lfloor \alpha^{-2}\beta^{-2} \rfloor$ for some $\beta > 0$, then*

$$x = \textbf{Subgrad}(\varphi, \mathcal{L}, x_0, R, N)$$

*satisfies*

$$\frac{\varphi(x)}{\gamma_0} \leq (1+\beta)R.$$

*Proof.* By Proposition 2.2.1 we have $\varphi(x) - \varphi^* \leq \gamma_1 R/\sqrt{N+1} \leq \gamma_0 \beta R$ and hence

$$\frac{\varphi(x)}{\gamma_0} \leq \frac{\varphi^*}{\gamma_0} + \beta R \leq R(1+\beta).$$

□

Lemma 2.2.4 essentially states that for *any* positive $R$ we can, at the cost of $O(\alpha^{-2}\beta^{-2})$ iterations of the subgradient method (Algorithm 1), either get a certificate that $\varphi^*/\gamma_0 \leq (1+\beta)R$ or that $R \leq \varphi^*/\gamma_0$. In any case we either get a *new* upper or lower bound on $\varphi^*/\gamma_0$. The *initial* lower and upper bounds come from (2.9): if we set $L_0 := \|x_0\|_U$ and $R_0 := \varphi(x_0)/\gamma_0$ then

$$\frac{\varphi(x_0)}{\gamma_1} \leq L_0 \leq \frac{\varphi^*}{\gamma_0} \leq R_0,$$

with $q_0 := R_0/L_0 \leq \frac{1}{\alpha}$. Assuming $(1+\beta)R \leq R_0$, the new lower and upper bounds are either $(L_1, R_1) = (L_0, (1+\beta)R)$, or $(L_1, R_1) = (R, R_0)$, depending on the outcome of the procedure suggested in Lemma 2.2.4 (see Figure 2.3). This bisection step is then repeated until the ratio $q_k := R_k/L_k$ gets down to a sufficiently small

value. It turns out that it is efficient to choose $\beta = \theta(1)$ and bisect only until $q_k$ decreases down to a constant value and then "finish the job" by taking $O(\alpha^{-2}\delta^{-2})$ additional subgradient steps, much in the way as we have seen with the "good but unavailable" upper bound.

The following lemma states how much of improvement in $q_k$ can be obtained by a single bisection step.

**Lemma 2.2.5.** *Assume $L_{k-1}$ and $R_{k-1}$ are lower and upper bounds on $\varphi^*/\gamma_0$, respectively, with $q_{k-1} > 1 + \beta$, and let*

$$R := \sqrt{\frac{L_{k-1}R_{k-1}}{1+\beta}}.$$

*If we run the subgradient method as indicated in Lemma 2.2.4 and if $L_k$ and $R_k$ are the new bounds, then*

$$q_k \leq \sqrt{1+\beta}\sqrt{q_{k-1}}. \tag{2.14}$$

*Proof.* First notice that the assumption $q_{k-1} > 1 + \beta$ implies that $L_{k-1} < R < (1+\beta)R < R_{k-1}$. Recall that we either have $(L_k, R_k) = (L_{k-1}, (1+\beta)R)$ or $(L_k, R_k) = (R, R_{k-1})$ and observe that $R$ is chosen so that the value of $q_k$ is the same under both eventualities:

$$\frac{(1+\beta)R}{L_{k-1}} = \frac{R_{k-1}}{R}.$$

Putting these observations together,

$$q_k = \frac{R_{k-1}}{R} = \sqrt{1+\beta}\sqrt{q_{k-1}}.$$

□

The ideas outlined above lead to Algorithm 3 whose performance is analyzed in Theorem 2.2.6.

**Algorithm 3 (SubBis)** Subgradient bisection scheme.

**Input:** $\varphi, \mathcal{L}, x_0, \gamma_0, \gamma_1, \beta, \delta$;

$k = 0,\ \hat{x}_0 = x_0,\ L_0 = \|x_0\|_U,\ R_0 = \varphi(x_0)/\gamma_0$;

$\alpha = \gamma_0/\gamma_1,\ c = 2(1+\beta),\ N = \left\lfloor \frac{1}{\alpha^2\beta^2} \right\rfloor$;

**while** $R_k/L_k > c$ **do**

  $k = k+1,\ R = \sqrt{\frac{L_{k-1}R_{k-1}}{1+\beta}},\ x = \textbf{Subgrad}(\varphi, \mathcal{L}, x_0, R, N)$;

  **if** $\varphi(x)/\gamma_0 \le (1+\beta)R$ **then**

    $R_k = \varphi(x)/\gamma_0,\ L_k = L_{k-1},\ \hat{x}_k = x$;

  **else**

    $L_k = R$;

    **if** $\varphi(x)/\gamma_0 \le R_{k-1}$ **then**

      $R_k = \varphi(x)/\gamma_0,\ \hat{x}_k = x$;

    **else**

      $R_k = R_{k-1},\ \hat{x}_k = \hat{x}_{k-1}$;

    **end if**

  **end if**

**end while**

$N = \left\lfloor \frac{c^2}{\alpha^2}\left(1+\frac{1}{\delta}\right)^2 \right\rfloor,\ \hat{x}_{k+1} = \textbf{Subgrad}(\varphi, \mathcal{L}, x_0, R, N)$;

**Output:** $\hat{x}_{k+1}$

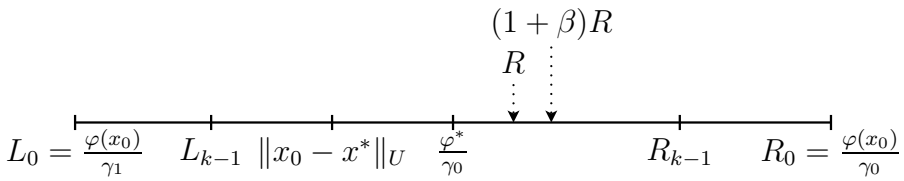



Figure 2.3: Bisection step $k$.

**Theorem 2.2.6.** *Algorithm 3 returns a $\delta$-approximate solution of $(P)$ and takes at most*

$$\frac{1}{\alpha^2\beta^2}\left(1+\log_2\log_2\frac{1}{\alpha}\right)+\frac{4(1+\beta)^2}{\alpha^2}\left(1+\frac{1}{\delta}\right)^2$$

*steps of the subgradient subroutine. If $\beta$ is chosen to be a constant, then the number of steps is*

$$O\left(\frac{1}{\alpha^2\delta^2}+\frac{1}{\alpha^2}\ln\ln\frac{1}{\alpha}\right). \tag{2.15}$$

*Proof.* Let us first analyze the bisection phase (the **while** loop). Repeated use of Lemma 2.2.5 gives

$$\begin{aligned}
q_k &\le (1+\beta)^{\frac{1}{2}} q_{k-1}^{\frac{1}{2}} \\
&\le (1+\beta)^{\frac{1}{2}}(1+\beta)^{\frac{1}{4}} q_{k-2}^{\frac{1}{4}} \\
&\cdots \\
&\le (1+\beta)^{\frac{1}{2}}(1+\beta)^{\frac{1}{4}}\cdots(1+\beta)^{\frac{1}{2^k}} q_0^{\frac{1}{2^k}} \\
&\le (1+\beta)\left(\tfrac{1}{\alpha}\right)^{\frac{1}{2^k}}.
\end{aligned}$$

The smallest integer $k$ for which $(1+\beta)\left(\frac{1}{\alpha}\right)^{\frac{1}{2^k}} \le 2(1+\beta)$ is $k^* := \left\lceil \log_2\log_2\left(\frac{1}{\alpha}\right)\right\rceil$ and hence the total number of lower-level subgradient method iterations of the bisection phase is at most $N_{bis} = \frac{1}{\alpha^2\beta^2}\left(1+\log_2\log_2\left(\frac{1}{\alpha}\right)\right)$. The guarantee (2.15) follows by adding $N_{bis}$ and the number of iterations needed for the finalization

phase (Step 17). It remains to show that the output of the algorithm is as specified. Notice that $\varphi(\hat{x}_{k+1})/\gamma_0 \in [L_k, R_k] = [L_k, \varphi(\hat{x}_k)/\gamma_0]$ and hence

$$\frac{\varphi(\hat{x}_k)}{\varphi(\hat{x}_{k+1})} \leq \frac{R_k}{L_k} = q_k \leq c.$$

Now we just need to apply Proposition 2.2.1 to the subgradient subroutine call of Step 17 of the algorithm using the inequality above:

$$\varphi(\hat{x}_{k+1}) - \varphi^* \leq \frac{\gamma_1}{\sqrt{N+1}} \frac{\varphi(\hat{x}_k)}{\gamma_0} \leq \frac{1}{\sqrt{N+1}} \frac{c\varphi(\hat{x}_{k+1})}{\alpha} \leq \frac{\delta}{1+\delta} \varphi(\hat{x}_{k+1}).$$

□

### 2.2.4 Non-restarting algorithms

Algorithms **SubSearch** and **SubBis** (Algorithms 2 and 3) use the subgradient subroutine *always started from one point*, denoted $x_0$, which is defined as the projection of the origin onto the feasibility set. This point is indeed special as it allows for the key inequalities (2.9) and (2.10) which in turn drive the analysis in both algorithms. The first of these inequalities makes $x_0$ indispensable as the starting point of the very *first* subgradient subroutine call in both algorithms, making it possible to construct *initial* lower and upper bounds on $\varphi^*/\gamma_0$. It is hard to think of a different readily computable point that could serve the same purpose.

The issue we are going to touch upon in this subsection concerns the use of $x_0$ as the starting point in all *subsequent* calls of the subroutine. In our view, *restarting* from this particular point seems to be *convenient* for the sake of the proofs rather than *efficient* algorithmically. Let us elaborate on this a bit. Both algorithms mentioned above can be viewed as simultaneously optimizing (solving $(P)$) and searching for a good upper bound on $\|x_0 - x^*\|_U$ in order to look less

like the *"do-it-all-with-the-available-but-bad-upper-bound"* and more like the *"do-it-all-with-the-good-but-unavailable-upper-bound"* algorithm. Combining these two goals is possible because $\varphi^*/\gamma_0$ is *both* the optimal value of $(P)$ (up to the known constant factor $\gamma_0$) *and* an upper bound on $\|x_0 - x^*\|_U$. It seems likely that the optimization goal could be attained faster if we could use the current best point, as opposed to $x_0$, to start every call of the subroutine. Although both algorithms gather information about increasingly better iterates $\{\hat{x}_k\}$, this knowledge is used *only* to update the upper bound on $\|x_0 - x^*\|_U$ in the next call of the subgradient subroutine and *not* to start the subroutine itself from a better point. There is a good reason for that though. Even if some point $\hat{x}_k$ obtained along the way in one of the algorithms is much better than $x_0$ in terms of its objective value, there are no theoretical guarantees that $\|\hat{x}_k - x^*\|_U$ will be smaller. Starting the subgradient subroutine from such a point thus means combining a probable advantage with a possible disadvantage. A simple observation reveals that the disadvantage factor is under control. Following Figure 2.4, note that for *any* feasible $\hat{x}_k$ we have

$$\|\hat{x}_k - x^*\|_U \leq \|\hat{x}_k\|_U + \|x^*\|_U \leq \frac{\varphi(\hat{x}_k)}{\gamma_0} + \frac{\varphi^*}{\gamma_0} \leq 2\frac{\varphi(\hat{x}_k)}{\gamma_0}.$$

This means that whenever the subgradient method outputs some point $\hat{x}_k$, we have an upper bound on $\|\hat{x}_k - x^*\|_U$ on tap and hence on next call we can run the method starting at $\hat{x}_k$ with $R = 2\varphi(\hat{x}_k)/\gamma_0$, which is exactly twice the upper bound we would use when restarting from $x_0$.

**Nonrestarting version of SubSearch**

Algorithm 4 is a modified version of Algorithm 2 in the spirit of the above discussion. The theoretical performance stays the same.

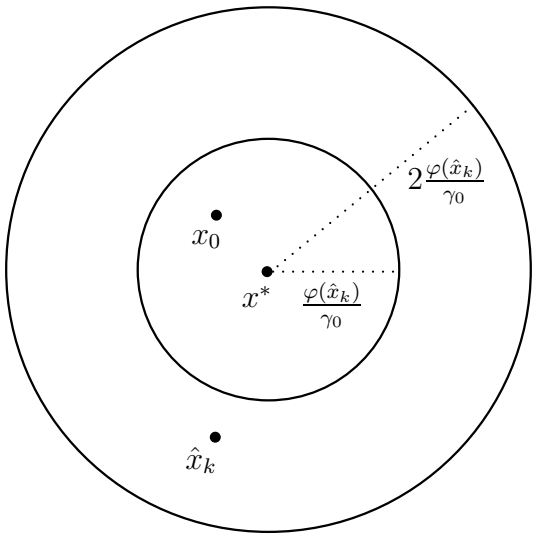



Figure 2.4: Restarting from $x_0$ versus starting from the current best point.

**Theorem 2.2.7.** *Algorithm 4 outputs a $\delta$-approximate solution of $(P)$. The number of calls of the subgradient subroutine is at most $1 + 2\ln\frac{1}{\alpha}$ and the total number of lower-level subgradient steps is hence at most*

$$\frac{4e}{\alpha^2}\left(1+\frac{1}{\delta}\right)^2\left(1+2\ln\frac{1}{\alpha}\right) = O\left(\frac{1}{\alpha^2\delta^2}\ln\frac{1}{\alpha}\right). \tag{2.16}$$

*Proof.* The proof of the upper bound on the number of the outer level iterations is exactly the same as for Algorithm 2. If the algorithm terminates with $k = 1$, it is identical to Nesterov's, and the result follows (we can drop the constant 4 in this case). If $k > 1$, the analysis is analogous except the $2c$ (instead of just $c$) in the definition of $N$ and 2 in the definition of $R$ cancel out:

$$\varphi(\hat{x}_k) - \varphi^* \le \frac{\gamma_1 R}{\sqrt{N+1}} \le \frac{\gamma_1}{\frac{2c}{\alpha}\left(1+\frac{1}{\delta}\right)}\frac{2\varphi(\hat{x}_{k-1})}{\gamma_0} \le \frac{1}{c\left(1+\frac{1}{\delta}\right)}c\varphi(\hat{x}_k) = \frac{\delta}{1+\delta}\varphi(\hat{x}_k).$$

□

**Algorithm 4 (SubSearchNR)** Nonrestarting subgradient search scheme.

**Input:** $\varphi, \mathcal{L}, x_0, \gamma_0, \gamma_1, \delta$;

$\hat{x}_0 = x_0,\ \alpha = \gamma_0/\gamma_1,\ c = \sqrt{e},\ k = 1$;

$N = \left\lfloor \frac{c^2}{\alpha^2}\left(1 + \frac{1}{\delta}\right)^2 \right\rfloor,\ R = \varphi(\hat{x}_0)/\gamma_0$;

$\hat{x}_k = \mathbf{Subgrad}(\varphi, \mathcal{L}, \hat{x}_0, R, N)$;

**while** $\varphi(\hat{x}_k) < \frac{1}{c}\varphi(\hat{x}_{k-1})$ **do**

    $k = k + 1$;

    $N = \left\lfloor \frac{4c^2}{\alpha^2}\left(1 + \frac{1}{\delta}\right)^2 \right\rfloor,\ R = 2\varphi(\hat{x}_{k-1})/\gamma_0$;

    $\hat{x}_k = \mathbf{Subgrad}(\varphi, \mathcal{L}, \hat{x}_{k-1}, R, N)$;

**end while**

**Output:** $\hat{x}_k$

### Nonrestarting bisection algorithm

The following fact plays the role of Lemma 2.2.4 in the design and analysis of a nonrestarting bisection algorithm (Algorithm 5).

**Lemma 2.2.8.** *Let $\hat{x}_{k-1} \in \mathcal{L}$ be arbitrary. If $\varphi^*/\gamma_0 \le R$ and $N = \lfloor \alpha^{-2}\beta^{-2} \rfloor$ for some $\beta > 0$, then*

$$\hat{x}_k := \mathbf{Subgrad}(\varphi, \mathcal{L}, \hat{x}_{k-1}, R + \|\hat{x}_{k-1}\|_U, N)$$

*satisfies*

$$\frac{\varphi(\hat{x}_k)}{\gamma_0} \le (1+\beta)R + \beta\|\hat{x}_{k-1}\|_U \le (1+\beta)R + \beta\frac{\varphi(\hat{x}_{k-1})}{\gamma_0}. \tag{2.17}$$

*Proof.* First notice that $\|\hat{x}_{k-1} - x^*\|_U \le \|\hat{x}_{k-1}\|_U + \|x^*\|_U \le \|\hat{x}_{k-1}\|_U + \varphi^*/\gamma_0 \le \|\hat{x}_{k-1}\|_U + R$ and hence by Proposition 2.2.1 we get

$$\varphi(\hat{x}_k) - \varphi^* \le \gamma_1 \frac{R + \|\hat{x}_{k-1}\|_U}{\sqrt{N+1}} \le \gamma_0\beta(R + \|\hat{x}_{k-1}\|_U).$$

Dividing the above inequality by $\gamma_0$ and rearranging the expression gives the result. The second inequality follows from (2.7). □

The idea with updating lower and upper bounds is the same as in the restarting version of the algorithm. Let $q_k := R_k/L_k$, as before. The improvement guaranteed by a single bisection step is given in the following result.

**Lemma 2.2.9.** *Assume $L_{k-1}$ and $R_{k-1} = \varphi(\hat{x}_{k-1})/\gamma_0$ are lower and upper bounds on $\varphi^*/\gamma_0$, respectively, with $q_{k-1} > 2(1+\beta)$, and let*

$$R := \sqrt{\frac{L_{k-1}R_{k-1}}{1+\beta}}.$$

*If we run the subgradient method as indicated in Lemma 2.2.8 and if $L_k$ and $R_k$ are the new bounds, then*

$$q_k \leq \left(\sqrt{\tfrac{1}{2}} + \beta\right) q_{k-1}. \tag{2.18}$$

*Proof.* Because $q_{k-1} > 2(1+\beta) > 1+\beta$, we are in the same situation as in Lemma 2.2.5 and so $L_{k-1} < R < (1+\beta)R < R_{k-1}$. Notice that the upper bound gets always updated to the value corresponding to the best point found so far, that is, $R_k = \varphi(\hat{x}_k)/\gamma_0$. So we either have $R_k \leq (1+\beta)R + \beta R_{k-1}$, in which case the lower bound stays unchanged, or $L_k = R$ (and $R_k \leq R_{k-1}$, possibly with equality). Therefore

$$q_k = \frac{R_k}{L_k} \leq \max\left\{\frac{(1+\beta)R + \beta R_{k-1}}{L_{k-1}}, \frac{R_{k-1}}{R}\right\}.$$

Notice that $R$ is chosen so that the two expressions in the maximum above are equal, neglecting the $\beta R_{k-1}$ portion of the first. The first expression must therefore be bigger and hence

$$\begin{aligned} q_k \leq \frac{(1+\beta)R + \beta R_{k-1}}{L_{k-1}} &= \sqrt{1+\beta}\sqrt{q_{k-1}} + \beta q_{k-1} \\ &= \left(\sqrt{\tfrac{1+\beta}{q_{k-1}}} + \beta\right) q_{k-1} \\ &< \left(\sqrt{\tfrac{1}{2}} + \beta\right) q_{k-1}. \end{aligned}$$

□

**Theorem 2.2.10.** *Algorithm 5 run with $\beta$ chosen to be a constant such that $\hat{\beta} := \sqrt{\frac{1}{2}} + \beta < 1$ returns a $\delta$-approximate solution of $(P)$ and takes*

$$O\left(\frac{1}{\alpha^2\delta^2} + \frac{1}{\alpha^2}\ln\frac{1}{\alpha}\right) \tag{2.19}$$

*steps of the subgradient subroutine.*

*Proof.* Let us first analyze the bisection phase. Repeated use of Lemma 2.2.9 gives

$$q_k \leq \hat{\beta}^k q_0 \leq \hat{\beta}^k \tfrac{1}{\alpha}.$$

The smallest integer $k$ for which the last quantity drops below $c = 2(1+\beta)$ is $k^* := \left\lceil \frac{\ln(\alpha^{-1}c^{-1})}{\ln\hat{\beta}^{-1}} \right\rceil = O(\ln\frac{1}{\alpha})$ and hence the total number of lower-level subgradient method iterations of the bisection phase is $N_{bis} = O\left(\frac{1}{\alpha^2}\ln\frac{1}{\alpha}\right)$. The guarantee (2.19) follows by adding $N_{bis}$ and the number of iterations needed for the finalization phase (Step 12). It remains to show that the output of the algorithm is as specified. The analysis, however, is identical to that in Theorem 2.2.7. □

Note that the nonrestarting version of the bisection algorithm has a slightly worse complexity bound — we have lost one logarithm in (2.19) in comparison with (2.15). However, the bisection strategy still manages to separate the $\delta$ from the logarithmic term as compared to the bound (2.16) for the **SubSearch** algorithm.

## 2.3 Algorithms based on smoothing

We have seen in Section 2.2 that problem $(P)$ allows for simple algorithms that require $O(\delta^{-2})$ iterations of the subgradient method. We have improved Nesterov's subgradient search algorithm (Algorithm 2), which needs $O(\alpha^{-2}\delta^{-2}\ln\frac{1}{\alpha})$ iterations, by incorporating a simple bisection idea and obtained Algorithm 3 with

**Algorithm 5 (SubBisNR)** Nonrestarting subgradient bisection scheme.

**Input:** $\varphi, \mathcal{L}, x_0, \gamma_0, \gamma_1, \beta, \delta$;

$k = 0,\ \hat{x}_0 = x_0,\ L_0 = \|x_0\|_U,\ R_0 = \varphi(x_0)/\gamma_0$;

$\alpha = \gamma_0/\gamma_1,\ c = 2(1+\beta),\ N = \left\lfloor \frac{1}{\alpha^2\beta^2} \right\rfloor$;

**while** $R_k/L_k > c$ **do**

  $k = k+1,\ R = \sqrt{\frac{L_{k-1}R_{k-1}}{1+\beta}},\ \hat{x}_k = \textbf{Subgrad}(\varphi, \mathcal{L}, \hat{x}_{k-1}, R, N)$;

  **if** $\varphi(\hat{x}_k)/\gamma_0 \le (1+\beta)R + \beta\varphi(\hat{x}_{k-1})/\gamma_0$ **then**

    $L_k = L_{k-1},\ R_k = \varphi(\hat{x}_k)/\gamma_0$;

  **else**

    $L_k = R,\ R_k = \varphi(\hat{x}_k)/\gamma_0$;

  **end if**

**end while**

$N = \left\lfloor \frac{4c^2}{\alpha^2}\left(1+\frac{1}{\delta}\right)^2 \right\rfloor,\ R = \frac{2\varphi(\hat{x}_k)}{\gamma_0},\ \hat{x}_{k+1} = \textbf{Subgrad}(\varphi, \mathcal{L}, \hat{x}_k, R, N)$;

**Output:** $\hat{x}_{k+1}$

the slightly better $O(\alpha^{-2}\delta^{-2} + \delta^{-2} \ln \ln \frac{1}{\alpha})$ guarantee. That is, we have improved the dependence on the rounding parameter $\alpha$, but not on the error parameter $\delta$.

We start in the following subsection by briefly describing Nesterov's smoothing technique [21] and the implied algorithm for smooth minimization of nonsmooth functions. It is not our intention to describe the approach in full generality; rather, we will adapt the results to the setting of problem $(P)$ – the minimization of a nonnegative sublinear (convex and homogeneous) function vanishing only at the origin.

### 2.3.1 The setting

In [21] Nesterov considers a rather general *nonsmooth* convex optimization problem and shows that it is possible to solve it in $O(\epsilon^{-1})$ iterations of a gradient-type method, if a solution within absolute error $\epsilon$ is sought. His novel approach involves two phases. The first is a pre-processing phase in which one approximates the objective function by a smooth function with Lipschitz continuous gradient. The second phase amounts to running an optimal smooth method of the type [18], [19] (Section 2.2) with complexity $O(\epsilon^{-1/2})$ applied to the smooth function.

We will describe the model for sublinear functions. Consider the following more general version of problem $(P)$, with $\varphi$ replaced by an arbitrary sublinear function and $\mathcal{L}$ (or $\mathcal{L}$ intersected with a large ball) replaced by a compact and convex subset $\mathcal{Q}_1$ of $\mathbf{E}_1 := \mathbf{E}$:

$$\boxed{\varphi^* := \min_x \{\varphi(x) \ : \ x \in \mathcal{Q}_1\}.} \qquad (P')$$

Notice that $\varphi$ can be written as

$$\varphi(x) = \max_g \{\langle g, x\rangle \ : \ g \in \partial\varphi(0)\}, \qquad (2.20)$$

To allow for some modeling flexibility, the purpose of which will be clear later, we will instead consider the following family of representations of the objective function:

$$\varphi(x) = \max_y \{\langle Ax, y\rangle \ : \ y \in \mathcal{Q}_2\}. \tag{2.21}$$

Here we are introducing a new finite-dimensional real vector space $\mathbf{E}_2$, a linear operator $A\colon \mathbf{E}_1 \to \mathbf{E}_2^*$ and a compact and convex set $\mathcal{Q}_2 \subset \mathbf{E}_2$.

**Definition 2.3.1.** The *adjoint* of $A$ is the operator $A^* : \mathbf{E}_2 \to \mathbf{E}_1^*$ defined via

$$\langle Ax, y\rangle = \langle A^*y, x\rangle \qquad \forall\ x \in \mathbf{E}_1,\ y \in \mathbf{E}_2.$$

We assume that the spaces $\mathbf{E}_1$ and $\mathbf{E}_2$ are equipped with norms $\|\cdot\|_1$ and $\|\cdot\|_2$ respectively[3], and the dual spaces $\mathbf{E}_1^*$ and $\mathbf{E}_2^*$ with the corresponding dual norms

$$\|g\|_1^* := \max\{\langle g, x\rangle \ : \ \|x\|_1 \le 1\} \quad \text{and} \quad \|h\|_2^* := \max\{\langle h, y\rangle \ : \ \|y\|_2 \le 1\}, \tag{2.22}$$

for $g \in \mathbf{E}_1^*$ and $h \in \mathbf{E}_2^*$.

**Definition 2.3.2.** The norm of $A$ is defined by

$$\|A\|_{1,2} := \max_{x,y} \{\langle Ax, y\rangle \ : \ \|x\|_1 = 1,\ \|y\|_2 = 1\}. \tag{2.23}$$

One can similarly define $\|A^*\|_{2,1}$.

It follows easily from the definition that

$$\|A\|_{1,2} = \max_x \{\|Ax\|_2^* \ : \ \|x\|_1 = 1\} = \|A^*\|_{2,1} = \max_y \{\|A^*y\|_1^* \ : \ \|y\|_2 = 1\}. \tag{2.24}$$

[3]The numbers are subscripts referring to the spaces in which the norms are defined and are not intended to suggest the use of the $\ell_1$ and $\ell_2$ norms.

**Example 2.3.3** (Example 1 in [21])**.** Consider the function

$$\varphi_\infty(x) := \max_i\{|\langle a_i, x\rangle| \ : \ i = 1, 2, \ldots, m\},$$

where $x \in \mathbf{E}_1 = \mathbf{R}^n$, $a_i \in \mathbf{E}_1^* = \mathbf{R}^n$ and $\langle g, x\rangle = \sum_{i=1}^n g_i x_i$. Note that in the following three representations of $\varphi_\infty$ the structure of the set $\mathcal{Q}_2$ gets simpler as the dimension of the space $\mathbf{E}_2$ increases.

1. $\mathbf{E}_2 = \mathbf{E}_2^* = \mathbf{R}^n$, $\mathcal{Q}_2 = \text{conv}\{\pm a_i \ : \ i = 1, 2, \ldots, m\}$ and $A = I$. This seems to be the most natural and straightforward representation.

2. $\mathbf{E}_2 = \mathbf{E}_2^* = \mathbf{R}^m$, $\mathcal{Q}_2 = \{y \in \mathbf{R}^m \ : \ \sum_{i=1}^m |y_i| \leq 1\}$ and $A$ is the $m \times n$ matrix with rows $a_1, \ldots, a_m$. In this case we have
$$\varphi_\infty(x) = \max\left\{\sum_{i=1}^m y_i\langle a_i, x\rangle \ : \ \sum_{i=1}^m |y_i| \leq 1\right\}.$$

3. $\mathbf{E}_2 = \mathbf{E}_2^* = \mathbf{R}^{2m}$, $\mathcal{Q}_2$ is the unit simplex in $\mathbf{R}^{2m}$ and $A$ is the $2m \times n$ matrix with rows composed of $a_1, \ldots, a_m$ and $-a_1, \ldots, -a_m$:
$$\varphi_\infty(x) = \max\left\{\sum_{i=1}^m (y_i' - y_i'')\langle a_i, x\rangle \ : \ \sum_{i=1}^m y_i' + y_i'' = 1, \ y_i', y_i'' \geq 0\right\}.$$

If we let

$$\theta(y) := \min_x\{\langle A^* y, x\rangle \ : \ x \in \mathcal{Q}_1\},$$

then because both $\mathcal{Q}_1$ and $\mathcal{Q}_2$ are convex and compact and $\langle A^* y, x\rangle \equiv \langle y, Ax\rangle$ is bilinear, we can apply a standard minimax result[4] and rewrite $(P')$ as follows:

$$\boxed{\varphi^* = \theta^* := \max_y\{\theta(y) \ : \ y \in \mathcal{Q}_2\}.} \qquad (P'')$$

[4]A *minimax* result is a theorem which asserts that $\min_{x\in\mathcal{Q}_1} \max_{y\in\mathcal{Q}_2} \rho(x, y) = \max_{y\in\mathcal{Q}_2} \min_{x\in\mathcal{Q}_1} \rho(x, y)$, under certain conditions imposed on the sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$ and the function $\rho$. For example, the equality holds if both sets are convex and compact subsets of a finite-dimensional real vector space and $\rho$ is bilinear. A classic reference is J. von Neumann and O. Morgenstern [34]. For a modern treatment based on Fenchel duality see chapters 3 and 4 of J.M. Borwein and A.S. Lewis [5], and, in particular, Exercise 4.2.16(c).

### 2.3.2 Smoothing and an efficient smooth method

In the first phase of Nesterov's approach, the objective function of $(P')$ is approximated by a smooth convex function with Lipschitz continuous gradient. An approximation with error $O(\epsilon)$ has gradient with Lipschitz constant of $O(1/\epsilon)$. The second phase consists of applying to $(P)$ (with the objective function replaced by its smooth approximation) an efficient smooth method (Algorithm 6) requiring $O(1/\sqrt{\epsilon})$ iterations of a gradient type. The smooth algorithm is capable of producing points $\hat{x}$ and $\hat{g}$ feasible to *both* $(P')$ and $(P'')$, respectively, such that $\varphi(\hat{x}) - \theta(\hat{g}) = O(1/\epsilon)$. Because $\varphi^* = \theta^*$, these points are approximate optimizers in their respective problems (in the absolute sense).

The smoothing approach assumes the availability of *prox-functions* $d_1$ and $d_2$ for the sets $\mathcal{Q}_1$ and $\mathcal{Q}_2$, respectively. These are continuous and strongly convex nonnegative functions defined on these sets, with convexity parameters $\sigma_1$ and $\sigma_2$, respectively. Let $x_0$ be the *center* of the set $\mathcal{Q}_1$ (think $\mathcal{Q}_1 = \mathcal{L}$):

$$x_0 := \arg\min_x \{d_1(x) \ : \ x \in \mathcal{Q}_1\}. \tag{2.25}$$

For example, if $d_1(x) := \frac{1}{2}\|x\|_1^2$ (so $\sigma_1 = 1$) and $\mathcal{Q}_1$ is the intersection of $\mathcal{L}$ and a large-enough ball centered at the origin, then $x_0$ coincides with its earlier definition. We assume that $d_1$ vanishes at its center and hence the above properties imply

$$d_1(x) \geq \tfrac{1}{2}\sigma_1\|x - x_0\|_1^2.$$

In the example above, we subtract $\|x_0\|_1^2/2$ from $d_1$ and then the inequality holds as an equation. In an analogous fashion we define the center $y_0$ of $\mathcal{Q}_2$ and assume that $d_2$ vanishes at $y_0$. Therefore

$$d_2(y) \geq \tfrac{1}{2}\sigma_2\|y - y_0\|_2^2.$$

Finally, let $D_1$ and $D_2$ satisfy

$$D_1 \geq \max_x \{d_1(x) \ : \ x \in \mathcal{Q}_1\}$$

and

$$D_2 \geq \max_y \{d_2(y) \ : \ y \in \mathcal{Q}_2\}.$$

**Proposition 2.3.4** (Nesterov [21], Theorem 1)**.** *For $\mu > 0$, the function*

$$\varphi_\mu(x) := \max_y \{\langle Ax, y\rangle - \mu d_2(y) \ : \ y \in \mathcal{Q}_2\}, \tag{2.26}$$

*is a continuously differentiable uniform approximation of $\varphi$:*

$$\varphi_\mu(x) \leq \varphi(x) \leq \varphi_\mu(x) + \mu D_2 \qquad \forall\, x \in \mathbf{E}_1. \tag{2.27}$$

*Moreover, if we denote by $y_\mu(x)$ the (unique) maximizer from (2.26), then the gradient of $\varphi_\mu(x)$ is given by $\nabla\varphi_\mu(x) = A^* y_\mu(x)$ and is Lipschitz continuous with constant*

$$\gamma_\mu = \frac{1}{\mu\sigma_2}\|A\|_{1,2}^2. \tag{2.28}$$

The smooth version of $(P')$ therefore is

$$\boxed{\min_x \{\varphi_\mu(x) \ : \ x \in \mathcal{Q}_1\}.} \tag{$P'_{smooth}$}$$

The main result of [21] is the following:

**Theorem 2.3.5** (Nesterov [21], Theorem 3)**.** *If we apply Algorithm 6 to problem $(P'_{smooth})$ with smoothing parameter*

$$\mu = \frac{2\|A\|_{1,2}}{N+1}\sqrt{\frac{D_1}{\sigma_1\sigma_2 D_2}} \tag{2.29}$$

*and if*

$$x = \mathbf{Smooth}(\varphi_\mu, \gamma_\mu, \mathcal{Q}_1, x_0, N),$$

*then*[5]

$$\varphi(x) - \varphi^* \leq \frac{4\|A\|_{1,2}}{N+1}\sqrt{\frac{D_1 D_2}{\sigma_1 \sigma_2}}.$$

**Algorithm 6 (Smooth)** Efficient smooth method.

1: **Input:** $\psi, \gamma, \mathcal{Q}_1, x_0, N$;

2: **for** $k = 0$ **to** $N$ **do**

3: Compute $\psi(x_k)$ and $\nabla\psi(x_k)$;

4: $y_k = \arg\min\{\langle\nabla\psi(x_k), x - x_k\rangle + \frac{\gamma}{2}\|x - x_k\|_1^2 \ : \ x \in \mathcal{Q}_1\}$;

5: $z_k = \arg\min\{\sum_{i=0}^{k} \frac{i+1}{2}\langle\nabla\psi(x_i), x - x_i\rangle + \frac{\gamma}{\sigma_1}d_1(x) \ : \ x \in \mathcal{Q}_1\}$;

6: $x_{k+1} = \frac{2}{k+3}z_k + \frac{k+1}{k+3}y_k$;

7: **end for**

8: **Output:** $y_N$

### 2.3.3 The main result

We will use the above theorem in the same way as Proposition 2.2.1 to devise a $O(1/\delta)$-algorithm for finding a $\delta$-approximate solution of $(P)$. Algorithms of this type, formulated for several specific choices of objective functions, were suggested already in [23] and [22]. These methods are similar in spirit to Algorithm 2, recursively updating an upper bound on $\varphi^*$. We give a single and faster algorithm applicable to the problems considered in the cited papers. Our contribution lies mainly in improving the theoretical complexity by incorporating a bisection speedup. As in the previous section, it is possible to formulate a nonrestarting version of our algorithm by sacrificing the double logarithm in the theoretical complexity for a single one.

[5]The original theorem states the result as a gap between $\varphi(x)$ and $\theta(y)$ for a certain $y \in \mathcal{Q}_2$.

**Preliminaries**

Let us return to problem $(P)$, using the representation (2.21) for the objective function (hence $\mathcal{Q}_1 = \mathcal{L}$), and approach it with the tools described in the previous subsections. Let $\mathbf{E}_1 := \mathbf{E}$ and assume that $U\colon \mathbf{E}_1 \to \mathbf{E}_1^*$ defines an ellipsoidal rounding of $\partial\varphi(0) = A^*\mathcal{Q}_2$ such that (2.3) holds with $\gamma_0 = 1$. Notice that the inequalities (2.7), (2.9) and (2.10) are implied by the former. To be able to obtain an algorithm guaranteeing a $\delta$-approximate output in relative scale, the choice of the primal norm as the norm coming from the rounding is crucial:

$$\|x\|_1 := \|x\|_U \qquad \forall\, x \in \mathbf{E}_1.$$

If we wish to apply Algorithm 6, we need to supply it a *bounded* subset of $\mathcal{L}$ (which is unbounded) containing the minimizer. Observe that as long as $\varphi^* \leq R$ for some positive number $R$, (2.10) guarantees that all minimizers of $(P)$ lie in the set

$$Q_1(R) := \mathcal{L} \cap \{x \ :\ \|x - x_0\|_U \leq R\},$$

where $x_0$ — the projection of the origin onto $\mathcal{L}$ in the $U$-norm — is the center of $\mathcal{Q}_1(R)$ as defined in (2.25) if we choose the prox-function for $\mathcal{Q}_1(R)$ to be

$$d_1(x) := \tfrac{1}{2}\|x - x_0\|_U^2.$$

In this case $\sigma_1 = 1$ and $D_1 = \max\{d_1(x) \ :\ x \in \mathcal{Q}_1(R)\} = \frac{1}{2}R^2$. We leave the choice of $d_2$ purposely open to allow for fine-tuning for particular applications.

A direct consequence of Theorem 2.3.5 with the settings described above is the following analogue of Lemma 2.2.4:

**Lemma 2.3.6.** *If $\varphi^* \leq R$, $\beta > 0$ and we set*

$$N = \left\lfloor \frac{2\sqrt{2}\|A\|_{1,2}}{\beta}\sqrt{\frac{D_2}{\sigma_2}} \right\rfloor$$

*for some* $\beta > 0$,

$$\mu = \frac{\sqrt{2}\|A\|_{1,2}R}{N+1}\sqrt{\frac{1}{\sigma_2 D_2}}$$

*and* $\gamma_\mu$ *as in (2.28), then*

$$x = \mathbf{Smooth}(\varphi_\mu, \gamma_\mu, \mathcal{Q}_1(R), x_0, N)$$

*satisfies*

$$\varphi(x) \leq (1+\beta)R.$$

The above lemma leads to a bisection algorithm (Algorithm 7) in the same way as we have seen it in the section on subgradient algorithms. The main result follows:

**Theorem 2.3.7.** *Algorithm 7 returns a* $\delta$*-approximate solution of* $(P)$ *and takes at most*

$$\frac{2\sqrt{2}\|A\|_{1,2}}{\beta}\sqrt{\frac{D_2}{\sigma_2}}\left(\log_2\log_2\frac{1}{\alpha}\right) + 2\sqrt{2}(1+\beta)\|A\|_{1,2}\left(1+\frac{1}{\delta}\right)\sqrt{\frac{D_2}{\sigma_2}}$$

*steps of the smooth optimization subroutine. If* $\beta$ *is chosen to be a constant, then the number of steps is*

$$O\left(\|A\|_{1,2}\sqrt{\frac{D_2}{\sigma_2}}\left(\frac{1}{\delta}+\ln\ln\frac{1}{\alpha}\right)\right). \tag{2.30}$$

*Proof.* The analysis is completely analogous to the proofs from the previous section. □

### 2.3.4 A direct representation of the objective function

We can get rid of the dependence on $\|A\|_{1,2}$ in (2.30) by identifying $\mathbf{E}_2$ with $\mathbf{E}_1^*$ (and consequently $\mathbf{E}_1$ with $\mathbf{E}_2^*$). In this case we can simply choose $A = I$ and

**Algorithm 7 (SmoothBis)** Smoothed bisection scheme.

**Input:** $\varphi, \alpha, \beta, \delta, x_0$;

$k = 0,\ \hat{x}_0 = x_0,\ L_0 = \|x_0\|_U,\ R_0 = \varphi(x_0)$;

$c = 2(1+\beta),\ N = \left\lfloor \frac{2\sqrt{2}\|A\|_{1,2}}{\beta}\sqrt{\frac{D_2}{\sigma_2}} \right\rfloor$;

**while** $R_k/L_k > c$ **do**

  $k = k + 1$;

  $R = \sqrt{\frac{L_{k-1}R_{k-1}}{1+\beta}},\ \mu = \frac{\sqrt{2}\|A\|_{1,2}R}{N+1}\sqrt{\frac{1}{\sigma_2 D_2}},\ \gamma_\mu = \frac{\|A\|_{1,2}^2}{\mu\sigma_2}$;

  $x = \mathbf{Smooth}(\varphi_\mu, \gamma_\mu, \mathcal{Q}_1(R), x_0, N)$;

  **if** $\varphi(x) \le (1+\beta)R$ **then**

    $R_k = \varphi(x),\ L_k = L_{k-1},\ \hat{x}_k = x$;

  **else**

    $L_k = R$;

    **if** $\varphi(x) \le R_{k-1}$ **then**

      $R_k = \varphi(x),\ \hat{x}_k = x$;

    **else**

      $R_k = R_{k-1},\ \hat{x}_k = \hat{x}_{k-1}$;

    **end if**

  **end if**

**end while**

$R = \varphi(\hat{x}_k)$;

$N = \left\lfloor 2\sqrt{2}c\|A\|_{1,2}(1+\frac{1}{\delta})\sqrt{\frac{D_2}{\sigma_2}} \right\rfloor,\ \mu = \frac{\sqrt{2}\|A\|_{1,2}R}{N+1}\sqrt{\frac{1}{\sigma_2 D_2}},\ \gamma_\mu = \frac{\|A\|_{1,2}^2}{\mu\sigma_2}$;

$\hat{x}_{k+1} = \mathbf{Smooth}(\varphi_\mu, \gamma_\mu, \mathcal{Q}_1(R), x_0, N)$;

**Output:** $\hat{x}_{k+1}$

consider the following structural model for the objective function:

$$\varphi(x) = \max_g \{\langle g, x\rangle \ : \ g \in \mathcal{Q}_2\}.$$

Let us set $\|g\|_2 = \|g\|_1^* = \|g\|_U^*$ and select the following prox-function for $\mathcal{Q}_2$ (with center at the origin):

$$d_2(g) = \tfrac{1}{2}(\|g\|_U^*)^2.$$

Clearly $\sigma_2 = 1$ and $D_2 \leq \frac{1}{2\alpha^2}$ — the second inequality follows from the ellipsoidal rounding inclusion (2.3) and the assumption $\gamma_0 = 1$. Also observe that since $\|\cdot\|_2^* \equiv \|\cdot\|_1$, we have

$$\|A\|_{1,2} = \max\{\|Ax\|_2^* \ : \ \|x\|_1 = 1\} = \max\{\|x\|_1 \ : \ \|x\|_1 = 1\} = 1.$$

Substituting for the values of these parameters into (2.30) gives the following guarantee:

$$O\left(\frac{1}{\alpha\delta} + \frac{1}{\alpha}\ln\ln\frac{1}{\alpha}\right).$$

**Remark 2.3.8.** *Observe that, in principle, we do not lose generality by "excluding" $A$ because we can simply set the "new" $\mathcal{Q}_2$ to be equal to the "old" $A^*\mathcal{Q}_2$. However, this sacrifice in modeling flexibility means that $\mathcal{Q}_2$ always coincides with $\partial\varphi(0)$, which has to be of a simple structure for the algorithm to work efficiently. This is mainly due to the need to compute derivatives of $\varphi_\mu$, which amounts to solving a concave quadratic maximization problem over $\mathcal{Q}_2$ (2.26). If this problem can not be solved efficiently (say in a closed form), the method will likely be impractical.*

## 2.4 Applications

In this section we apply the fastest of the algorithms developed in this chapter — the bisection algorithm based on smoothing (**SmoothBis**) — to several problems of the form $(P)$.

### 2.4.1 Minimizing the maximum of absolute values of linear functions

In this subsection we consider problem $(P)$ with the objective function from Example 2.3.3:

$$\min\{\varphi_\infty(x) \ : \ x \in \mathcal{L}\}. \tag{2.31}$$

Many seemingly unrelated problems can be reformulated into the above form. For example, by (2.31) we can model:

- the truss topology design problem,
- the problem of the construction of a $c$-optimal statistical design, and
- the problem of finding a solution of an underdetermined linear system with the smallest $\ell_1$ norm.

In all the examples above the feasible set $\mathcal{L}$ is one-dimensional. We postpone the discussion of these applications until Chapter 3 since in it we focus on developing specialized algorithms for solving a certain reformulation of (2.31). Let us at least show now how we can solve this problem using the results of Section 2.3.

#### Applying the algorithm

We will work with the last of the three representations for the objective function from Example 2.3.3:

$$\varphi_\infty(x) = \max\{|\langle a_i, x\rangle| \ : \ i = 1, 2, \ldots, m\} = \max_y\{\langle Ax, y\rangle \ : \ y \in \mathcal{Q}_2\},$$

with $\mathcal{Q}_2$ being the unit simplex in $\mathbf{R}^{2m}$ and $A$ the $2m \times n$ matrix with rows $a_i, -a_i, \ i = 1, \ldots, m$. In addition, assume that the vectors $a_i, \ i = 1, 2, \ldots, m$,

span $\mathbf{E}_1^* = \mathbf{R}^n$. It seems natural to choose $\|y\|_2 := \sum_i |y_i|$ so that $\|y\|_2 = 1$ for all $y \in \mathcal{Q}_2$. If we let

$$d_2(y) := \ln 2m + \sum_{i=1}^{2m} y_i \ln y_i$$

and define $0 \times \ln 0 := \lim_{\tau \downarrow 0} \tau \ln \tau = 0$, then by the following lemma, $d_2$ is a prox-function on $\mathcal{Q}_2$ with center $y_0 := (\frac{1}{2m}, \ldots, \frac{1}{2m})$:

**Lemma 2.4.1.** *$d_2$ is strongly convex on $\mathcal{Q}_2$, with respect to $\|\cdot\|_2$, with convexity parameter $\sigma_2 = 1$.*

*Proof.* It suffices to show that $d_2(y) \geq \frac{1}{2}\|y - y_0\|_2^2$. This can be proved by elementary means using only the Cauchy-Schwarz inequality (see, for example, Exercise 3.3.25(d) in [5]) or, using differentiation and a certain knowledge about convex functions (Lemma 3 in [21]). □

It is easy to see that $D_2 = \sup\{d_2(y) \ : \ y \in \mathcal{Q}_2\} = \ln 2m$ (the supremum is attained at each of the boundary vertices). Finally, let us compute the norm of the linear operator $A$:

$$\begin{aligned}
\|A\|_{1,2} &= \max\{\|Ax\|_2^* \ : \ \|x\|_1 = 1\} \\
&= \max\{\|Ax\|_\infty \ : \ \|x\|_U = 1\} \\
&= \max\{\varphi(x) \ : \ \|x\|_U = 1\} \\
&= \tfrac{1}{\alpha}.
\end{aligned}$$

The last step follows from inequality (2.7) in view of our assumption that $\gamma_0 = 1$. It is shown in Lemma 4 of [21] that the smooth approximation of $\varphi$ is given by

$$\varphi_\mu(x) = \mu \ln \left( \frac{1}{2m} \sum_{i=1}^{m} \left[ e^{\langle a_i, x \rangle / \mu} + e^{\langle -a_i, x \rangle / \mu} \right] \right).$$

Since $\partial\varphi(0) = \operatorname{conv}\{\pm a_i, i = 1, 2, \ldots, m\}$ is a centrally symmetric subset of $\mathbf{R}^n$, we may assume that a good rounding, with $\frac{1}{\alpha} = O(\sqrt{n})$, is available to us.

**The complexity**

The performance of Algorithm 7 for this problem then by substituting into (2.30) is

$$O\left(\sqrt{n \ln m}\left(\ln \ln n + \frac{1}{\delta}\right)\right).$$

This improves on the result in [22], where the author gives a bound of

$$O\left(\frac{\sqrt{n \ln m}}{\delta} \ln n\right).$$

### 2.4.2 Minimizing the sum of absolute values of linear functions

Consider problem $(P)$ with the following objective function:

$$\varphi_1(x) = \sum_{i=1}^{m} |\langle a_i, x \rangle|.$$

As usual, we assume that the vectors $a_1, a_2, \ldots, a_m$ span $\mathbf{E}_1^*$.

**Applying the algorithm**

Let $\mathbf{E}_1 = \mathbf{E}_1^* = \mathbf{R}^n$ and $\mathbf{E}_2 = \mathbf{E}_2^* = \mathbf{R}^m$ and let us represent $\varphi_1$ as

$$\varphi_1(x) = \max_y \{\langle Ax, y \rangle \ : \ y \in \mathcal{Q}_2\}, \tag{2.32}$$

where $\mathcal{Q}_2 = \{y \in \mathbf{R}^m \ : \ |y_i| \leq 1, \ i = 1, 2, \ldots, m\}$ and $A$ is the $m \times n$ matrix with rows $a_1, \ldots, a_m$. Usually we first find a rounding of $\partial\varphi_1(0)$ and using the rounding operator define a norm on $\mathbf{E}_1$. Because of the simple structure of $\mathcal{Q}_2$, we will instead start by defining $\|y\|_2 := (\sum_i y_i^2)^{1/2}$ and noting that this leads to a $\sqrt{m}$-rounding of $\mathcal{Q}_2$:

$$\mathcal{B}(I, 1) \subseteq \mathcal{Q}_2 \subseteq \mathcal{B}(I, \sqrt{m}), \tag{2.33}$$

with $I\colon \mathbf{R}^m \to \mathbf{R}^m$ denoting the identity operator. We will show now how this naturally leads to a rounding operator defined on $\mathbf{E}_1$ enjoying the same quality of rounding.

**Lemma 2.4.2** (Nesterov [23], Lemma 2)**.** *If the vectors $a_1, \ldots, a_m$ span $\mathbf{R}^m$, then $\|x\|_1 := \|Ax\|_2^*$ defines a norm on $\mathbf{R}^n$. Moreover, if we let $U := A^T A$ (a positive definite matrix), then*

$$\|\cdot\|_1 \equiv \|\cdot\|_U$$

*and*

$$\mathcal{B}(U, 1) \subseteq \partial\varphi(0) = A^T \mathcal{Q}_2 \subseteq \mathcal{B}(U, \sqrt{m}).$$

*Proof.* Note that $\|x\|_1 = \|Ax\|_2^* = \langle Ax, Ax\rangle^{1/2} = \langle Ux, x\rangle^{1/2} = \|x\|_U$. The equality $\partial\varphi(0) = A^T\mathcal{Q}_2$ follows from (2.32). In view of (2.33) we obtain

$$\varphi(x) = \max_{y\in\mathcal{Q}_2}\langle Ax, y\rangle \leq \max_{y\in\mathcal{B}(I,\sqrt{m})}\langle Ax, y\rangle = \max_{\|y\|_2\leq\sqrt{m}}\langle Ax, y\rangle = \sqrt{m}\|Ax\|_2^* = \sqrt{m}\|x\|_1$$

and

$$\varphi(x) = \max_{y\in\mathcal{Q}_2}\langle Ax, y\rangle \geq \max_{y\in\mathcal{B}(I,1)}\langle Ax, y\rangle = \max_{\|y\|_2\leq 1}\langle Ax, y\rangle = \|Ax\|_2^* = \|x\|_1.$$

□

Let us define

$$d_2(y) := \tfrac{1}{2}\|y\|_2^2,$$

so that the convexity parameter of this prox-function is $\sigma_2 = 1$. It follows from (2.33) that $D_2 = \max\{d_2(y) \ : \ y \in \mathcal{Q}_2\} \leq \frac{1}{2}m$. Finally, let us compute the norm of the linear operator $A$:

$$\|A\|_{1,2} = \max\{\|Ax\|_2^* \ : \ \|x\|_1 = 1\} = \max\{\|x\|_1 \ : \ \|x\|_1 = 1\} = 1.$$

**The complexity**

The performance of Algorithm 7 on this problem then by substituting into (2.30) is

$$O\left(\sqrt{m}\left(\frac{1}{\delta}+\ln\ln m\right)\right).$$

This improves on the result in [23], where the author gives the bound

$$O\left(\frac{\sqrt{m}\ln m}{\delta}\right).$$

### 2.4.3 Minimizing the maximum of linear functions over a simplex

**Motivation: The value of a two-person zero-sum matrix game with non-negative coefficients**

Let $\hat{A} \in \mathbf{R}^{m\times n}$ be a real matrix with nonnegative entries and rows $a_1, \ldots, a_m$. Consider the following game. There are two players: a row player ($R$) and a column player ($C$). Player $R$ chooses a probability distribution $y$ over the rows of matrix $\hat{A}$ and $C$ chooses a probability distribution $x$ over the columns. After that, $C$ pays $y^T\hat{A}x$ dollars to $R$. Assume the players are *conservative*, that is, $C$ wishes to minimize his worst-case loss and $R$ wants to maximize his worst-case win. That is, $C$ prefers to choose strategy

$$x^* \in \arg\min_{x\in\Delta_n}\max_{y\in\Delta_m} y^T\hat{A}x$$

and similarly, $R$ wishes to choose strategy

$$y^* \in \arg\max_{y\in\Delta_m}\min_{x\in\Delta_n} y^T\hat{A}x.$$

The set $\Delta_n$ (resp. $\Delta_m$) denotes the unit simplex in $\mathbf{R}^n$ (resp. $\mathbf{R}^m$). A classical result by von Neumann [34] says that[6]

$$\varphi^* := \min_{x\in\Delta_n}\max_{y\in\Delta_m} y^T\hat{A}x = \max_{y\in\Delta_m}\min_{x\in\Delta_n} y^T\hat{A}x.$$

The value $\varphi^*$ is called the *value of the game*. Note that if we let $\mathcal{Q}_1 := \Delta_n$ and

$$\varphi(x) = \max\{\langle a_i, x\rangle \ : \ i = 1, 2, \ldots, m\},$$

then we can write

$$\varphi^* = \min_x\{\varphi(x) \ : \ x \in \mathcal{Q}_1\}.$$

**Applying the algorithm**

First note that

$$\partial\varphi(0) = \mathrm{conv}\{a_i \ : \ i = 1, 2, \ldots, m\},$$

which fails to satisfy (2.1) due to the assumption on nonnegativity of the entries of $\hat{A}$. To remedy this situation, we will follow a trick suggested in Nesterov [22]. Notice that we are interested in $\varphi$ as defined on $\Delta_n$ only, which is a subset of the nonnegative orthant. Let us therefore define

$$\hat{\varphi}(x) := \max\{\langle a_i, |x|\rangle \ : \ i = 1, 2, \ldots, m\},$$

where $|x| = (|x_1|, \ldots, |x_n|)$ and observe that

$$\hat{\varphi}(x) = \varphi(x) \quad \forall\ x \in \mathbf{R}^n_+$$

and

$$\partial\hat{\varphi}(0) = \mathrm{conv}\bigcup_{i=1}^{m}\{g \ : \ -a_i \le g \le a_i\}.$$

[6]For a modern proof based on Fenchel duality see, for example, Exercise 4.2.16 in [5].

It is particularly interesting to note that $\partial\hat{\varphi}(0)$ is a *sign-invariant* set, one that with every point $g$ contains all points obtained by arbitrarily changing the signs of the coordinates of $g$. In fact, $\partial\hat{\varphi}(0)$ is the smallest sign-invariant set containing $\partial\varphi(0)$. Nesterov shows that sign-invariant convex bodies admit a more efficient rounding algorithm than the more general central-symmetric sets mainly due to the possibility of working only with *diagonal* positive definite matrices defining the rounding.

Instead of rounding $\partial\varphi(0)$ one can therefore find an ellipsoidal rounding of $\partial\hat{\varphi}(0)$ (defined by a diagonal positive definite matrix $U$) with $\frac{1}{\alpha} = O(\sqrt{n})$ and then deduce inequality (2.7), which holds for all $x \in \mathbf{R}^n_+$ (Lemma 5, [22]). Smoothing of $\varphi$ (and hence of $\hat{\varphi}$ on the domain of interest) can be performed in complete analogy with the situation in Subsection 2.4.1. The choice of the representation of the objective function, the choice of the prox-function for $\mathcal{Q}_2$ and the implied bounds are all identical (the only change is that the dimension drops from $2m$ to $m$).

**The complexity**

The complexity guarantee of Algorithm 3 as applied to the problem of computing the value of a two-person matrix game with nonnegative coefficients is:

$$O\left(\sqrt{n \ln m}\left(\frac{1}{\delta} + \ln\ln n\right)\right).$$

This improves on the result in [22] (Algorithm 4.4), where the author gives the bound

$$O\left(\frac{\sqrt{n \ln m}}{\delta} \ln n\right).$$

### 2.4.4 Comparison of algorithms

We will conclude this section with a table comparing the complexities of the algorithms we have discussed:

| Method | Number of iterations | Work per iteration |
|---|---|---|
| **SubSearch** | $O(\frac{1}{\alpha^2\delta^2}\ln\frac{1}{\alpha})$ | $O(mn)$ |
| **SubBis** | $O(\frac{1}{\alpha^2\delta^2}+\frac{1}{\alpha^2}\ln\ln\frac{1}{\alpha})$ | $O(mn)$ |
| **SubSearchNR** | $O(\frac{1}{\alpha^2\delta^2}\ln\frac{1}{\alpha})$ | $O(mn)$ |
| **SubBisNR** | $O(\frac{1}{\alpha^2\delta^2}+\frac{1}{\alpha^2}\ln\frac{1}{\alpha})$ | $O(mn)$ |
| **SmoothBis** | $O(\frac{1}{\alpha\delta}+\frac{1}{\alpha}\ln\ln\frac{1}{\alpha})$ | $O(mn)$ |

Figure 2.5: Algorithms of Chapter 2.

Let us very briefly put the above results in perspective with the very popular interior-point methods (IPM) for convex optimization. While IPMs, in theory, need only $O(\ln(\frac{1}{\epsilon}))$ iterations to find a point within the (absolute) error $\epsilon$ of the optimum, each iteration is considerably more expensive because of the need to work with second-order information. In this sense, the fastest methods presented in this chapter are promising for problems where the desired accuracy is not too high, and the dimension of the problem is huge so that performing even a single iteration of an IPM is impossible.

Finally, let us note that the computation of an ellipsoidal rounding of the set $\mathcal{Q} := \mathrm{conv}\{\pm a_i \ : \ i = 1, 2, \ldots, m\}$ of quality $\frac{1}{\alpha} = O(\sqrt{n})$ can be performed in $O(n^2 m \ln m)$ arithmetic operations. Efficient rounding algorithms can be found in [22], [32], [16], [33] and [1]. See Algorithm 8 from the next section.

## 2.5 Combining the rounding and subgradient phases

In designing the algorithms of this chapter we have assumed the *availability* of a *good* ellipsoidal rounding of a certain convex and compact body containing the origin in its interior. As a quick introduction into the topic we have merely stated the celebrated theorem of John (Proposition 2.1.3) guaranteeing the *existence* of a rounding of certain quality which depends on the dimension of the underlying space and on the symmetry properties of the set.

We start in Subsection 2.5.1 by describing the details of a generic algorithm for rounding a centrally symmetric convex body $\mathcal{Q}$. The discussion will lead us to the observation that under certain conditions, the rounding algorithm can be viewed as performing optimization steps for a particular sublinear function – the support function of $\mathcal{Q}$. This leads to the idea of combining the rounding and optimization phases, as opposed to strictly adhering to the *round-first-optimize-later* strategy employed in the previous sections.

In Subsection 2.5.4 we describe an algorithm of this type and prove its convergence to the optimum. Although the complexity guarantee is nowhere near as good as the bounds obtained in the previous sections, the basic idea can be refined and leads to the development of Chapter 3 where we give algorithms with comparable provable convergence speeds.

### 2.5.1 Khachiyan's ellipsoidal rounding algorithm

In this part we describe a simplified version due to Nesterov [22] of a rounding algorithm of Khachiyan [15] applied to a centrally symmetric convex set $\mathcal{Q} \subset \mathbf{E}^*$. Let us note at this point that the general (not centrally symmetric) case can

be solved by rewriting it into a related centrally symmetric problem in a setting of one dimension higher. For more information on ellipsoidal rounding and the intimately related problem of finding the minimum volume enclosing ellipsoid we refer the reader to [32], [16], [33], [37] and [1].

**The setup**

Let $a_1, \ldots, a_m \in \mathbf{E}^*$ and consider

$$\mathcal{Q} := \operatorname{conv}\{\pm a_i \ : \ i = 1, 2, \ldots, m\}.$$

We will assume that the vectors $a_1, \ldots, a_m$ span $\mathbf{E}^*$, in which case $\mathcal{Q}$ is full-dimensional. Note that $\mathcal{Q} = \partial\varphi(0)$ where $\varphi$ is the support function of $\mathcal{Q}$ (see Example 2.3.3):

$$\varphi(x) := \max_g\{\langle g, x\rangle \ : \ g \in \mathcal{Q}\} = \max_i\{|\langle a_i, x\rangle| \ : \ i = 1, 2, \ldots, m\}. \tag{2.34}$$

The next two lemmas are Nesterov's restatements of Khachiyan's results. Let us start by noting that there is a readily available pair of central ellipsoids which give a $\sqrt{m}$-rounding of $\mathcal{Q}$:

**Lemma 2.5.1.** *If we let* $U_0 := \frac{1}{m}\sum_i a_i a_i^*$ *then*

$$\mathcal{B}(U_0, 1) \subseteq \mathcal{Q} \subseteq \mathcal{B}(U_0, \sqrt{m}).$$

*Proof.* The proof we give is due to Nesterov [22]. Since $\varphi(x) = \max_g\{\langle g, x\rangle \ : \ g \in \mathcal{Q}\}$, the following inequalities imply the result:

$$\max_{g \in \mathcal{B}(U_0, 1)} \langle g, x\rangle = \|x\|_{U_0} = \left(\frac{1}{m}\sum_{i=1}^m \langle a_i, x\rangle^2\right)^{1/2} \leq \max_{1 \leq i \leq m} |\langle a_i, x\rangle| = \varphi(x),$$

and

$$\max_{g \in \mathcal{B}(U_0, \sqrt{m})} \langle g, x\rangle = \sqrt{m}\|x\|_{U_0} = \sqrt{m}\left(\frac{1}{m}\sum_{i=1}^m \langle a_i, x\rangle^2\right)^{1/2} = \left(\sum_{i=1}^m \langle a_i, x\rangle^2\right)^{1/2} \geq \varphi(x).$$

□

The following lemma is the central result motivating the algorithm:

**Lemma 2.5.2** (Nesterov [22], Lemma 1)**.** *For a positive definite operator* $U : \mathbf{E} \to \mathbf{E}^*$ *and arbitrary* $g \in \mathbf{E}^*$ *let*

$$\mathcal{G}(U, g) := \operatorname{conv}\{\mathcal{B}(U), \pm g\};$$

*the convex hull of the ellipsoid* $\mathcal{B}(U) := \mathcal{B}(U, 1)$ *and the set* $\{\pm g\}$*. If for arbitrary* $0 \le \lambda \le 1$ *we denote* $U(\lambda) := (1 - \lambda)U + \lambda gg^*$*, then (see Figure 2.6)*

$$\mathcal{B}(U(\lambda)) \subseteq \mathcal{G}(U, g).$$

*If* $\sigma := \frac{1}{n}(\|g\|_U^*)^2 - 1 > 0$*, then the function*

$$V(\lambda) := \ln \frac{\det U(\lambda)}{\det U(0)} = \ln(1 + \lambda(n(1 + \sigma) - 1)) + (n - 1)\ln(1 - \lambda)$$

*is maximized at* $\lambda^* := \frac{\sigma}{n(1+\sigma)-1}$ *with* $V(\lambda^*) \ge \ln(1 + \sigma) - \frac{\sigma}{1+\sigma} \ge \frac{\sigma^2}{2(1+\sigma)^2}$.

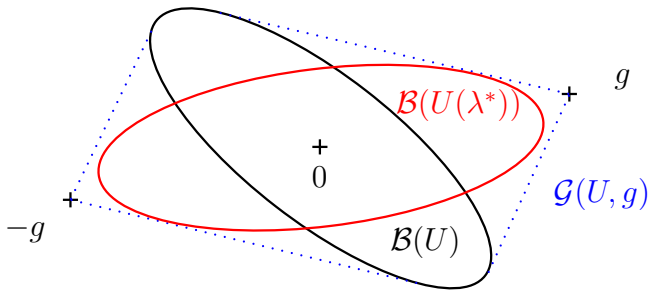



Figure 2.6: A single step of Khachiyan's ellipsoidal rounding algorithm.

### The algorithm

The above result is used in an algorithm as follows. We start with the rounding given by $U_0$ as described in Lemma 2.5.1. At each iteration, we choose $g = a_j$ so

that the volume of the new ellipsoid $\mathcal{B}(U(\lambda^*))$ is as large as possible. Lemma 2.5.2 guarantees that the new ellipsoid is contained in $\mathcal{G}(U, g)$, which in turn is a subset of $\mathcal{Q}$, by induction. Therefore, all ellipsoids constructed by the algorithm satisfy

$$\mathcal{B}(U, 1) \subseteq \mathcal{Q}. \tag{2.35}$$

Function $V$ is (proportional to) the logarithm of the ratio of the volumes of the new and old ellipsoids. Since it is increasing in $\sigma$, we choose $a_j$ so as to make $\sigma$ as large as possible:

$$j = \arg \max_{1 \leq i \leq m} \|a_i\|_U^*. \tag{2.36}$$

The crucial observation is that if there is $i$ for which

$$\|a_i\|_U^* \geq \gamma \sqrt{n} \tag{2.37}$$

for some arbitrary but fixed parameter $\gamma > 1$, then $V(\lambda^*)$ is bounded below by a positive constant, which then implies that the volume of the new ellipsoid increases by a constant fraction depending on $\gamma$. In view of Lemma 2.5.1, this leads to an upper bound on the number of steps. The algorithm terminates when (2.37) can not be satisfied by any $i$, which means that

$$a_i \in \mathcal{B}(U, \gamma \sqrt{n}) \quad \forall i,$$

which in turn implies

$$\mathcal{Q} \subseteq \mathcal{B}(U, \gamma \sqrt{n}). \tag{2.38}$$

The inclusions (2.35) and (2.38) imply that we have obtained a $\gamma\sqrt{n}$-rounding of $\mathcal{Q}$. This informal discussion leads to Algorithm 8 whose theoretical performance is described in Theorem 2.5.3.

This result is originally due to Khachiyan [15], while the simplified analysis described above, due to Nesterov [22], serves the purpose of motivating the central

discussion of this section. An efficient implementation updates $U^{-1}$, or a Cholesky factorization of $U$, and hence the quantities $\|a_i\|_U^*$, in $O(mn)$ arithmetic operations per iteration.

**Algorithm 8 (EllipsRound)** Khachiyan's ellipsoidal rounding algorithm.

1: **Input:** $a_1, \ldots, a_m; \gamma > 1$;

2: $k = 0,\ U_0 = \frac{1}{m}\sum_i a_i a_i^*$;

3: $j = \arg\max_i\{\|a_i\|_{U_k}^* \ :\ i = 1, 2, \ldots, m\},\ g_k = a_j,\ \rho_k = \|g_k\|_{U_k}^*$;

4: **while** $\rho_k > \gamma\sqrt{n}$ **do**

5: $\quad \lambda_k = \frac{1}{n}\frac{\rho_k^2 - n}{\rho_k^2 - 1},\ U_{k+1} = (1 - \lambda_k)U_k + \lambda_k g_k g_k^*$;

6: $\quad k = k + 1$;

7: $\quad j = \arg\max_i\{\|a_i\|_{U_k}^* \ :\ i = 1, 2, \ldots, m\},\ g_k = a_j,\ \rho_k = \|g_k\|_{U_k}^*$;

8: **end while**

9: **Output:** $U_k$

**Theorem 2.5.3** (Nesterov [22], Theorem 1)**.** *Algorithm 8 produces a $\gamma\sqrt{n}$-rounding of $\mathcal{Q}$ and terminates in at most*

$$\frac{n \ln m}{2 \ln \gamma - 1 + \gamma^{-2}}$$

*iterations.*

*Proof.* The termination criterion of the algorithm is equivalent to $\sigma_k := \frac{1}{n}\rho_k^2 - 1 < \gamma^2 - 1$. So if the method is still running at iteration $k$, Lemma 2.5.2 implies that

$$\ln \frac{\det U_{k+1}}{\det U_k} \geq \ln(1 + \sigma_k) - \frac{\sigma_k}{1 + \sigma_k} \geq 2 \ln \gamma - \frac{\gamma^2 - 1}{\gamma^2},$$

which gives a positive lower bound on $V$. Now since the volume of $\mathcal{B}(U, 1)$ is proportional to $\det U^{1/2}$, we obtain

$$\frac{\det U_k^{1/2}}{\det U_0^{1/2}} = \frac{\operatorname{vol} \mathcal{B}(U_k)}{\operatorname{vol} \mathcal{B}(U_0)} \leq \frac{\operatorname{vol} \mathcal{Q}}{\operatorname{vol} \mathcal{B}(U_0)} \leq \frac{\operatorname{vol} \mathcal{B}(U_0, m^{1/2})}{\operatorname{vol} \mathcal{B}(U_0)} = m^{n/2}.$$

To obtain the iteration bound it remains to compare the two displayed inequalities using the fact that for a positive definite matrix we have $\det U^{1/2} = (\det U)^{1/2}$. The rounding guarantee follows from the termination criterion. □

### 2.5.2 Preliminaries

In this and the following two subsections we consider problem $(P)$ with objective function as in (2.34) and a simple affine feasibility set given by a nonzero vector $d \in \mathbf{E}^*$:

$$\boxed{\varphi^* := \min_x \{\varphi(x) \equiv \max_i |\langle a_i, x\rangle| \ : \ \langle d, x\rangle = 1\}.} \qquad (P1)$$

We continue to assume that the vectors $a_1, \ldots, a_m$ span $\mathbf{E}^*$. This problem is the starting point of the development of Chapter 3 and we also call it $(P1)$ there.

#### Updated projection and the direction of the negative subgradient

Let $x_0$ be as usual — the projection of the origin onto the feasibility set. Suppose $U : \mathbf{E} \to \mathbf{E}^*$ is the positive definite operator coming from a rounding procedure for $\mathcal{Q}$, and let $\mathbf{E}$ be equipped with the norm $\|\cdot\|_U$ and $\mathbf{E}^*$ with the dual norm $\|\cdot\|_U^*$. Notice that in this setting we have

$$x_0 = \frac{U^{-1}d}{(\|d\|_U^*)^2}. \qquad (2.39)$$

As we have seen, a generic step of a rounding algorithm performs the following update:

$$U_+ := (1-\lambda)U + \lambda g g^*.$$

If $0 < \lambda < 1$, then by the Sherman-Morrison formula (see, for example, [10] or [38]), the updated operator is invertible and its inverse is given by

$$U_+^{-1} = \frac{1}{1-\lambda}\left(U^{-1} - \frac{\lambda U^{-1} g g^* U^{-1}}{1-\lambda+\lambda(\|g\|_U^*)^2}\right). \qquad (2.40)$$

Notice that the second denominator vanishes for a single *negative* value of $\lambda$ and hence the expression is well-defined. Using (2.40) we can compute the updated version of $x_0$:

$$x_0^+ = \frac{U_+^{-1}d}{(\|d\|_{U_+}^*)^2} = \frac{U^{-1}d - \frac{\lambda}{\kappa}\langle d, U^{-1}g\rangle U^{-1}g}{\langle d, U^{-1}d\rangle - \frac{\lambda}{\kappa}\langle d, U^{-1}g\rangle^2} = \frac{p-q}{r-s},$$

where

$$\kappa = 1 - \lambda + \lambda(\|g\|_U^*)^2$$

and

$$p = U^{-1}d, \quad q = \tfrac{\lambda}{\kappa}\langle d, U^{-1}g\rangle U^{-1}g, \quad r = \langle d, p\rangle, \quad s = \langle d, q\rangle.$$

Assume now that

$$\langle d, U^{-1}g\rangle = 0 \tag{2.41}$$

and notice that then $q = 0$ and $s = 0$ and, in turn, $x_0^+ = p/r = x_0$. Now if (2.41) does not hold, we may write

$$x_0^+ = \frac{p-q}{r-s} = \frac{p}{r} + \frac{s}{r-s}\left(\frac{p}{r} - \frac{q}{s}\right).$$

This is useful because one can easily verify that $\frac{p}{r} = x_0 \in \mathcal{L}$ and $\frac{q}{s} \in \mathcal{L}$ and hence the step leading from $x_0$ to $x_0^+$ is

$$h_1 := x_0^+ - x_0 = \frac{s}{r-s}\left(\frac{U^{-1}d}{\langle d, U^{-1}d\rangle} - \frac{U^{-1}g}{\langle d, U^{-1}g\rangle}\right). \tag{2.42}$$

Note that under the assumption that (2.41) fails, $h_1$ is zero if and only if $d$ and $g$ are collinear. We have obtained the following result:

**Lemma 2.5.4.** *$x_0^+ = x_0$ if and only if either $\langle d, U^{-1}g\rangle = 0$ ($d$ and $g$ are orthogonal under the inner product defined by $U^{-1}$) or $d$ and $g$ are collinear.*

Our next result asserts that if we choose $g$ to be a subgradient of $\varphi$ at $x_0$, then $x_0^+$ can be interpreted as a point in the direction of the negative subgradient of $\varphi$ restricted to $\mathcal{L}$ taken at $x_0$.

**Proposition 2.5.5.** *If* $\langle d, U^{-1}g\rangle \neq 0$ *and* $g \in \partial\varphi(x_0)$*, then*

$$x_0^+ - x_0 = -\beta \frac{h}{\|h\|_U},$$

*where* $h \in \partial_U\varphi|_{\mathcal{L}}(x_0)$ *and*

$$\beta = \frac{s\|h\|_U}{(r-s)\langle d, U^{-1}g\rangle}. \tag{2.43}$$

*Proof.* By (2.6) we have $U^{-1}g \in \partial\varphi_U(x_0)$. It can be easily verified from the definition of the subgradient that to obtain $h$ as specified it suffices to project $U^{-1}g$ onto $\{x \in \mathbf{E} \ : \ \langle d, x\rangle = 0\}$ (in the inner product defined by $U$). The projection formula is also easy to derive. Since point $\hat{x} \in \mathbf{E}$ gets mapped to $\hat{x}|_{\mathcal{L}} := \hat{x}(\mu) = \hat{x} + \mu U^{-1}d$ such that $\langle d, \hat{x}(\mu)\rangle = 0$, it follows that $\mu = -\langle d, \hat{x}\rangle/(\|d\|_U^*)^2$ and finally $\hat{x}|_{\mathcal{L}} = \hat{x} - \langle d, \hat{x}\rangle x_0$. Therefore,

$$\begin{aligned} h &:= U^{-1}g - \langle d, U^{-1}g\rangle \frac{U^{-1}d}{\langle d, U^{-1}d\rangle} \\ &= \langle d, U^{-1}g\rangle \left[\frac{U^{-1}g}{\langle d, U^{-1}g\rangle} - \frac{U^{-1}d}{\langle d, U^{-1}d\rangle}\right] \in \partial_U\varphi|_{\mathcal{L}}(x_0). \end{aligned}$$

We see by looking at (2.42) that the vectors $h$ and $h_1 = x_0^+ - x_0$ are collinear. A straightforward calculation gives (2.43). Also note that

$$\|h\|_U^2 = \langle g, U^{-1}g\rangle - \frac{\langle d, U^{-1}g\rangle^2}{\langle d, U^{-1}d\rangle}.$$

□

### 2.5.3 Properties of a general rounding sequence

In this subsection we investigate the rounding properties of a sequence of ellipsoids generated by a process slightly more general than the one used in Khachiyan's rounding algorithm. Let us start with a formal definition of the concept:

**Definition 2.5.6.** Let $\mathcal{Q} \subset \mathbf{E}^*$ be an arbitrary centrally symmetric convex body. We call $(U_k, g_k, \lambda_k)_0^{K-1}$ a *rounding sequence* for $\mathcal{Q}$ with parameters $R > 0$ and $\gamma > 1$ if the following properties are satisfied:

1. $\mathcal{B}(U_0, 1) \subseteq \mathcal{Q} \subseteq \mathcal{B}(U_0, R)$,

2. $g_k \in \mathcal{Q}$ and $\|g_k\|_{U_k}^* > \gamma\sqrt{n}$, and

3. $U_{k+1} = (1 - \lambda_k)U_k + \lambda_k g_k g_k^*$ for all $k = 0, 1, \ldots, K-1$.

If the update parameters $\lambda_k$ are chosen in accordance with Step 5 of Algorithm 8, we will refer to the object as an *optimal rounding sequence*. If, moreover, the vectors $g_k$ are chosen as in Step 3 of Algorithm 8, we will use the term *Khachiyan's rounding sequence*.

Notice that it follows from the proof of Theorem 2.5.3 that optimal rounding sequences can not be too long:

$$K \leq \frac{2n \ln R}{\gamma^2 - 1 + 2\ln\gamma}. \tag{2.44}$$

Note that Theorem 2.5.3 can be reformulated to handle also non-polyhedral sets (although performing Step 3 of Algorithm 8 becomes tricky). In the language of the definition above, this theorem says that maximal Khachiyan's rounding sequences terminate with $\mathcal{B}(U_K, 1) \subseteq \mathcal{Q} \subseteq \mathcal{B}(U_K, \gamma\sqrt{n})$. It is not clear how to extend the argument leading to this conclusion to also guarantee certain rounding properties of the intermediary ellipsoids generated in the process. In fact, while the method seeks to greedily maximize the volume of the next iterate ellipsoid, the rounding quality of the iterates will in general not be monotonically improving. Let us illustrate this with an example.

**Example 2.5.7.** Let $I$ be the $2 \times 2$ identity matrix and let $\mathcal{Q} = 2\mathcal{B}(I)$ – the ball of radius 2 in $\mathbf{R}^2$. The matrix $U := I$ defines a 2-rounding of $\mathcal{Q}$ since $\mathcal{B}(U) \subseteq \mathcal{Q} \subseteq 2\mathcal{B}(U)$. Now consider updating $U$ to $U_+ = U(\lambda)$ with $g = (2, 0)$ and $\lambda$ as in Step 5 of Algorithm 8:

$$\lambda = \frac{\frac{1}{2}\|g\|_U^{*\,2} - 1}{\|g\|_U^{*\,2} - 1} = \frac{\frac{1}{2}4 - 1}{4 - 1} = \frac{1}{3}.$$

We get $U_+ = (1-\lambda)U + \lambda g g^* = \left(\begin{smallmatrix} 3\lambda+1 & 0 \\ 0 & 1-\lambda \end{smallmatrix}\right) = \left(\begin{smallmatrix} 2 & 0 \\ 0 & 2/3 \end{smallmatrix}\right)$. The updated ellipsoid $\mathcal{B}(U_+)$ is axis-aligned, with axes lengths equal to the square roots of the diagonal elements of $U_+$ (see Figure 2.7). Note that the new ellipsoid, although of a larger volume, has a *worse* rounding capability. Indeed, we have

$$\mathcal{B}(U_+) \subseteq \mathcal{Q} \subseteq 2\mathcal{B}(U) \subseteq 2\frac{1}{\sqrt{1-\lambda}}\mathcal{B}(U_+),$$

and hence $U_+$ generates a $2\sqrt{\frac{3}{2}}$-rounding of $\mathcal{Q}$ (see Lemma 2.5.8). Note also that the equality $\|g\|_{U_+}^* = \frac{2}{\sqrt{2}} = \sqrt{2} = \sqrt{n}$ is not a coincidence (Lemma 2.5.14).

We will now analyze the rounding behavior of optimal rounding sequences. We give a simple bound on the measure of deterioration of the rounding quality of successive iterate ellipsoids, which leads to the conclusion that all ellipsoids corresponding to such a sequence produce at worst something slightly weaker than a $m$-rounding of $\mathcal{Q}$. This gives us a tool for the analysis of methods which would attempt to combine the rounding and optimization phases by choosing the vector $g_k$ in a different manner from Step 3 of Algorithm 8: perhaps choosing $g_k$ to be the subgradient of the support function of $\mathcal{Q}$ at the current point $x_0$.

**Lemma 2.5.8.** *For any positive definite self-adjoint operator $U : \mathbf{E} \to \mathbf{E}^*$, $g \in \mathbf{E}^*$ and $\lambda \in [0, 1)$ we have*

$$\mathcal{B}(U) \subseteq \frac{1}{\sqrt{1-\lambda}}\mathcal{B}(U(\lambda)). \tag{2.45}$$

*This multiplicative factor is the best possible.*

*Proof.* Recall that $U(\lambda) = (1-\lambda)U + \lambda gg^*$ and let $U_+ = U(\lambda)$. Then for any $h \in \mathbf{E}^*$ the Sherman-Morrison formula (2.40) implies

$$\begin{aligned}
\|h\|_{U_+}^* &= \langle h, U_+^{-1} h\rangle^{1/2} \\
&= \left\langle h, \frac{1}{1-\lambda}\left(U^{-1} - \frac{\lambda}{\kappa} U^{-1} g g^* U^{-1}\right) h \right\rangle^{1/2} \\
&= \frac{1}{\sqrt{1-\lambda}}\left[(\|h\|_U^*)^2 - \frac{\lambda}{\kappa}\langle h, U^{-1} g\rangle^2\right]^{1/2} \\
&\leq \frac{1}{\sqrt{1-\lambda}} \|h\|_U^*,
\end{aligned}$$

where the last step holds because $\kappa = 1 - \lambda + \lambda(\|g\|_U^*)^2 > 0$. The inclusion is tight since we can choose $h$ with $\|h\|_U^* = 1$ and $\langle h, U^{-1} g\rangle = 0$ and hence $\|h\|_{U_+}^* = \frac{1}{\sqrt{1-\lambda}}\|h\|_U^* = \frac{1}{\sqrt{1-\lambda}}$ (see Example 2.5.7 and Figure 2.7 for illustration of this). ☐

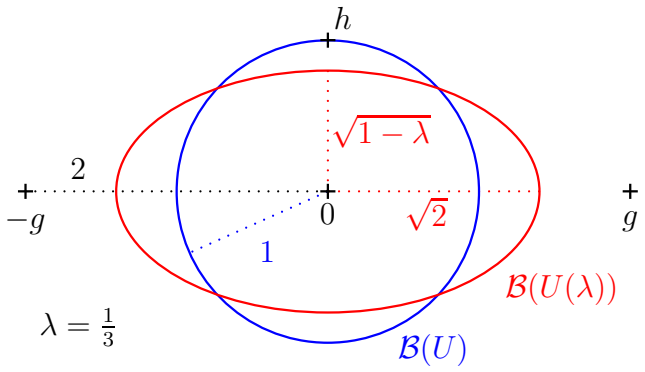



Figure 2.7: Illustration of Lemma 2.5.8.

**Corollary 2.5.9.** *If $\|g\|_U^* > \sqrt{n}$ and we choose $\lambda = \lambda^*$ (as in Step 5 of Algorithm 8), then*

$$\mathcal{B}(U) \subseteq \sqrt{\frac{n}{n-1}}\mathcal{B}(U(\lambda)).$$

*Proof.* Notice that $0 < \lambda < \frac{1}{n}$ and hence $\frac{1}{\sqrt{1-\lambda}} \leq \sqrt{\frac{n}{n-1}}$. ☐

This is the main result of this subsection:

**Proposition 2.5.10.** *If $(U_k, g_k, \lambda_k)$, $k \geq 0$, is a optimal rounding sequence with parameters $R = \sqrt{m}$ and $\gamma > 1$, then*

$$\mathcal{B}(U_k) \subset \mathcal{Q} \subset m^{1/2} \left(\frac{n}{n-1}\right)^{k/2} \mathcal{B}(U_k) \subset m^{\beta} \mathcal{B}(U_k), \quad k \geq 0,$$

*where*

$$\beta := \frac{1}{2} + \frac{n}{2(n-1)(\gamma^{-2} - 1 + 2\ln\gamma)}.$$

*Proof.* First notice that $\mathcal{B}(U_0) \subseteq \mathcal{Q} \subseteq m^{1/2}\mathcal{B}(U_0)$ because $R = \sqrt{m}$ (see Lemma 2.5.1 for a possible choice of $U_0$ satisfying this). The first inclusion follows by induction from Lemma 2.5.2, the second by induction from Corollary 2.5.9. Finally, inequality (2.44) implies

$$\begin{aligned}\left(\frac{n}{n-1}\right)^{k/2} \leq \left(\frac{n}{n-1}\right)^{K/2} &\leq \left(1 + \frac{1}{n-1}\right)^{(n-1)\frac{n}{n-1}\frac{\ln\sqrt{m}}{\gamma^{-2}-1+2\ln\gamma}} \\ &\leq \exp\left\{\frac{n}{2(n-1)}\frac{\ln m}{\gamma^{-2} - 1 + 2\ln\gamma}\right\},\end{aligned}$$

establishing the last inclusion. □

**Remark 2.5.11.** *Note that if we choose $\gamma$ such that $\gamma^{-2} - 1 + 2\ln\gamma = 1$ ($\gamma \approx 2.511$) then $\beta \approx 1$ for large $n$ and hence any ellipsoid generated by a rounding sequence of this type is guaranteed to produce at worst something only slightly weaker than an $m$-rounding of $\mathcal{Q}$.*

### 2.5.4 Alternating rounding and subgradient steps

The discussion in the previous subsections has brought to light certain connections between the rounding and (subgradient) optimization phases which, as we have seen, are completely split in the approach of Section 2.2. In Subsection 2.5.2 we have shown that a single generic step of the rounding procedure with a specific

choice of the update vector $g$ ($g \in \partial\varphi(x_0)$) corresponds to taking a step from $x_0$ in the direction of the negative subgradient of $\varphi$ at that point. This observation raises the question of whether it is possible to *alternate* the rounding and optimization steps, combining the two previously separate phases into a single convergent algorithmic scheme. Another reason for trying to combine the two phases is the fact that in certain circumstances the arithmetical complexity of the rounding algorithm phase ($O(n^2 m \ln m)$, see [22]) may be the dominant computational burden. Let us describe several possible approaches:

**Approach 1 - Primarily rounding**

In the rounding step of this approach we always take $g = a_j$ with $j$ defined as in (2.36) and then update $U$ to $U_+ = U(\lambda)$ with $\lambda = \lambda^*$. This means that we perform a rounding step exactly as in Algorithm 8.

In the optimization step we first compute $x_0$ – our primal iterate – and decide whether or not to take a subgradient step (or a sequence of such steps) from this point. A reasonable criterion for this decision could be the size of the subgradient. For example, if $\hat{g} \in \partial\varphi(x_0)$ and $\|\hat{g}\|_U^* < \gamma\sqrt{n}$, then the subgradient is well-rounded (by $U$) and "hence" there is no need to take a rounding step. We may take one subgradient step or a fixed number of such steps or perhaps continue until we attain approximate optimality or encounter a point with a large subgradient. In the latter case the procedure gets "restarted" by taking a rounding step and starting everything again from the new $x_0$.

This approach primarily concentrates on performing the rounding with the hope to obtain some good primal iterates along the way by taking subgradient steps starting from the projection points $x_0$.

**Approach 2**

In the rounding step of this variation on the combine-the-two-phases theme we always take $g \in \partial\varphi(x_0)$, motivated by Proposition 2.5.5. Matrix $U$ then gets updated to $U_+ = U(\lambda^*)$. Hence we perform a rounding step as in Algorithm 8 with the exception that we are *not* following the greedy strategy of trying to maximize the volume of the new ellipsoid. Instead, we try to *combine* the rounding and optimization steps into a single step which can be interpreted as performing *both* rounding and optimization work.

This approach is still slightly a rounding-oriented one because of the choice of the "line-search" parameter $\lambda$. Although the primal steps (in **E**) are taken in the direction of the negative subgradient, the steplengths are determined by the desire to maximize the volume of the next ellipsoid, given the choice of $g$.

A variation on this theme would be to shift the emphasis to the optimization routine by allowing rounding steps only if $\|g\|_U^* > \gamma\sqrt{n}$ and performing a fixed number of subgradient steps starting from the current point $x_0$. See Algorithm 9.

**Theorem 2.5.12.** *Algorithm 9 outputs a $\delta$-approximate solution to* $(P1)$.

*Proof.* At the $k$-th call of the subgradient method the quantity $\frac{1}{\alpha_k}$ represents the rounding quality of $\mathcal{B}(U_k)$. Notice that the number of steps of the subgradient subroutine is chosen precisely so that the method outputs a $\delta$-approximate minimizer of $(P1)$, provided that it takes all the prescribed steps and is not stopped by the condition on the size of the subgradient. However, since $(U_k, g_k, \lambda_k), k \geq 0$, forms an optimal rounding sequence with parameters $R = \sqrt{m}$ and $\gamma$, all subgradients will be small enough, in the then-current norm, when the sequence reaches maximality. This happens at most after a finite number of iterations given in

**Algorithm 9 (SubRound)** Rounding while optimizing

**Input:** $a_1, \ldots, a_m, d, \gamma > 1, \delta$;

$U_0 = \frac{1}{m}\sum_i a_i a_i^*,\ \alpha_0 = \frac{1}{\max_i \|a_i\|^*_{U_0}},\ x_0 = \frac{{U_0}^{-1}d}{(\|d\|^*_{U_0})^2}; k = 0;$

**OPTIMIZE:**

$N = \lfloor \frac{1}{\alpha_k^4 \delta^2} \rfloor;$

$x = \textbf{Subgrad}(\varphi, \mathcal{L} = \{x \ :\ \langle d, x\rangle = 1\}, x_0, \varphi(x_0), N);$

Stop the execution of the subroutine if a large subgradient is encountered

$(\|g\|^*_{U_k} > \gamma\sqrt{n})$, otherwise **exit**;

Set $g_k = g$ and proceed with the rounding phase;

**ROUND:**

$\lambda_k = \frac{1}{n}\frac{(\|g_k\|^*_U)^2 - n}{(\|g_k\|^*_U)^2 - 1},\ U_{k+1} = (1-\lambda_k)U_k + \lambda_k g_k g_k^*;$

$\alpha_{k+1} = \frac{1}{\max_i \|a_i\|^*_{U_{k+1}}};$

$x_{k+1} = \frac{{U_{k+1}}^{-1}d}{(\|d\|^*_{U_{k+1}})^2};$

$k = k + 1;$

proceed with the optimization phase;

**Output:** $x$

(2.44). □

Note that Algorithm 9, as stated, has worse guaranteed performance than a scheme which would run the subgradient subroutine a single time with the "available-but-bad" upper bound (see Subsection 2.2.2). Several modifications are desired to make this into a more practical algorithm. For example, one could allow for variable step-lengths in the subgradient subroutine, introduce nonrestarting behavior, etc. However, it is possible that at least for some problem instances the subgradient subroutine will encounter a large subgradient early, avoiding the need to take the prescribed number of steps. We do not know how to obtain a simple modification of the algorithm which would guarantee performance comparable to any of the methods discussed before. There is one approach leading to a $O(1/\delta)$ algorithm, but it involves radical changes in the rounding sequence away from actually trying to round $\mathcal{Q}$ and towards aiming to round the crucial part of this set — its intersection with the line passing through the origin and the vector $d$. Chapter 3 is devoted to the development of an algorithm of this type.

**Approach 3 - Primarily optimization**

Consider taking $g \in \partial\varphi(x_0)$ at every iteration and choosing $\lambda$ greedily from the optimization perspective. It is not obvious how one would go about defining "the optimization viewpoint" and construct details of an algorithm of this type.

In Chapter 3 we give a $O(1/\delta)$ algorithm for $(P1)$ that can be understood as adhering to this approach. Let us sketch some details of how this will be done. First notice that if we let $j := \arg\max_i |\langle a_i, x_0\rangle|$, then $a_j \in \partial\varphi(x_0)$, where $\varphi$ is the objective function from problem $(P1)$. We can therefore choose $g = a_j$. Also observe that because $x_0$ and $U^{-1}d$ are proportional, we could have equally well

defined $j$ via $j = \arg\max_i |\langle a_i, U^{-1}d\rangle|$. The steplength $\lambda$ will chosen so as to minimize the value of $\|d\|^*_{U_+}$. It can be shown that this is equivalent to choosing $\lambda$ so that the $U_+$-norm of $x_0^+$ is as small as possible. We will explain the reasoning behind this choice in Chapter 3.

### 2.5.5 Rounding the observed part of a set

We have seen that every ellipsoid of an optimal rounding sequence with $R = m^{1/2}$ produces a rounding of $\mathcal{Q}$ of quality somewhere between $m^{1/2}$ and $m^\beta \approx m$ (Proposition 2.5.10). If we do not assume that the vectors $g_k$ are chosen in accordance with some clever strategy (as, for example, in Algorithm 8), it seems that the deteriorating nature of the rounding bounds is necessary. In the definition of a rounding sequence we are abstracting from the process of selecting the points $g_k$. Perhaps there is a subroutine which is providing us with vectors $g_k \in \mathcal{Q}$ of sufficiently large norms ($\|g_k\|^*_{U_k} > \gamma\sqrt{n}$). If such vectors do not exist, then, of course, the rounding sequence terminates with a $\gamma\sqrt{n}$-rounding of $\mathcal{Q}$. However, we will assume here that either there is no global oracle available to tell us whether such points exist (and hence we do not have first-hand information on the quality of the rounding given by the current iterate of the rounding sequence), or that the points $g_k$ are produced by some external process which may, for reasons of its internal structure, fail to yield another point, even though globally such points might exist. As an example of the latter situation think of running the subgradient algorithm for $(P1)$ and choosing $g_k$ to be the subgradients of the iterates. While it may very well happen that this external process fails to output a large enough subgradient, this does not mean that points of large norm do not exist in $\mathcal{Q}$.

Due to the assumed local behavior of the process generating the points $g_k$, we

will concentrate on a local result by asking the following question: How well does a rounding sequence perform when it comes to rounding the portion of $\mathcal{Q}$ "seen so far" by it? Let us start with a definition clarifying this concept:

**Definition 2.5.13.** For a rounding sequence $(U_k, g_k, \lambda_k),\ k \geq 0$, we define

$$\mathcal{Q}_k := \text{conv}\{\mathcal{B}(U_0), \pm g_0, \ldots, \pm g_{k-1}\}, \qquad k \geq 1.$$

The set $\mathcal{Q}_k$ represents the best model of $\mathcal{Q}$ at a particular time. In other words, this is the portion of $\mathcal{Q}$ as seen by the above rounding sequence at iteration $k$. We now proceed to prove that a rounding sequence does a much better job at rounding $\mathcal{Q}_k$ than $\mathcal{Q}$. Let us start with a couple of intermediary results.

**Lemma 2.5.14.** *If $\|g\|_U^* > \sqrt{n}$ and $\lambda = \lambda^*$ (as in Corollary 2.5.9), then*

$$\|g\|_{U_+}^* = \sqrt{n}.$$

*Proof.* We proceed similarly as in Lemma 2.5.8:

$$\begin{aligned}
\|g\|_{U_+}^* &= \frac{1}{\sqrt{1-\lambda}} \left[ {\|g\|_U^*}^2 - \frac{\lambda}{\kappa} {\|g\|_U^*}^4 \right]^{1/2} \\
&= \frac{\|g\|_U^*}{\sqrt{1 - \frac{\frac{1}{n}{\|g\|_U^*}^2 - 1}{{\|g\|_U^*}^2 - 1}}} \left[ 1 - \frac{\frac{\frac{1}{n}{\|g\|_U^*}^2 - 1}{{\|g\|_U^*}^2 - 1}}{\frac{1}{n}{\|g\|_U^*}^2} {\|g\|_U^*}^2 \right]^{1/2} \\
&= \sqrt{n}.
\end{aligned}$$

□

**Lemma 2.5.15.** *If $(U_k, g_k, \lambda_k),\ k \geq 0$, is a rounding sequence, then*

$$\cup_{i=0}^{k} \mathcal{B}(U_i) \subset \mathcal{Q}_k, \quad k \geq 1$$

*and hence*

$$\mathcal{Q}_k = \text{conv}\{\mathcal{B}(U_0), \ldots, \mathcal{B}(U_k), \pm g_0, \ldots, \pm g_{k-1}\}, \quad k \geq 1.$$

*Proof.* By the definition of $\mathcal{Q}_k$ we have $\mathcal{B}(U_0) \subset \mathcal{Q}_k$. By Lemma 2.5.2 we have $\mathcal{B}(U_1) \subset \mathrm{conv}\{\mathcal{B}(U_0), \pm g_0\}$ and hence $\mathcal{B}(U_1) \subset \mathcal{Q}_k$. The result follows by induction. □

The following is a local analogue of Proposition 2.5.10:

**Proposition 2.5.16.** *If* $(U_k, g_k, \lambda_k),\ k \geq 0$, *is an optimal rounding sequence, then*

$$\mathcal{B}(U_k) \subset \mathcal{Q}_k \subset \sqrt{n}\left(\frac{n}{n-1}\right)^{(k-1)/2} \mathcal{B}(U_k), \quad k \geq 1. \tag{2.46}$$

*Proof.* The first inclusion follows from Lemma 2.5.15. For the second inclusion, we will inductively use Corollary 2.5.9 and Lemma 2.5.14 which state that

$$\mathcal{B}(U_{k-1}) \subset \left(\frac{n}{n-1}\right)^{1/2} \mathcal{B}(U_k) \quad \text{and} \quad \{\pm g_{k-1}\} \subset \sqrt{n}\mathcal{B}(U_k).$$

Combining these two we get

$$\{\pm g_{k-2}\} \subset \sqrt{n}\mathcal{B}(U_{k-1}) \subset \sqrt{n}\left(\frac{n}{n-1}\right)^{1/2} \mathcal{B}(U_k).$$

It is easy to see that by induction we obtain $\{\pm g_0\} \subset \sqrt{n}\left(\frac{n}{n-1}\right)^{(k-1)/2} \mathcal{B}(U_k)$ and in turn

$$\{\pm g_0, \ldots, \pm g_{k-1}\} \subset \sqrt{n}\left(\frac{n}{n-1}\right)^{(k-1)/2} \mathcal{B}(U_k). \tag{2.47}$$

Also,

$$\mathcal{B}(U_0) \subset \left(\frac{n}{n-1}\right)^{k/2} \mathcal{B}(U_k) \subset \sqrt{n}\left(\frac{n}{n-1}\right)^{(k-1)/2} \mathcal{B}(U_k). \tag{2.48}$$

Finally note by taking the convex hull of the sets appearing at the left hand sides of the inclusions (2.47) and (2.48) we obtain $\mathcal{Q}_k$. The right hand side of both inclusions is the same and coincides with the expression in (2.46). □

**Bounding the support functions**

Recall that an ellipsoidal rounding of a convex set gives lower and upper bounds on the support function of that set (see (2.7)). Let us therefore define

$$\varphi(x) := \max_g \{\langle g, x\rangle \ : \ g \in \mathcal{Q}\}, \tag{2.49}$$

and

$$\varphi_{\mathcal{Q}_k}(x) := \max_g \{\langle g, x\rangle \ : \ g \in \mathcal{Q}_k\}, \tag{2.50}$$

and note that $\varphi_{\mathcal{Q}_k}(x) \leq \varphi(x)$ for all $x$ since $\mathcal{Q}_k \subseteq \mathcal{Q}$. Also observe that while Proposition 2.5.10 implies

$$\|x\|_{U_k} \leq \varphi(x) \leq \sqrt{m}\left(\frac{n}{n-1}\right)^{k/2} \|x\|_{U_k}, \quad k \geq 0, \tag{2.51}$$

Proposition 2.5.16 gives

$$\|x\|_{U_k} \leq \varphi_{\mathcal{Q}_k}(x) \leq \sqrt{n}\left(\frac{n}{n-1}\right)^{(k-1)/2} \|x\|_{U_k}, \quad k \geq 1,$$

which gives a better bound.

**A subgradient optimal rounding sequence**

Can we construct an inequality of the type

$$\varphi(x) \leq \beta_k \|x\|_{U_k},$$

with $\beta_k$ better than the constant in (2.51)? This might be possible to ensure, but it seems likely that we will have to be ready to make a sacrifice. Perhaps we should require the inequality to hold only for certain values of $x$. We show in the remainder of this subsection how this can be done.

Let us consider an optimal rounding sequence $(U_k, g_k, \lambda_k),\ k \geq 0$, with a very specific choice of the vectors $g_k$:

$$g_k \in \partial\varphi(x_k),$$

where $x_k,\ k \geq 0$, are some points in $\mathbf{E}$. Define

$$\mathcal{P}_k := \operatorname{conv}\{x_i \mid i = 0, \ldots, k-1\}, \quad k \geq 1$$

and

$$\operatorname{diam} \mathcal{P}_k := \max_{0 \leq i,j < k} \{\|x_i - x_j\|_{U_k}\}. \tag{2.52}$$

From now on, let us fix some arbitrary $k$ and consider $x \in \mathcal{P}_k$, assuming the following representation:

$$x = \sum_{i=0}^{k-1} w_i x_i, \quad \sum_{i=0}^{k-1} w_i = 1, \quad w_i \geq 0.$$

Notice that in the course of proving Proposition 2.5.16, we have essentially shown that

$$\{\pm g_i, \ldots, \pm g_{k-1}\} \subset \sqrt{n} \left(\frac{n}{n-1}\right)^{(k-i-1)/2} \mathcal{B}(U_k), \qquad 0 \leq i \leq k-1. \tag{2.53}$$

Taking the supremum of the linear functional $\langle \cdot, x_i \rangle$ over these sets, for any $0 \leq i \leq k-1$, we get

$$\begin{aligned} \varphi(x_i) &= |\langle g_i, x_i \rangle| \\ &\leq \sup\{\langle g, x_i \rangle \ : \ g \in \{\pm g_i, \ldots, \pm g_{k-1}\}\} \\ &\leq \sqrt{n} \left(\frac{n}{n-1}\right)^{(k-i-1)/2} \|x_i\|_{U_k}. \end{aligned}$$

Now using convexity of $\varphi$, (2.53), the triangle inequality and the definition of

diam $\mathcal{P}_k$,

$$\begin{aligned}
\varphi(x) &= \varphi\left(\sum_{i=0}^{k-1} w_i x_i\right) \\
&\leq \sum_{i=0}^{k-1} w_i \varphi(x_i) \\
&\leq \sum_{i=0}^{k-1} w_i \sqrt{n} \left(\frac{n}{n-1}\right)^{(k-i-1)/2} \|x_i\|_{U_k} \\
&\leq \sqrt{n} \sum_{i=0}^{k-1} w_i \left(\frac{n}{n-1}\right)^{(k-i-1)/2} (\|x\|_{U_k} + \|x_i - x\|_{U_k}) \\
&\leq \sqrt{n} \left[\sum_{i=0}^{k-1} w_i \left(\frac{n}{n-1}\right)^{(k-i-1)/2}\right] (\|x\|_{U_k} + \operatorname{diam} \mathcal{P}_k) \\
&\leq \sqrt{n} \left(\frac{n}{n-1}\right)^{(k-1)/2} (\|x\|_{U_k} + \operatorname{diam} \mathcal{P}_k).
\end{aligned}$$

Note that instead of using the more refined inequality (2.53), we could have directly used Proposition 2.5.16. However, the former could be useful when we need to analyze a particular point $x \in \mathcal{P}_k$ for which the weights $w_i$ grow (perhaps exponentially) with increasing $i$. The weighted average in the big square brackets could then become considerably smaller than the above general bound representing the maximum of the numbers $[n/(n-1)]^{(k-i-1)/2}$, $i = 0, 1, \ldots, k-1$.

# Chapter 3

# Ellipsoid algorithms for computing the intersection of a centrally symmetric body with a line in relative scale

## 3.1 Introduction

The primary objects of this chapter are nonzero vectors $d, a_1, \ldots, a_m \in \mathbf{E}^*$ and the centrally symmetric convex set

$$\mathcal{Q} := \operatorname{conv}\{\pm a_i \ : \ i = 1, 2, \ldots, m\}. \tag{3.1}$$

As before, $\mathbf{E}$ is a finite dimensional real vector space and $\mathbf{E}^*$ is its dual. Our main goal is to find the intersection point of $\mathcal{Q}$ and the line passing through $d$ and the origin.

While this problem can be treated with the methods of the previous chapter, we propose a novel approach by constructing a sequence of ellipsoids inscribed in $\mathcal{Q}$, greedily "converging" towards the intersection points. We develop three algorithms. Our first method is not practical but it serves as the motivational starting point for the development of more efficient approaches. The more practical variants can be viewed as nontrivial modifications of Khachiyan's ellipsoidal rounding algorithm (Algorithm 8 from Chapter 2) to our problem. While the generic structure of an iteration is identical to that of Khachiyan, we employ a different strategy for choosing the update vector and work with a different line search objective function. One aspect of our contribution is therefore showing that modifications of this

type can produce meaningful sequences of ellipsoids. Our algorithms can also be interpreted as performing Frank-Wolfe steps for a specific convex function [8] on the unit simplex in $\mathbf{R}^m$.

We consider several other closely related problems and show that our methods simultaneously approximately solve all of them — within relative error $\delta$ — in $O(1/\delta)$ iterations of a first-order type. One of these is the problem of minimizing the maximum of absolute values of the linear functionals $\langle a_i, x\rangle$ over the hyperplane defined by $\langle d, x\rangle = 1$. This is an unconstrained piecewise-linear convex problem. Another is the problem of finding the smallest $\ell_1$ norm solution of a full-rank underdetermined linear system. Finally, we consider maximization of the linear functional $\langle d, \cdot\rangle$ over a centrally symmetric polytope, the *polar* of $\mathcal{Q}$:

$$\mathcal{Q}^\circ = \{x \in \mathbf{E} \ : \ |\langle a_i, x\rangle| \leq 1, \ i = 1, 2, \ldots, m\}. \tag{3.2}$$

This chapter is organized as follows. In Section 3.2 we formulate the various interrelated problems, explore the relationships among them and establish convexity and smoothness of the objective function of the main problem. We finish the section by proving that a single optimality (approximate optimality) condition implies optimality (approximate optimality) in all these problems. The discussion of the algorithms and their analysis is contained in Section 3.3. Finally, in Section 3.5 we describe applications of our methods to truss topology design and optimal design of statistical experiments.

## 3.2 Problem formulations

### 3.2.1 Supports, gauges and polarity

In this part we review some basic convex analysis concepts and establish several simple results which will become useful in later subsections.

**Supports.** The *support function* of a nonempty set $\mathcal{X} \subset \mathbf{E}$ ($\mathcal{G} \subset \mathbf{E}^*$) is the function $\xi_{\mathcal{X}} \colon \mathbf{E}^* \to \bar{\mathcal{R}}$ ($\xi_{\mathcal{G}} \colon \mathbf{E} \to \bar{\mathcal{R}}$) defined by

$$\xi_{\mathcal{X}}(g) := \sup\{\langle g, x\rangle \ : \ x \in \mathcal{X}\}$$

$$(\xi_{\mathcal{G}}(x) := \sup\{\langle g, x\rangle \ : \ g \in \mathcal{G}\}).$$

For example, $\xi_{\mathcal{Q}}(x) = \max\{|\langle a_i, x\rangle| \ : \ i = 1, 2, \ldots, m\}$.

**Polars.** The *polar* of a convex set $\mathcal{X} \subset \mathbf{E}$ ($\mathcal{G} \subset \mathbf{E}^*$) is the set $\mathcal{X}^\circ \subset \mathbf{E}^*$ ($\mathcal{G}^\circ \subset \mathbf{E}$) defined by

$$\mathcal{X}^\circ := \{g \in \mathbf{E}^* \ : \ \langle g, x\rangle \leq 1 \ \text{ for all } \ x \in \mathcal{X}\}$$

$$(\mathcal{G}^\circ := \{x \in \mathbf{E} \ : \ \langle g, x\rangle \leq 1 \ \text{ for all } \ g \in \mathcal{G}\}).$$

For example, see (3.2).

**Gauges.** A *gauge* is a nonnegative positively homogeneous convex function with values in $\bar{\mathcal{R}} := \mathbf{R} \cup \{+\infty\}$, vanishing at the origin. Norms are real-valued positive definite (vanishing only at the origin) symmetric gauges. Seminorms, as opposed to norms, are allowed to vanish at nonzero points. Notice that gauges need not be symmetric, are allowed to vanish at nonzero points and can take on the value $+\infty$. If $\gamma \colon \mathbf{E} \to \bar{\mathcal{R}}$ is a gauge, it is easy to see that

$$\gamma(x) = \gamma_{\mathcal{X}}(x) := \inf\{\tau \geq 0 \ : \ x \in \tau\mathcal{X}\}, \tag{3.3}$$

where

$$\mathcal{X} := \{x \ : \ \gamma(x) \leq 1\}.$$

Note that $0 \in \mathcal{X}$ and that $\mathcal{X}$ is necessarily convex as a sublevel set of a convex function. If $\gamma$ is a *closed* function (i.e. if its epigraph, which is a convex cone in $\mathbf{E} \times \mathbf{R}_+$, is a closed set), as will be the case for the gauges appearing in this text, then the set $\mathcal{X}$ defined above is the *unique* closed convex set containing the origin for which $\gamma(x) = \gamma_{\mathcal{X}}(x)$. This relation is best understood intuitively as follows. If one thinks of $\gamma$ as being a norm, then $\mathcal{X}$ corresponds to the unit ball ($\mathcal{X}$ may not be closed or bounded) and the above description of $\gamma$ says that the norm of $x$ is equal to the smallest nonnegative number $\tau$ by which one has to scale the unit ball in order to contain $x$.

**Two important gauges.** One of the important gauges encountered in this chapter is a seminorm on $\mathbf{E}$ defined by a positive semidefinite self-adjoint linear operator $U : \mathbf{E} \to \mathbf{E}^*$:

$$\|x\|_U := \langle Ux, x\rangle^{1/2}. \tag{3.4}$$

It can easily be verified that

$$\|x\|_U = 0 \quad \Leftrightarrow \quad x \in \operatorname{null}(U). \tag{3.5}$$

Another crucial gauge is a norm defined on $\operatorname{range}(U) \subset \mathbf{E}^*$ and extended to a gauge on $\mathbf{E}^*$ by allowing it to take the value $+\infty$ on the remainder of the space:

$$\|g\|_U^* := \begin{cases} \langle g, x\rangle^{1/2} & \text{if } g \in \operatorname{range}(U) \text{ with } Ux = g, \\ +\infty & \text{otherwise.} \end{cases} \tag{3.6}$$

Notice that $\langle g, x'\rangle = \langle g, x''\rangle$ whenever $Ux' = g$ and $Ux'' = g$ because $U$ is self-adjoint and hence $\langle g, x'\rangle = \langle Ux'', x'\rangle = \langle Ux', x''\rangle = \langle g, x''\rangle$, all of which are non-

negative since $U \succeq 0$ and, for example, $\langle g, x' \rangle = \langle Ux', x' \rangle \geq 0$. Hence (3.6) gives a valid definition.

In view of the representation (3.3), let us establish special notation for the sublevel sets of $\|\cdot\|_U$ and $\|\cdot\|_U^*$:

$$\mathcal{B}^\circ(U) := \{x \in \mathbf{E} \ : \ \|x\|_U \leq 1\}, \quad \text{and} \tag{3.7}$$

$$\mathcal{B}(U) := \{g \in \mathbf{E}^* \ : \ \|g\|_U^* \leq 1\}, \tag{3.8}$$

so that

$$\|x\|_U = \gamma_{\mathcal{B}^\circ(U)}(x) \quad \text{and} \quad \|g\|_U^* = \gamma_{\mathcal{B}(U)}(g). \tag{3.9}$$

Note that $\mathcal{B}^\circ(U)$ is an *ellipsoidal cylinder* in $\mathbf{E}$ and $\mathcal{B}(U)$ is an *ellipsoid* in range$(U)$.

We shall now show that the gauges defined in (3.4) and (3.6) and their level sets are related via polarity: $\|\cdot\|_U$ is the support function of $\mathcal{B}(U)$, $\|\cdot\|_U^*$ is the support function of $\mathcal{B}^\circ(U)$ and the sets $\mathcal{B}^\circ(U)$ and $\mathcal{B}(U)$ are mutually polar, justifying the notation. We refer to the following fact.

**Fact 3.2.1.** *Closed convex sets $\mathcal{X} \in \mathbf{E}$ and $\mathcal{G} \in \mathbf{E}^*$ containing the origin are mutually polar if and only if $\xi_{\mathcal{G}} = \gamma_{\mathcal{X}}$ and $\xi_{\mathcal{X}} = \gamma_{\mathcal{G}}$:*

*Proof.* Follows from Rockafellar [28], Theorems 14.5 and 15.1. □

**Proposition 3.2.2.** *We have $\xi_{\mathcal{B}(U)}(x) = \|x\|_U$ and $\xi_{\mathcal{B}^\circ(U)}(g) = \|g\|_U^*$ and the sets $\mathcal{B}(U)$ and $\mathcal{B}^\circ(U)$ are mutual polars.*

*Proof.* Once we have shown the first two statements, the assertion that $\mathcal{B}(U)$ and $\mathcal{B}^\circ(U)$ are mutually polar sets follows from (3.9) and Fact 3.2.1. We will give a detailed proof of the identity $\xi_{\mathcal{B}(U)}(x) = \|x\|_U$; the second one can be shown in an

analogous way. First, notice that

$$\begin{aligned}\xi_{\mathcal{B}(U)}(x) &= \sup_g\{\langle g, x\rangle \ : \ \|g\|_U^* \leq 1\} && (P^*)\\ &= \sup_{g,y}\{\langle g, x\rangle \ : \ \langle g, y\rangle^{1/2} \leq 1,\ Uy = g\} \\ &= \sup_y\{\langle Ux, y\rangle \ : \ \langle Uy, y\rangle \leq 1\}. && (P^{**})\end{aligned}$$

We will first argue that $(P^*)$ has a maximizer. For this we just need to note that the objective function is continuous (linear) and that the set $\mathcal{B}(U)$ is compact because it is the unit ball with respect to the norm $\|\cdot\|_U^*$ defined on range$(U)$. If $g$ is the maximizer, then in particular it must be feasible whence $g \in$ range$(U)$. If we let $y$ be any solution of $Uy = g$, then $y$ is a maximizer of $(P^{**})$. Let $y'$ be any such optimal point. Notice that both the objective and the constraint functions of $(P^{**})$ are differentiable. The Mangasarian-Fromovitz constraint qualification[1](MFCQ) for $(P^{**})$ holds at $y'$ if the derivative of the constraint function at $y'$ is nonzero provided that the constraint is active; i.e. MFCQ holds at $y'$ exactly when the following implication holds:

$$\langle Uy', y'\rangle = 1 \quad \Rightarrow \quad 2Uy' \neq 0.$$

This is satisfied trivially, and hence the (necessary) Karush-Kuhn-Tucker[1] (KKT) conditions imply the existence of a nonnegative multiplier $\lambda$ such that

$$Ux = \lambda(2Uy') \qquad \text{and} \qquad \lambda(\langle Uy', y'\rangle - 1) = 0. \tag{3.10}$$

If $\lambda = 0$ then $x \in$ null$(U)$, in which case every feasible $y$ is a maximizer with the maximum equal to 0. The result clearly holds in this case since $\|x\|_U = 0$. If $\lambda > 0$ then the KKT conditions (3.10) imply $y' \in x/(2\lambda) +$ null$(U)$ and $\langle Uy', y'\rangle = 1$.

[1]See, for example, Section 2.3 in [5].

Since range$(U) \perp$ null$(U)$, we obtain $2\lambda = \|x\|_U$. The optimal objective value of $(P^{**})$ therefore is

$$\langle Ux, y' \rangle = \langle Ux, x/\|x\|_U \rangle = \|x\|_U,$$

which finishes the proof. □

Notice that if $x \in$ null$(U)$, then the set of maximizers of $(P^{**})$ is $\mathcal{B}^\circ(U) := \{y \ : \ \langle Uy, y \rangle \leq 1\}$ (all points of this set have equal objective and the necessary KKT conditions say that all optimal points must lie in this set) and hence the set of optimal points of $(P^*)$ is

$$U[\mathcal{B}^\circ(U)] = \{g \ : \ g = Uy, \ \langle Uy, y \rangle \leq 1\} = \{g \ : \ \|g\|_U^* \leq 1\} = \mathcal{B}(U),$$

with the optimum equal to 0. In case $x \notin$ null$(U)$, the set of maximizers of $(P^{**})$ is $\mathcal{Z} := x/\|x\|_U +$ null$(U)$. Hence the set of optimal points of $(P^*)$ is $\{g = Uz \ : \ z \in \mathcal{Z}\} = \{Ux/\|x\|_U\}$ – a singleton. Let us rephrase this observation:

1. If $\|x\|_U = 0$ then $\langle g, x \rangle = \|x\|_U = 0$ for all $g$ with $\|g\|_U^* \leq 1$ (in fact, for all $g \in$ range$(U)$).

2. If $\|x\|_U \neq 0$ then $\langle g, x \rangle \leq \|x\|_U$ for all $g$ with $\|g\|_U^* \leq 1$, with equality exactly when $g = Ux/\|x\|_U$.

A direct consequence of this is a Cauchy-Schwarz type inequality for gauges:

**Corollary 3.2.3** (Cauchy-Schwarz)**.** *For all $x \in \mathbf{E}$ and $g \in$ range$(U)$ we have*

$$\langle g, x \rangle \leq \|g\|_U^* \|x\|_U, \tag{3.11}$$

*with equality exactly in one of the two cases*

1. $\|x\|_U = 0$, *or*

*2.* $\|x\|_U \neq 0$ *and* $g$ *is a nonnegative multiple of* $Ux$*.*

Corollary 3.2.3 can be viewed as a special case (with $\mathcal{G} = \mathcal{B}(U)$) of the following general result:

**Fact 3.2.4** (Cauchy-Schwarz for general gauges)**.** *If* $\mathcal{G} \subset \mathbf{E}^*$ *and* $\mathcal{X} \subset \mathbf{E}$ *are mutually polar sets (both must then be closed, convex and contain the origin), then*

$$\langle g, x\rangle \leq \gamma_{\mathcal{G}}(g)\gamma_{\mathcal{G}^\circ}(x) \quad \textit{for all} \quad g \in \operatorname{dom}\gamma_{\mathcal{G}},\ x \in \operatorname{dom}\gamma_{\mathcal{G}^\circ}.$$

*Proof.* See the definition of a polar gauge and Theorem 15.1 in Rockafellar [28]. □

**Proposition 3.2.5** (Projection)**.** *We have*

$$\min_{\bar{x}}\{\|\bar{x}\|_U \ :\ \langle d, \bar{x}\rangle = 1\} = 0 \quad \Leftrightarrow \quad d \notin \operatorname{range}(U), \tag{3.12}$$

*and the following statements are equivalent:*

*(i)* $x \in \arg\min_{\bar{x}}\{\|\bar{x}\|_U \ :\ \langle d, \bar{x}\rangle = 1\},\ d \in \operatorname{range}(U)$*,*

*(ii)* $Uy = d,\ x = y/(\|d\|_U^*)^2$ *for some* $y$*, and*

*(iii)* $\langle d, x\rangle = \|d\|_U^*\|x\|_U = 1$*.*

*Proof.* Although statement (3.12) can be obtained using a standard separation result, we will use an optimization argument that will also be useful in proving the equivalence of $(i)$, $(ii)$ and $(iii)$. The KKT conditions (necessary and sufficient by convexity of objective and linearity of constraints) for the minimization problem above (with the objective function replaced by $\|\bar{x}\|_U^2$) are

$$2Ux = \lambda d, \qquad \langle d, x\rangle = 1, \qquad \lambda \in \mathbf{R}, \tag{3.13}$$

and we immediately get $\|x\|_U^2 = \langle Ux, x\rangle = \frac{\lambda}{2}\langle d, x\rangle = \frac{\lambda}{2}$, and in particular, $\lambda \geq 0$. If the optimal objective value $\|x\|_U$ is nonzero, then $\lambda > 0$ and hence $d \in \operatorname{range}(U)$

by (3.13). Conversely, if $d \in \text{range}(U)$ and $\|x\|_U = 0$, or, equivalently $x \in \text{null}(U)$ by (3.5), then $\langle d, x\rangle = 0$, which is a contradiction. This establishes (3.12).

If we assume $(i)$, then by (3.13) we must have $\lambda > 0$ since otherwise $\|x\|_U = 0$, which by (3.12) implies $d \notin \text{range}(U)$. We claim that $y := \frac{2}{\lambda}x$ satisfies $(ii)$. Indeed, $Uy = d$ follows from (3.13) and we also get

$$y = \tfrac{2}{\lambda}x = \tfrac{2}{\lambda}\langle d, x\rangle x = \langle d, y\rangle x = (\|d\|_U^*)^2 x.$$

If $x$ and $y$ are as in $(ii)$ then $\langle d, x\rangle = \langle d, y\rangle/(\|d\|_U^*)^2 = 1$ and

$$\|d\|_U^*\|x\|_U = \|d\|_U^* \frac{\|y\|_U}{(\|d\|_U^*)^2} = \frac{\|y\|_U}{\|d\|_U^*} = \frac{\langle Uy, y\rangle^{1/2}}{\langle d, y\rangle^{1/2}} = 1,$$

establishing $(iii)$. For $(iii) \Rightarrow (i)$ notice that for any $\bar{x}$ satisfying $\langle d, \bar{x}\rangle = 1$, the Cauchy-Schwarz inequality (3.11) gives $\|x\|_U = \|x\|_U\langle d, \bar{x}\rangle \leq \|x\|_U\|d\|_U^*\|\bar{x}\|_U = \|\bar{x}\|_U$. Also, $d \in \text{range}(U)$ since otherwise $\|d\|_U^* = +\infty$, which contradicts $(iii)$.

□

### 3.2.2 The first five problems

For $x \in \mathbf{E}$ let

$$\varphi(x) := \xi_{\mathcal{Q}}(x) = \max_i |\langle a_i, x\rangle| \tag{3.14}$$

and consider the problem

$$\boxed{\varphi^* := \min_x \{\varphi(x) \ : \ \langle d, x\rangle = 1\}.} \tag{$P1$}$$

The objective function is a nonnegative sublinear (convex and positively homogeneous) function with subdifferential at the origin equal to $Q$. Note that we always have $0 \in \mathcal{Q}$. We will however further assume that

$$0 \in \text{int } \mathcal{Q}. \tag{3.15}$$

This implies that $\varphi$ vanishes only at the origin, which is then the *unique* global minimizer of $\varphi$, whence $\varphi^* > 0$. Assumption (3.15) is equivalent to

$$\text{range}(A) = \mathbf{E}^*, \tag{3.16}$$

where $A = [a_1, \ldots, a_m]\colon \mathbf{R}^m \to \mathbf{E}^*$ is the linear operator mapping the $i$-th unit vector of $\mathbf{R}^m$ to $a_i$. By $A^*$ we denote the adjoint of $A$. This is the operator $A^*\colon \mathbf{E} \to (\mathbf{R}^m)^*$ defined by $\langle Av, x\rangle = \langle A^*x, v\rangle$ for all $x \in \mathbf{E},\ v \in \mathbf{R}^m$, so that $A^*x = [\langle a_1, x\rangle, \ldots, \langle a_m, x\rangle]^T$ and hence

$$\varphi(x) = \|A^*x\|_\infty.$$

The (Lagrangian) dual of problem $(P1)$ can be shown to be equivalent to

$$\boxed{\varphi^* = \max_\tau\{\tau \ :\ \tau d \in \mathcal{Q}\}} \tag{$D1$}$$

and hence

$$\varphi^* d \in \text{bdry } \mathcal{Q}. \tag{3.17}$$

As an exercise, let us check weak duality. Assume we have $x$ with $\langle d, x\rangle = 1$ and $\tau$ with $\tau d \in \mathcal{Q}$. Then $\tau d$ is a weighted average of points from $\{\pm a_i,\ i = 1, 2, \ldots, m\}$ and hence $\tau = \langle \tau d, x\rangle$ is equal to a weighted average of inner products from $\{\langle \pm a_i, x\rangle,\ i = 1, 2, \ldots, m\}$. Therefore, the maximum inner product is at least $\tau$.

Formulation $(D1)$ has an evident geometric meaning: find the intersection of $\mathcal{Q}$ with the half-line $\{\tau d \ :\ \tau \geq 0\}$ by exploring the portion of the line belonging to $\mathcal{Q}$. Because $\tau d$ is always required to lie in $\mathcal{Q}$, this is an *internal* description of the problem. The same underlying geometry can be expressed by considering the portion of the line lying outside $\mathcal{Q}$, thus arriving at the following *external* description:

$$\boxed{\varphi^* = \min_\tau\{\tau > 0 \ :\ \tau d \notin \text{int } \mathcal{Q}\}.} \tag{$D'1$}$$

We mention *both* $(D1)$ and $(D'1)$ because the algorithms we design in later sections give a lower *and* an upper bound on $\varphi^*$, thus producing feasible solutions (with certain prescribed relative accuracy) to *both* problems. In fact, *all* the problems we consider in this chapter have optimal value either $\varphi^*$ or $1/\varphi^*$ and hence all can be viewed as specific formulations of the same underlying (one-dimensional) geometric problem.

Problem $(P1)$ can be reformulated as follows:

$$\begin{aligned}
\varphi^* &:= \min_x\{\max_i |\langle a_i, x\rangle| \text{ s.t. } \langle d, x\rangle = 1\} \\
&= \min_{x,\tau}\{\tau \ : \ \max_i |\langle a_i, x\rangle| \le \tau,\ \langle d, x\rangle = 1\} \\
&= \min_{x,\tau}\{\tau \ : \ x \in \tau\mathcal{Q}^\circ,\ \langle d, x\rangle = 1,\ \tau \ge 0\} \\
&= \min_{z,\tau}\{\tau \ : \ z \in \mathcal{Q}^\circ,\ \langle d, z\rangle = 1/\tau,\ \tau \ge 0\} \\
&= \left[\max_{z,\tau}\{1/\tau \ : \ z \in \mathcal{Q}^\circ,\ \langle d, z\rangle = 1/\tau,\ \tau \ge 0\}\right]^{-1} \\
&= \left[\max_z\{\langle d, z\rangle \ : \ z \in \mathcal{Q}^\circ\}\right]^{-1},
\end{aligned}$$

and therefore

$$\boxed{\tfrac{1}{\varphi^*} = \max_z\{\langle d, z\rangle \ : \ z \in \mathcal{Q}^\circ\} = \xi_{\mathcal{Q}^\circ}(d).} \qquad (P2)$$

If $x$ is feasible for $(P1)$ then $z := x/\varphi(x)$ is feasible for $(P2)$ as $\max_i |\langle a_i, z\rangle| = \max_i |\langle a_i, x\rangle|/\varphi(x) = 1$. On the other hand, if $z$ is feasible for $(P2)$ then $x := z/\langle d, z\rangle$ is feasible for $(P1)$ because $\langle d, x\rangle = \langle d, z\rangle/\langle d, z\rangle = 1$. A slightly more careful look at the above chain of equalities reveals the following:

**Proposition 3.2.6.** *Point $x = z/\langle d, z\rangle$ is a minimizer of $(P1)$ with optimal value $\varphi^*$ if and only if $z = x/\varphi(x)$ is a maximizer of $(P2)$ with optimal value $1/\varphi^*$.*

Consider now the dual of $(P2)$. It can be written as

$$\boxed{\tfrac{1}{\varphi^*} = \min_v\{\|v\|_1 \ : \ Av = d,\ v \in \mathbf{R}^m\}.} \qquad (D2)$$

This is the problem of finding the smallest $\ell_1$ norm solution of the underdetermined full rank (see assumption (3.16)) linear system $Av = d$. Let us again check weak duality. For any $z \in \mathcal{Q}^\circ$ and $v$ with $Av = d$, one has

$$\begin{aligned}
\|v\|_1 - \langle d, z\rangle &= \|v\|_1 - \langle Av, z\rangle \\
&= \|v\|_1 - \langle v, A^* z\rangle \\
&= \|v\|_1 - \sum_{i=1}^{m} v_i \langle a_i, z\rangle \\
&= \sum_{i=1}^{m} (|v_i| - v_i \langle a_i, z\rangle) \\
&\geq \sum_{i=1}^{m} (|v_i| - |v_i||\langle a_i, z\rangle|) \\
&\geq 0.
\end{aligned}$$

We arrive at this straightforward observation:

**Proposition 3.2.7.** *Point $z$ feasible for (P2) ($v$ feasible for (D2)) is optimal if and only if there is $v$ feasible for (D2) ($z$ feasible for (P2)) such that the following complementary slackness conditions hold:*

$$|v_i| = v_i \langle a_i, z\rangle, \qquad i = 1, 2, \ldots, m. \tag{3.18}$$

Using the complementary slackness condition (3.18) between problem $(P2)$ and its dual $(D2)$ together with the relationship between problems $(P1)$ and $(P2)$ given by Proposition 3.2.6 and the discussion preceding it, we have arrived at the following complementary slackness condition between problems $(P1)$ and $(D2)$:

$$|v_i|\varphi(x) = v_i \langle a_i, x\rangle, \quad i = 1, 2, \ldots, m. \tag{3.19}$$

Note that (3.19) is equivalent to

$$v_i \neq 0 \quad \Rightarrow \quad \varphi(x) = |\langle a_i, x\rangle|, \quad \text{and} \quad \operatorname{sign}(\langle a_i, x\rangle) = \operatorname{sign}(v_i). \tag{3.20}$$

We have thus shown the following.

**Proposition 3.2.8.** *Point $x$ feasible for (P1) ($v$ feasible for (D2)) is optimal if and only if there is $v$ feasible for (D2) ($x$ feasible for (P1)) such that the following complementary slackness conditions hold for $i = 1, 2 \ldots, m$:*

$$v_i > 0 \quad \Rightarrow \quad \varphi(x) = \langle a_i, x \rangle, \quad \textit{and}$$

$$v_i < 0 \quad \Rightarrow \quad \varphi(x) = \langle -a_i, x \rangle.$$

Alternatively, the statement above is equivalent to saying that there is a subdifferential of the objective function $\varphi$ at $x$ such that its negative lies in the normal cone to the constraint set at $x$.

### 3.2.3 Convex combinations of rank-one operators

The operator $U$ of interest in the remainder of this chapter is one arising as a weighted average of rank-one operators coming from the points defining $\mathcal{Q}$:

$$U(w) := \sum_{i=1}^{m} w_i a_i a_i^*, \quad w \in \Delta_m. \tag{3.21}$$

For notational convenience we let $\mathcal{B}(w) := \mathcal{B}(U(w))$ and $\mathcal{B}^\circ(w) := \mathcal{B}^\circ(U(w))$. The following simple fact about the dependence of the range of the operator $U(w)$ on the weights defining it will be needed at several occasions in the text.

**Proposition 3.2.9.** $\operatorname{range} U(w) = \operatorname{span}\{a_i \ : \ w_i \neq 0\}$.

*Proof.* Let $U = U(w)$ and $\tilde{A}$ be the matrix obtained from $A = [a_1, \ldots, a_m]$ by excluding all columns with zero weights. Let $\tilde{w}$ be defined in an analogous fashion. Note that for any $x$, $Ux$ is a linear combination of columns of $\tilde{A}$ and thus $\operatorname{range}(U) \subset \operatorname{range}(\tilde{A})$. However,

$$\operatorname{rank}(U) = \operatorname{rank}(\tilde{A} \operatorname{diag}(\tilde{w}) \tilde{A}^*) = \operatorname{rank}(\tilde{A}\tilde{A}^*) = \operatorname{rank}(\tilde{A})$$

and hence range$(U)$ = range$(\tilde{A})$. □

By Proposition 3.2.9, $U(w)$ is invertible (and hence $\mathcal{B}(w)$ is a full-dimensional ellipsoid) if the vectors $a_i$ with nonzero weights span $\mathbf{E}^*$. Since range$(A) = \mathbf{E}^*$ by (3.16), this happens, in particular, when all weights are positive.

The starting point of our discussion is the simple observation that $\mathcal{B}(w) \subset \mathcal{Q}$ for all $w \in \Delta_m$ and hence by taking polars, $\mathcal{Q}^\circ \subset \mathcal{B}^\circ(w)$.

**Proposition 3.2.10.** *The following holds:*

*(i) For all $x \in \mathbf{E}$ and $w \in \Delta_m$,*

$$\|x\|_{U(w)} \leq \varphi(x)$$

*with equality if and only if the following condition holds:*

$$w_i > 0 \quad \Rightarrow \quad \varphi(x) = |\langle a_i, x\rangle|, \qquad i = 1, 2, \ldots, m.$$

*(ii)* $\varphi(x) = \max_w\{\|x\|_{U(w)} \ : \ w \in \Delta_m\}$.

*(iii)* $\mathcal{B}(w) \subset \mathcal{Q}$ *and* $\mathcal{Q}^\circ \subset \mathcal{B}^\circ(w)$ *for all* $w \in \Delta_m$.

*Proof.* Part $(i)$ follows from

$$\xi_{\mathcal{B}(w)}(x) = \|x\|_{U(w)} = [\sum_i w_i \langle a_i, x\rangle^2]^{1/2} \leq \max_i |\langle a_i, x\rangle| = \xi_{\mathcal{Q}}(x) \equiv \varphi(x).$$

Condition for equality and parts $(ii)$ and $(iii)$ follow easily. □

**Example 3.2.11** (Polarity)**.** In Figure 3.1, $\mathcal{Q}$ is the $\ell_\infty$ unit ball and $\mathcal{Q}^\circ$ is the $\ell_1$ unit ball. Since $\mathcal{B}(w) \subset \mathcal{Q}$, the polar sets will satisfy the reverse inclusion: $\mathcal{Q}^\circ \subset \mathcal{B}^\circ(w)$. The numbers represent the *relative* sizes of the respective line segments, one unit corresponding to $\sqrt{2}/4$.

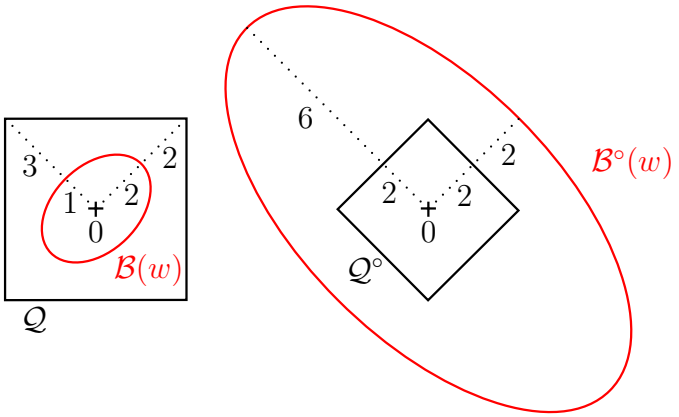



Figure 3.1: Polarity (Example 3.2.11).

### 3.2.4 The main problem

Observe that

$$
\begin{aligned}
\varphi^* &= \min_{x:\langle d,x\rangle=1} \varphi(x) \\
&= \min_{x:\langle d,x\rangle=1} \max_{w\in\Delta_m} \|x\|_{U(w)} \quad \text{(part } (ii) \text{ of Proposition 3.2.10)} \\
&= \max_{w\in\Delta_m} \min_{x:\langle d,x\rangle=1} \|x\|_{U(w)} \\
&= \max_{\substack{w\in\Delta_m \\ d\in\mathrm{range}(U(w))}} \min_{x:\langle d,x\rangle=1} \|x\|_{U(w)} \qquad \text{by (3.12)} \\
&= \max_{\substack{w\in\Delta_m \\ d\in\mathrm{range}(U(w))}} \|x_w\|_{U(w)} \quad (x_w = \text{ minimizer from Proposition 3.2.5, } U = U(w)) \\
&= \max_{\substack{w\in\Delta_m \\ d\in\mathrm{range}(U(w))}} 1/\|d\|^*_{U(w)} \quad \text{(part } (iii) \text{ of Proposition 3.2.5)} \\
&= \max_{w\in\Delta_m} 1/\|d\|^*_{U(w)} \\
&= \left[\min_{w\in\Delta_m} \|d\|^*_{U(w)}\right]^{-1}.
\end{aligned}
$$

The interchange of minimum and maximum in the derivation above can be justified using Hartung's [12] generalization of Sion's [31] minimax theorem. If we write $\psi(w) := \|d\|^*_{U(w)}$, this observation shows that $\psi^* = 1/\varphi^*$, where

$$\boxed{\psi^* := \min_w \{\|d\|^*_{U(w)} \ : \ w \in \Delta_m\}.} \qquad (P3)$$

There is another way of seeing that $\psi^* = 1/\varphi^*$ and that the minimum is attained. The proof will give us an important insight into the relationship between the feasible solutions of problems $(P3)$ and $(D2)$, revealing an algorithmic idea for solving both problems. We will need two intermediate results.

**Lemma 3.2.12.** *Assume $U(w)y = d$ for some $w \in \Delta_m$ and $y \in \mathbf{E}$. If we set $v_i = w_i\langle a_i, y\rangle$ for $i = 1, \ldots, m$, then $Av = d$ and $\|v\|_1 \leq \|d\|^*_{U(w)}$.*

*Proof.* By the inequality between the weighted arithmetic and quadratic means we get

$$\|v\|_1 = \sum_i w_i|\langle a_i, y\rangle| \leq [\sum_i w_i\langle a_i, y\rangle^2]^{1/2} = \langle U(w)y, y\rangle^{1/2} = \langle d, y\rangle^{1/2} = \|d\|^*_{U(w)}.$$

□

This result says: if $w$ is feasible for $(P3)$ with finite objective value, then the objective value of $(D2)$ for some $v$ is no bigger than that of $(P3)$ for $w$.

**Lemma 3.2.13.** *For $0 \neq v \in \mathbf{R}^m$ let $c := Av$ and $w := |v|/\|v\|_1$. Then*

$$\|c\|^*_{U(w)} \leq \|v\|_1.$$

*Proof.* We can wlog assume that $\|v\|_1 = 1$ since both sides of the inequality are positively homogeneous in $v$. Let $U = U(|v|)$. By Proposition 3.2.9, there is $y \in \mathbf{E}$

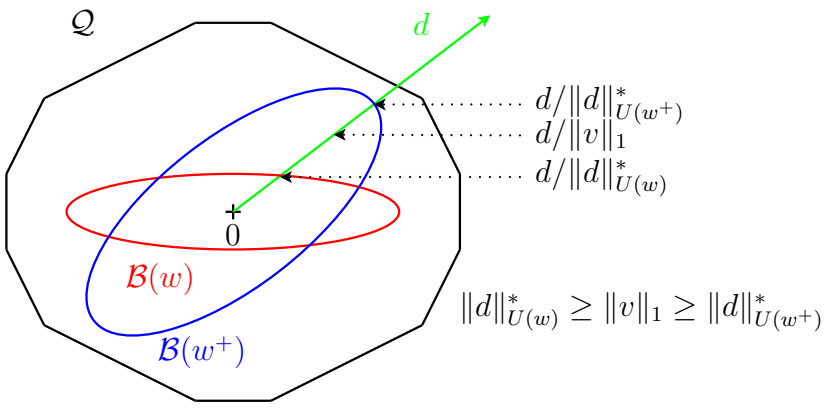



Figure 3.2: Geometry of Lemma 3.2.12 and Lemma 3.2.13.

for which $Uy = c$ and

$$\langle c, y\rangle = \langle \sum_i v_i a_i, y\rangle \leq \sum_i |v_i||\langle a_i, y\rangle|$$
$$\leq [\sum_i |v_i|\langle a_i, y\rangle^2]^{1/2} = \langle Uy, y\rangle^{1/2} = \langle c, y\rangle^{1/2}.$$

We have again used the inequality between the weighted arithmetic and quadratic means. It follows that $\|c\|^*_U = \langle c, y\rangle^{1/2} \leq 1$. □

The interpretation of the above result that we will use is the following: if $v$ is feasible for $(D2)$ then the objective value of $(P3)$ for some $w$ is no bigger than that of $(D2)$ for $v$. Lemma 3.2.12 and Lemma 3.2.13 have a nice geometric interpretation.

First notice that $c' = Av/\|v\|_1$ lies in $\mathcal{Q}$ and that any point of $\mathcal{Q}\backslash\{0\}$ can be we written in this form for some $0 \neq v \in \mathbf{R}^m$, i.e. $\mathcal{Q} = \{0\} \cup \{Av/\|v\|_1 \ : \ v \in \mathbf{R}^m\backslash\{0\}\}$. Also notice that $0 \in \mathcal{B}(w)$ for any $w \in \Delta_m$. Lemma 3.2.13 therefore says that any point $c'$ of $\mathcal{Q}$ can be enclosed into $\mathcal{B}(w)$ (i.e. $\|c'\|^*_{U(w)} \leq 1$) for properly chosen weights $w$. By part $(ii)$ of Proposition 3.2.10, $c' \in \mathcal{B}(w) \subset \mathcal{Q}$.

If the set $\{a_i \ : \ w_i \neq 0\}$ spans $\mathbf{E}^*$, then $\mathcal{B}(w)$ is a full dimensional ellipsoid

contained in $\mathcal{Q}$ and containing the point $c' = c/\|v\|_1$. In particular, any point $c'$ in the interior of $\mathcal{Q}$ or in the relative interior of a full dimensional face of $\mathcal{Q}$ can be enclosed into a full-dimensional ellipsoid which is, in turn, contained in $\mathcal{Q}$. We have obtained the following:

**Theorem 3.2.14.** *The optimal value of (P3) is $1/\varphi^*$ and it is attained.*

*Proof.* The first part is a direct consequence of Lemma 3.2.12, Lemma 3.2.13, and the fact that the optimal value of $(D2)$ is $1/\varphi^*$. Attainment follows from the fact that the minimum is attained in $(D2)$ by some $v^*$, which can be used via Lemma 3.2.13 and the first part of this theorem to establish the existence of a minimizer of $(P3)$. □

### 3.2.5 Common origin of the many optimization problems

The purpose of this section is to expose a unified geometric point of view explaining the origin of the numerous problems encountered in previous subsections.

If we wish to generate a number of optimization problems from Figure 3.1, we can consider the support functions of the sets $\mathcal{Q}$, $\mathcal{B}(w)$, $\mathcal{Q}^\circ$ and $\mathcal{B}^\circ(w)$ — see Figure 3.3. The value of $\xi_{\mathcal{Q}}$ at a particular point $x \in \mathbf{E}$ can be computed in $O(mn)$ arithmetic operations. It is easy to see that the support functions of the ellipsoids $\mathcal{B}(w)$ and $\mathcal{B}^\circ(w)$ are the gauges $\|\cdot\|_{U(w)}$ and $\|\cdot\|^*_{U(w)}$ (Proposition 3.2.2) and can be both computed in $O(mn^2)$ arithmetic operations. Indeed, the formation of $U(w)$ takes $O(mn^2)$ work, the multiplication $U(w)x$ takes $O(n^2)$ operations and the computation of $\langle U(w)x, x\rangle$ takes an additional $O(n)$ operations (let us neglect the square root work). In the evaluation of $\|x\|^*_{U(w)}$ we also need to invert $U(w)$, which takes $O(n^3)$ arithmetic operations. Since $m \geq n$ by our assumption on

full-dimensionality of $\mathcal{Q}$, the dominant work is performed by the formation of the matrix. Finally, evaluation of $\xi_{\mathcal{Q}^\circ}$ at any particular point $g \in \mathbf{E}^*$ amounts to solving a linear program (LP).

| Problem | Arithmetic operations |
|---|---|
| $\xi_{\mathcal{Q}}(x) = \max\{\lvert\langle a_i, x\rangle\rvert \ : \ i = 1, 2, \ldots, m\}$ | $O(mn)$ |
| $\xi_{\mathcal{B}(w)}(x) = \lVert x\rVert_{U(w)}$ | $O(mn^2)$ |
| $\xi_{\mathcal{Q}^\circ}(d) = \max\{\langle d, x\rangle \ : \ x \in Q^\circ\}$ | LP |
| $\xi_{\mathcal{B}^\circ(w)}(d) = \lVert d\rVert^*_{U(w)}$ | $O(mn^2)$ |

Figure 3.3: Support functions of $\mathcal{Q}$ and $\mathcal{B}(w)$ and their polars.

Notice that while Figure 3.1 enjoys considerable symmetry, our focus on a *fixed* $d \in \mathbf{E}^*$ disrupts this symmetry. In particular, $\xi_{\mathcal{Q}}(x)$ depends on $x$, $\xi_{\mathcal{B}(w)}(x)$ on both $x$ and $w$, $\xi_{\mathcal{Q}^\circ}(d)$ is constant and finally $\xi_{\mathcal{B}^\circ(w)}(d)$ depends on $w$. Finally, if we now look at the optimization problems derived from the support functions of Figure 3.3, we recognize some of the problems of this chapter — see Figure 3.4.

| problem | related to |
|---|---|
| $\min_x\{\xi_{\mathcal{Q}}(x) \ : \ \langle d, x\rangle = 1\}$ | see $(P1)$ |
| $\max_w\{\xi_{\mathcal{B}(w)}(x) \ : \ w \in \Delta_m\}$ | Proposition 3.2.10 |
| $\min_x\{\xi_{\mathcal{B}(w)}(x) \ : \ \langle d, x\rangle = 1\}$ | Proposition 3.2.5 |
| evaluate $\xi_{\mathcal{Q}^\circ}(d)$ | see $(P2)$ |
| $\min_w\{\xi_{\mathcal{B}^\circ(w)}(d) \ : \ w \in \Delta_m\}$ | see $(P3)$ |

Figure 3.4: Common origin of the many optimization problems.

### 3.2.6 Convexity and smoothness

In this subsection we establish the convexity of the function

$$\psi^2(w) = \|d\|^*_{U(w)}{}^2,$$

and derive formulae for its first and second derivatives. This is the objective function of our main problem squared. Let us start by showing that the domain of $\psi$ (or equivalently of $\psi^2$), defined the usual way as

$$\operatorname{dom}\psi := \{w \in \Delta_m \ : \ \psi(w) < +\infty\} = \{w \in \Delta_m \ : \ d \in \operatorname{range} U(w)\},$$

is convex. For this we will need the following lemma.

**Lemma 3.2.15.** *For any $w', w'' \in \Delta_m$ and $w = \lambda w' + (1-\lambda)w''$ with $0 < \lambda < 1$,*

$$\operatorname{range} U(w') \cup \operatorname{range} U(w'') \subset \operatorname{range} U(w).$$

*Proof.* Notice that for any $i$, the weight $w_i$ is positive if and only if at least one of the weights $w'_i, w''_i$ is positive and hence

$$\{a_i \ : \ w'_i > 0 \text{ or } w''_i > 0\} = \{a_i \ : \ w_i > 0\}.$$

By Proposition 3.2.9,

$$\begin{aligned}
\operatorname{range} U(w') \cup \operatorname{range} U(w'') &= \operatorname{span}\{a_i \ : \ w'_i > 0\} \cup \operatorname{span}\{a_i \ : \ w''_i > 0\} \\
&\subset \operatorname{span}\{a_i \ : \ w'_i > 0 \text{ or } w''_i > 0\} \\
&= \operatorname{span}\{a_i \ : \ w_i > 0\} \\
&= \operatorname{range} U(w).
\end{aligned}$$

□

**Proposition 3.2.16.** *The domain of $\psi$ is a convex set.*

*Proof.* Assume $\psi(w') < +\infty$ and $\psi(w'') < +\infty$ for some $w', w'' \in \Delta_m$, or equivalently, $d \in \operatorname{range} U(w') \cap \operatorname{range} U(w'')$, and consider $w = \lambda w' + (1-\lambda)w''$ for $0 < \lambda < 1$. By Lemma 3.2.15, $d \in \operatorname{range} U(w)$ and hence $\psi(w) < +\infty$. □

Convexity of $\psi^2$ is related to the following well-known fact about the map $C \mapsto C^{-1}$: for $C_1 \succ 0, C_2 \succ 0$ and $0 < \lambda < 1$,

$$(\lambda C_1 + (1-\lambda)C_2)^{-1} \preceq \lambda C_1^{-1} + (1-\lambda)C_2^{-1}.$$

Indeed, notice that this readily implies

$$\langle d, (\lambda C_1 + (1-\lambda)C_2)^{-1} d\rangle \leq \lambda \langle d, C_1^{-1} d\rangle + (1-\lambda)\langle d, C_2^{-1} d\rangle,$$

which can be written as

$$(\|d\|^*_{\lambda C_1 + (1-\lambda)C_2})^2 \leq \lambda (\|d\|^*_{C_1})^2 + (1-\lambda)(\|d\|^*_{C_2})^2.$$

This argument is sufficient to establish the convexity of $\psi^2$ on the set of weights corresponding to invertible matrices:

$$\{w \in \Delta_m \ : \ U(w) \text{ is invertible}\} = \{w \in \Delta_m \ : \ \operatorname{range} U(w) = \mathbf{E}^*\}.$$

In particular, $\psi^2$ is convex on $\operatorname{rint} \Delta_m \subset \operatorname{dom} \psi$. The general argument follows:

**Proposition 3.2.17** (Convexity)**.** *$\psi^2$ is convex on its domain.*

*Proof.* For the sake of the proof we treat $\mathbf{E}$ and $\mathbf{E}^*$ as $\mathbf{R}^n$. Let $w', w'' \in \operatorname{dom} \psi$ and $y', y'' \in \mathbf{E}$ be such that $U(w')y' = d$ and $U(w'')y'' = d$. Further let $w = \lambda w' + (1-\lambda)w''$ for arbitrary $\lambda \in (0,1)$ and $y$ be such that $U(w)y = d$ (we know that $w \in \operatorname{dom} \psi$). We want to show that

$$(\|d\|^*_{U(w)})^2 \leq \lambda (\|d\|^*_{U(w')})^2 + (1-\lambda)(\|d\|^*_{U(w'')})^2,$$

or equivalently,

$$\langle d, y\rangle \leq \lambda\langle d, y'\rangle + (1-\lambda)\langle d, y''\rangle. \tag{3.22}$$

For this we will use the fact that positive semidefinite matrices can be simultaneously diagonalized by a nonsingular matrix (see, for example, [38], Theorem 6.6). Let $P$ be an invertible matrix and $D'$ and $D''$ diagonal matrices with nonnegative entries such that

$$U(w') = PD'P^*, \quad U(w'') = PD''P^*,$$

and hence

$$U(w) = P(\lambda D' + (1-\lambda)D'')P^*.$$

Then (3.22) can be written as

$$\langle P^{-1}d, P^*y\rangle \leq \lambda\langle P^{-1}d, P^*y'\rangle + (1-\lambda)\langle P^{-1}d, P^*y''\rangle,$$

or

$$(\|P^{-1}d\|^*_{\lambda D'+(1-\lambda)D''})^2 \leq \lambda(\|P^{-1}d\|^*_{D'})^2 + (1-\lambda)(\|P^{-1}d\|^*_{D''})^2.$$

This way we have managed to transform the statement to the case of diagonal matrices. For simpler reference and indexing, let us write $c := P^{-1}d$, $x := P^*y$, $x' := P^*y'$ and $x'' := P^*y''$. In terms of this notation, we have reduced the problem to the following: if $D'_{ii}x'_i = c_i, D''_{ii}x''_i = c_i$ and $(\lambda D'_{ii} + (1-\lambda)D''_{ii})x_i = c_i$ for $i = 1, 2, \ldots, n$, then

$$\sum_i c_i x_i \leq \lambda \sum_i c_i x'_i + (1-\lambda)\sum_i c_i x''_i. \tag{3.23}$$

Notice that if either $D'_{ii} = 0$ or $D''_{ii} = 0$ for some $i$, then $c_i = 0$ and the corresponding terms on both sides of (3.23) vanish. It turns out that we can establish (3.23) term-by-term. It is therefore enough to show that

$$\frac{1}{\lambda D'_{ii} + (1-\lambda)D''_{ii}} \leq \lambda\frac{1}{D'_{ii}} + (1-\lambda)\frac{1}{D''_{ii}}$$

for all $i$ such that both $D'_{ii}$ and $D''_{ii}$ are nonzero. However, this follows directly from the convexity of the univariate function $\tau \mapsto 1/\tau$ on $\mathbf{R}_{++}$. □

Alternatively, convexity of $\psi^2$ can be also viewed as a consequence of convexity of the cone of positive semidefinite matrices. We claim that for $\tau \in \mathbf{R}$ and $w \in \Delta_m$,

$$\tau \geq \psi^2(w) \quad \Leftrightarrow \quad \begin{pmatrix} \tau & d^* \\ d & U(w) \end{pmatrix} \succeq 0, \tag{3.24}$$

with the "matrix" representing a linear map from $\mathbf{R} \times \mathbf{E}$ to $\mathbf{R} \times \mathbf{E}^*$ defined the obvious way.

If $d \notin \operatorname{range} U(w)$, then $\psi^2(w) = +\infty$ and we need to show that the operator is *not* positive definite for any $\tau$. Since $U(w)$ is singular, $\operatorname{null} U(w)$ is nontrivial. Clearly there must be $y \in \operatorname{null} U(w)$ for which $\langle d, y\rangle \neq 0$ since otherwise $d$ would be a member of $(\operatorname{null} U(w))^\perp = \operatorname{range} U(w)$. Choose $y$ with $\langle d, y\rangle < 0$ and consider

$$\begin{aligned} \begin{pmatrix} \delta & y^* \end{pmatrix} \begin{pmatrix} \tau & d^* \\ d & U(w) \end{pmatrix} \begin{pmatrix} \delta \\ y \end{pmatrix} &= \tau\delta^2 + 2\langle d, y\rangle\delta + \langle U(w)y, y\rangle \\ &= \tau\delta^2 + 2\langle d, y\rangle\delta, \end{aligned}$$

which is negative for all sufficiently small positive $\delta$.

Now assume $d \in \operatorname{range} U(w)$ and take $y$ such that $U(w)y = d$. For this part of the argument we treat the spaces $\mathbf{E}$ and $\mathbf{E}^*$ as $\mathbf{R}^n$. We do this because we will use a diagonalization technique. The operator from (3.24) is positive semidefinite if and only if the following $(n+1) \times (n+1)$ matrix is positive semidefinite

$$\begin{pmatrix} 1 & -y^T \\ 0 & I_n \end{pmatrix} \begin{pmatrix} \tau & d^T \\ d & U(w) \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -y & I_n \end{pmatrix} = \begin{pmatrix} \tau - \langle d, y\rangle & 0 \\ 0 & U(w) \end{pmatrix}.$$

This happens precisely when $\tau \geq \langle d, y\rangle = \psi^2(w)$.

Consider now $(w', \tau')$, $(w'', \tau'') \in \operatorname{epi} \psi^2$ and $\lambda \in (0,1)$. Letting $\tau = \lambda\tau' + (1-\lambda)\tau''$ and $w = \lambda w' + (1-\lambda)w''$, notice that

$$\begin{pmatrix} \tau & d^* \\ d & U(w) \end{pmatrix} = \lambda \begin{pmatrix} \tau' & d^* \\ d & U(w') \end{pmatrix} + (1-\lambda) \begin{pmatrix} \tau'' & d^* \\ d & U(w'') \end{pmatrix}.$$

Convexity of the epigraph of $\psi^2$ (and hence of $\psi^2$) now follows from convexity of the cone of positive semidefinite matrices.

Note that if $\tau = 0$, the left-hand side statement (and hence both statements) of the equivalence (3.24) holds if and only if $d = 0$. Since we assume $d$ to be nonzero, we can restrict our attention to positive values of $\tau$ only. In such a case, the block matrix of (3.24) is positive semidefinite if and only if $U(w) - \frac{1}{\tau} dd^*$, the Schur complement of $\tau$, is positive semidefinite (see, for example, Theorem 6.13 in [38]). Let us formulate some of these observations (in a bit more general way):

**Lemma 3.2.18.** *If $U\colon \mathbf{E} \to \mathbf{E}^*$ is positive semidefinite and self-adjoint, $g \in \mathbf{E}^*$ and $\tau$ a positive real parameter, then the following statements are equivalent:*

*(i)* $\tau \geq (\|g\|_U^*)^2$,

*(ii)* $\left(\begin{smallmatrix} \tau & g^* \\ g & U \end{smallmatrix}\right) \succeq 0$,

*(iii)* $U - \frac{1}{\tau} gg^* \succeq 0$.

If $U(w)$ is invertible, then $\psi^2(w) = \langle d, U(w)^{-1} d \rangle$ is differentiable at $w$. For an invertible linear operator $C\colon \mathbf{E} \to \mathbf{E}^*$, let $\theta$ be defined by $\theta(C) = C^{-1}$. The fact that $D\theta(C)H = -C^{-1}HC^{-1}$ together with the chain rule gives the following formulae for the first and second (Frechét) derivatives of $\psi^2$.

**Proposition 3.2.19** (Differentiability)**.** *If $U(w)$ is invertible, then $\psi^2$ is differentiable at $w$ and for $h \in \mathbf{R}^m$ we have the following formulae:*

*(i)* $D\psi^2(w)h = -\langle d, U(w)^{-1}U(h)U(w)^{-1}d\rangle$, *and*

*(ii)* $D^2\psi^2(w)[h,h] = 2\langle d, U(w)^{-1}U(h)U(w)^{-1}U(h)U(w)^{-1}d\rangle$.

It is apparent from the form of the Hessian of $\psi^2$ that it is positive semidefinite. Indeed, for any $h \in \mathbf{R}^m$ let $g = U(h)U(w)^{-1}d$ and note that $D^2\psi^2(w)[h,h] = 2\langle g, U(w)^{-1}g\rangle \geq 0$. This is another way to establish convexity of $\psi^2$ (on a smaller set than $\operatorname{dom}\psi$ though).

The following lemma states that the domain of $\psi$ is open *relative* to $\Delta_m$.

**Lemma 3.2.20** (Topology of the domain)**.** *Every $w \in \operatorname{dom}\psi$ has a neighborhood $\mathcal{N}$ such that $\mathcal{N} \cap \Delta_m \subset \operatorname{dom}\psi$.*

*Proof.* For all sufficiently small $h \in \mathbf{R}^m$ and all $i$ we have $w_i + h_i > 0$ whenever $w_i > 0$. If, in addition, $w + h \in \Delta_m$, then our assumption about $w$ and Proposition 3.2.9 imply $d \in \operatorname{range}U(w) \subset \operatorname{range}U(w+h)$. □

**Example 3.2.21.** Consider an example with $n = m = 2$ as in Figures 3.5 and 3.6. We have $a_1 = (1,-1)$, $a_2 = (1,1)$ and $d = (2,0)$ and hence for $(w_1, w_2) \in \Delta_2$ we get

$$U(w_1,w_2) = w_1 a_1 a_1^T + w_2 a_2 a_2^T = \begin{pmatrix} 1 & w_2 - w_1 \\ w_2 - w_1 & 1 \end{pmatrix}.$$

Assuming $w_1 > 0$ and $w_2 > 0$, the system $U(w_1,w_2)y = d$ has the unique solution

$$y_1 = \frac{1}{2}\left(\frac{1}{w_1} + \frac{1}{w_2}\right), \qquad y_2 = \frac{1}{2}\left(\frac{1}{w_2} - \frac{1}{w_1}\right),$$

and therefore

$$\psi^2(w_1,w_2) = \langle d, y\rangle = \frac{1}{w_1} + \frac{1}{w_2}.$$

Note that $\|d\|^*_{U(0.5,0.5)} = \psi(0.5,0.5) = 2$, which geometrically corresponds to the ball $\mathcal{B}(0.5,0.5)$ cutting vector $d$ in half (Figure 3.5). Also observe that as

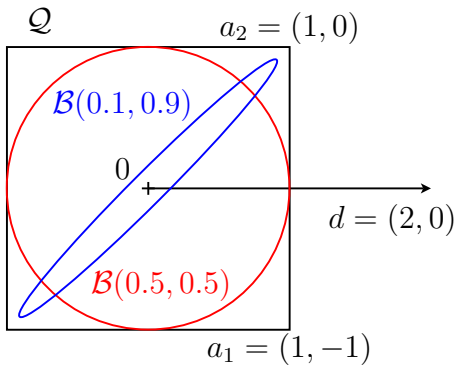



Figure 3.5: Example 3.2.21.

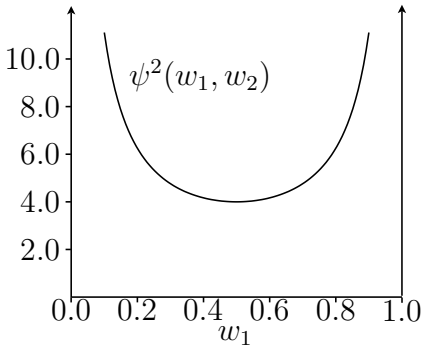



Figure 3.6: The graph of $\psi^2$ (Example 3.2.21).

$w_1 \to 0$, the $\|\cdot\|^*_{U(w)}$-norm of $d$ increases to infinity. This translates to the ellipsoid $\mathcal{B}(w_1, w_2)$ getting thinner, "approaching" the lower dimensional ellipsoid $\mathcal{B}(0, 1)$ — the line segment with endpoints $a_2$ and $-a_2$.

If $w_i = 0$ then $\operatorname{range} U(w_1, w_2) = \operatorname{span}\{a_{3-i}\}$ and we conclude that in either case $d \notin \operatorname{range} U(w_1, w_2)$, implying $\psi^2(w_1, w_2) = +\infty$. Notice that $\psi^2$ is a convex function (as asserted in Proposition 3.2.17; also see Figure 3.6) with convex domain $\{w \in \Delta_m \ : \ w_1 > 0, \ w_2 > 0\}$ (Proposition 3.2.16), which is an open set relative to $\Delta_m$ (Lemma 3.2.20). For any $(w_1, w_2) \in \operatorname{dom} \psi$ and $h$ with $h_1 + h_2 = 0$ we have

$$D\psi^2(w_1, w_2)h = -\left(\frac{h_1}{w_1^2} + \frac{h_2}{w_2^2}\right) = -\langle U(h)y, y\rangle,$$

which agrees with the formula from Proposition 3.2.19.

### 3.2.7 Optimality conditions

In Subsection 3.2.2 we have discussed the basic relationship among the problems $(P1)$, $(D1)$, $(D'1)$, $(P2)$ and $(D2)$. Here we first investigate the necessary and sufficient optimality conditions for problem $(P3)$ and then show that these imply

optimality in $(P1)$, $(P2)$, their duals $(D1)$, $(D2)$, and in $(D'1)$. Finally, we give a single condition implying approximate optimality in all the problems above.

**Lemma 3.2.22.** *Let $U\colon \mathbf{E} \to \mathbf{E}^*$ be self-adjoint and positive semidefinite. Further let $0 \neq c \in \operatorname{range} U$ and assume that $y \in \mathbf{E}$ defines a supporting hyperplane to $\mathcal{B}(U)$ at $c' := c/\|c\|_U^*$ in the following sense:*

$$\langle g, y\rangle < \langle c', y\rangle \qquad \forall\, c' \neq g \in \mathcal{B}(U). \tag{3.25}$$

*Then $c = \lambda U y$ for some $\lambda > 0$.*

*Proof.* First notice that because $0 \neq c \in \operatorname{range} U$, we have $0 < \|c\|_U^* < \infty$. Now observe that $c'$ lies in the relative boundary of $\mathcal{B}(U)$, which implies that a vector $y$ as above exists. By Proposition 3.2.2 and (3.25) we have

$$\|y\|_U = \xi_{\mathcal{B}(U)}(y) = \max\{\langle g, y\rangle \ :\ g \in \mathcal{B}(U)\} = \langle c', y\rangle,$$

and hence

$$\|y\|_U \|c\|_U^* = \langle c, y\rangle.$$

The condition for equality in the Cauchy-Schwarz inequality (Proposition 3.2.3) now implies that either $\|y\|_U = 0$, or otherwise $\|y\|_U \neq 0$ and $c$ is a nonnegative multiple of $Uy$. We claim that the first case can be excluded. Indeed, if $\|y\|_U = 0$ then by (3.5) we get $Uy = 0$, which would in turn imply that $\langle g, y\rangle = 0$ for all $g \in \operatorname{range} U \supset \mathcal{B}(U)$, violating (3.25). The statement of the lemma then follows from the second case discussed above by noting that the assumption $c \neq 0$ implies that the nonnegative multiplier must be in fact positive. □

The above lemma will be used to prove the necessity part of the following optimality condition.

**Proposition 3.2.23.** *Point $w \in \Delta_m$ is optimal for $(P3)$ if and only if there exists $y \in \mathbf{E}$ such that*

*(i)* $U(w)y = d$, *and*

*(ii)* $\varphi(y) = \psi(w)$.

*Condition (ii) can be replaced by*

*(ii')* $w_i > 0 \quad \Rightarrow \quad \varphi(y) = |\langle a_i, y\rangle|, \qquad i = 1, 2, \ldots, m.$

*Proof.* If $(i)$ holds then $x = y/\psi^2(w)$ is feasible for $(P1)$ and hence $\varphi(y/\psi^2(w)) = \varphi(x) \geq \varphi^* = 1/\psi^*$. By homogeneity of $\varphi$ we obtain

$$\varphi(y) \geq \frac{\psi^2(w)}{\psi^*} \geq \psi(w). \tag{3.26}$$

If we additionally assume $(ii)$ then (3.26) must hold will equality and thus $\psi(w) = \psi^*$. As a side product, this also shows that $x$ is optimal for $(P1)$. Conversely, assume $w \in \Delta_m$ is a minimizer of $(P3)$. Then $\|d\|^*_{U(w)} = \psi^* = 1/\varphi^*$ and $\varphi^* d \in \operatorname{bdry} \mathcal{Q}$ by (3.17). Let $y \in \mathbf{E}$ define a supporting hyperplane to $\mathcal{Q}$ at $c' := \varphi^* d$ (and hence to $\mathcal{B}(U(w))$ at the same point) so that $\langle \cdot, y\rangle$ is maximized over $\mathcal{Q}$ at $c'$ (and hence uniquely over $\mathcal{B}(U(w))$ at the same point). Applying Lemma 3.2.22 with $U := U(w)$ and $c := d$ we conclude that $d = \lambda U(w)y$ for some positive $\lambda$. Let us scale $y$ so that $d = U(w)y$, establishing $(i)$. Part $(ii)$ follows from

$$\varphi(y) = \xi_{\mathcal{Q}}(y) = \max\{\langle g, y\rangle \ : \ g \in \mathcal{Q}\} = \langle c', y\rangle = \langle \varphi^* d, y\rangle = \varphi^* \psi^2(w) = \psi(w).$$

The first equality is the definition of $\varphi$, the third is a consequence of the choice of $y$ and the last one follows from $\psi(w) = \psi^* = 1/\varphi^*$. Finally, the equivalence of $(ii)$ and $(ii')$, assuming $(i)$, is apparent if we note that $\varphi(y) = \max_i\{|\langle a_i, y\rangle| \ : \ i = 1, 2, \ldots, m\}$ and $\psi(w) = \|d\|^*_{U(w)} = \|y\|_{U(w)} = \left(\sum_i w_i \langle a_i, y\rangle^2\right)^{1/2}$ (see also part $(i)$ of Proposition 3.2.10). ∎

The optimality conditions of the previous result have a clear geometric meaning (see Figure 3.7). A point $w \in \Delta_m$ is optimal for $(P3)$ precisely when there exists a hyperplane $\mathcal{H}_y$ passing through $d/\|d\|^*_{U(w)}$ which also happens to be a supporting hyperplane of $\mathcal{Q}$. The set $\mathcal{F}_y := \mathcal{H}_y \cap \mathcal{Q}$ is therefore a face of $\mathcal{Q}$ exposed by the direction $y$. Optimality condition $(ii')$ then requires one of the points $a_i$ or $-a_i$ to lie in $\mathcal{F}_y$ if the corresponding weight $w_i$ is positive. In other words, if both $a_i$ and $-a_i$ lie outside this face, then they must have zero weights, at optimality.

Note also that for optimal $w$, the point $v \in \mathbf{R}^m$ defined by $v_i = w_i\langle a_i, y\rangle$ is optimal for $(D2)$, which is a consequence of Lemma 3.2.12 and the fact that the optimal values of problems $(P3)$ and $(D2)$ are equal. The intersection point $\varphi^* d$ of $\{\tau d \ : \ \tau \geq 0\}$ and $\mathcal{Q}$ can be written as

$$\varphi^* d = \varphi^* U(w) y = \varphi^* \sum_{i=1}^m w_i \langle a_i, y\rangle a_i = \sum_{i=1}^m \varphi^* v_i a_i$$

with $\|\varphi^* v\|_1 = \varphi^* \|v\|_1 = \varphi^* \psi^* = 1$. Hence the point $\varphi^* d$ can be written as a convex combination of points $\pm a_i$ lying in $\mathcal{F}_y$, implying that it also lies on the face.

Our next result says that once we are in the possession of an optimal point of $(P3)$, it is easy to construct optimal solutions to the problems $(P1)$, $(D1)$, $(D'1)$, $(P2)$ and $(D2)$. This is another reason why the former problem deserves special attention. Given the detailed discussion of the various connections between the problems, there is a number of ways to proving the result. For example, we have seen in the proof of Proposition 3.2.23 that if $w$ satisfies the optimality conditions $(i)$ and $(ii)$, then $x = y/\psi^2(w)$ is optimal for $(P1)$. Alternatively, this can be proved by constructing a feasible point for $(D2)$, as we will do in the proof below, such that the complementary slackness condition formulated in Proposition 3.2.8 holds. In doing so, we automatically obtain an optimal point for $(D2)$. It is not

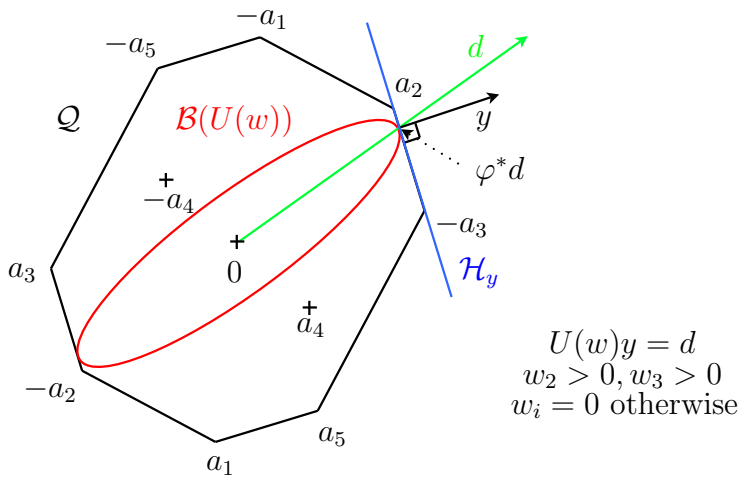



Figure 3.7: Geometry at optimality.

the intention of the author to give an exhaustive list of the many possible proofs. In fact, we will do quite the opposite: cut through the argument as fast as possible using the help of the results we have proved in previous sections.

**Theorem 3.2.24** (Universal optimality condition)**.** *Assuming that the optimality conditions* $(i)$ *and* $(ii)$ *from Proposition 3.2.23 hold for some* $w \in \Delta_m$ *and* $y \in \mathbf{E}$*, then*

*(i)* $x = y/\psi^2(w)$ *is a minimizer of* $(P1)$*,*

*(ii)* $\tau = 1/\psi(w)$ *is the maximizer of* $(D1)$*,*

*(ii')* $\tau' = \varphi(y)/\psi^2(w) = \varphi(x)$ *is the minimizer of* $(D'1)$*,*

*(iii)* $z = y/\varphi(y)$ *is a maximizer of* $(P2)$*,*

*(iv)* $v \in \mathbf{R}^m$ *given by* $v_i = w_i\langle a_i, y\rangle$ *is a minimizer of* $(D2)$*, and*

*(v)* $w$ *is a minimizer of* $(P3)$*.*

*Proof.* Parts $(i)$ and $(iv)$ follow from Proposition 3.2.8, $(iii)$ from optimality of $x$ and Proposition 3.2.6. Statement $(v)$ is implied by Proposition 3.2.23 and $(ii)$ is then implied by optimality of $w$ and $(ii')$ by optimality of $x$. □

Observe that whenever $\varphi(y) = \psi(w)$, the values of $\tau$ and $\tau'$ are equal. The reason for defining the former as $1/\psi(w)$ and the latter as $\varphi(x)$ is because for any $w$ feasible for $(P3)$ the value $1/\psi(w)$ gives a lower bound on $\varphi^*$, thus producing a feasible point for $(D1)$, while for any $x$ feasible for $(P1)$ the value $\varphi(x)$ always yields an upper bound on $\varphi^*$, giving a feasible point for $(D'1)$. This distinction will be needed in the formulation of Theorem 3.2.25.

We can hardly expect from an algorithmic scheme for solving $(P3)$ to yield an exact minimizer. In this sense, Theorem 3.2.24 is not practical. Also, it would be good to be able to say something about the quality of the current solution because this could suggest what work needs to be done to obtain the next iterate. At the same time, we would like to be able to say something about the quality of the derived points $x, \tau, z$ and $v$ in their respective problems even if the current point $w$ is not optimal, but perhaps nearly optimal. Theorem 3.2.25 below states that a relaxed version of the optimality conditions, in view of inequality (3.26), gives the right answer.

**Theorem 3.2.25** (Approximate universal optimality condition)**.** *Let $U(w)y = d$ for some $w \in \Delta_m$ and $y \in \mathbf{E}$ and assume the following $\delta$-approximate optimality condition holds:*

$$\varphi(y) \leq (1 + \delta)\psi(w). \tag{3.27}$$

*Then the points $x, \tau, \tau', z, v$ and $w$ considered in Theorem 3.2.24 are feasible and satisfy the following $\delta$-optimality conditions in their respective problems:*

*(i)* $\varphi(x) \leq (1+\delta)\varphi^*$,

*(ii)* $\tau \geq (1+\delta)^{-1}\varphi^* \geq (1-\delta)\varphi^*$,

*(ii')* $\tau' \leq (1+\delta)\varphi^*$,

*(iii)* $\langle d, z\rangle \geq (1+\delta)^{-1}\psi^* \geq (1-\delta)\psi^*$,

*(iv)* $\|v\|_1 \leq (1+\delta)\psi^*$, *and*

*(v)* $\psi(w) \leq (1+\delta)\psi^*$.

*Moreover,* $\|v\|_1 \leq \psi(w) \leq (1+\delta)\langle d, z\rangle$ *and* $\varphi(x)\langle d, z\rangle = 1$.

*Proof.* Feasibility in each case follows from the definition of the corresponding point: note that the proof of the previous theorem did not use the optimality condition to establish feasibility. Since $x = y/\psi^2(w)$ and $\psi(w) \geq \psi^* = 1/\varphi^*$, condition (3.27) yields

$$\varphi(x) \leq (1+\delta)\frac{1}{\psi(w)} \leq (1+\delta)\varphi^*,$$

establishing $(i)$. Part $(ii)'$ then follows from $(i)$ as $\tau' = \varphi(x)$. Reversing the first of the displayed inequalities above gives $(v)$ since $\varphi(x) \geq \varphi^*$. By definition, $\tau = 1/\psi(w)$ and hence $(ii)$ can be obtained from $(v)$ by taking the reciprocals and substituting $\psi^* = 1/\varphi^*$. Part $(iii)$ follows from $(i)$ by noting that $z = y/\varphi(y)$, $x = y/\psi^2(w)$ and $\langle d, x\rangle = 1$ implies

$$\langle d, z\rangle = \frac{\langle d, y\rangle}{\varphi(y)} = \frac{\langle d, x\rangle\psi^2(w)}{\varphi(x)\psi^2(w)} = \frac{1}{\varphi(x)}.$$

Inequality $(iv)$ follows from $(v)$ and Lemma 3.2.12. The final statement can be easily extracted from the proof. □

## 3.3 Algorithms

In this section we will put to work the theory developed in the preceding text to design and analyze algorithms for finding a $\delta$-approximate solution to problem $(P3)$. In view of Theorem 3.2.25, we are simultaneously solving several other problems. The development in this section can be viewed as the continuation of the effort for combining the rounding and optimization steps for solving problem $(P1)$ initiated in Section 2.5 of Chapter 2.

### 3.3.1 A multiplicative weight update algorithm

The inequalities formulated as Lemma 3.2.12 and Lemma 3.2.13 reveal a close relationship between the feasible solutions of problems $(D2)$ and $(P3)$. As we have seen in the previous section, these two lemmas can be used to argue that the two problems have equal optima (see Theorem 3.2.14). However, they are more interesting to us because of their geometry (see Figure 3.2.4) and algorithmic implications.

Assuming we start with a feasible solution to problem $(P3)$, Lemma 3.2.12 provides us with a feasible solution to problem $(D2)$ with a better objective value. Now in turn, starting from this feasible solution, Lemma 3.2.13 gets us back to a feasible solution to problem $(P3)$, again with a better objective value. Using this observation we have arrived at our first algorithm of this chapter (Algorithm 10), updating the weights in a multiplicative fashion at every iteration.

Due to their simplicity, multiplicative weight update algorithms have been proposed in the literature for many computer science problems. For a recent unifying review of such approaches we refer the reader to [2].

**Algorithm 10 (MultWeight)** Multiplicative weight updates

**Input:** $a_1, \ldots, a_m \in \mathbf{E}^*$, $d \in \mathbf{E}^*$, $\delta > 0$;

**Initialize:** $k = 0,\ w_0 = e/m$;

**Iterate:**

  $U_k = \sum_i w_k^{(i)} a_i a_i^*,\ y_k = U_k^{-1} d$;

  $\alpha_k = \langle d, y_k \rangle,\ j = \arg\max_i |\langle a_i, y_k \rangle|,\ \beta_k = \langle a_j, y_k \rangle$;

  $\delta_k = \frac{|\beta_k|}{\sqrt{\alpha_k}} - 1$;

  **if** $\delta_k \leq \delta$

    **terminate;**

  **else**

    $v_i = w_k^{(i)} \langle a_i, y_k \rangle,\ i = 1, 2, \ldots, m$;

    $w_{k+1} = |v| / \|v\|_1$;

    $k \leftarrow k + 1$;

  **end if**

**Output:** $w_k$ satisfying $\|d\|^*_{U(w_k)} = \sqrt{\alpha_k} \leq (1 + \delta)\psi^*$;

Notice that the stopping criterion of Algorithm 10 is equivalent to the condition (3.27) of Theorem 3.2.25 and hence the point $w_k$ output by the algorithm, if it terminates, is a $\delta$-approximate minimizer of problem $(P3)$. The algorithm, however, suffers from at least two shortcomings.

First, it can fail to terminate (see Figure 3.8). In short, this is because once a weight is set to zero, it can never be increased to a nonzero value, even if the corresponding point $a_i$ is required to have a positive weight in the optimum. Imagine we run the algorithm starting with positive weights only for $i \in \mathcal{I}$ with $\mathcal{I}$ being a proper subset of the index set $\{1, 2, \ldots, m\}$. It is clear that the algorithm will never be able to work with points $a_i$ with $i \notin \mathcal{I}$ and hence we are actually at best trying to find the intersection of the half-line emanating from the origin in the direction $d$ and the convex hull of $\{\pm a_i,\ i \in \mathcal{I}\}$, which is a proper subset of $\mathcal{Q}$. If the algorithm happens to drop weights to zero along the way, it will never be able to recover them back to a nonzero value. Because $\mathcal{I}_{k+1} \subseteq \mathcal{I}_k$ holds for all $k$, the "scope" of the method will be the gradually diminishing convex set $\mathcal{Q}_{\mathcal{I}_k} = \text{conv}\{\pm a_i \ :\ i \in \mathcal{I}_k\}$.

Another obvious disadvantage of the algorithm is its high computational cost per iteration due to the need to update $U$ in a full-rank fashion. The inverse (or a factorization) of $U$ will therefore have to be fully recomputed at every iteration at the cost of at least $O(n^3)$ arithmetic operations.

### 3.3.2 Ingredients of a rank-one update algorithm

As we have already mentioned above, the multiplicative weight update algorithm has the obvious disadvantage of altering weights in a rather *nonuniform* way, resulting in the need to *fully* resolve a system of the form $U(w)y = d$ at every

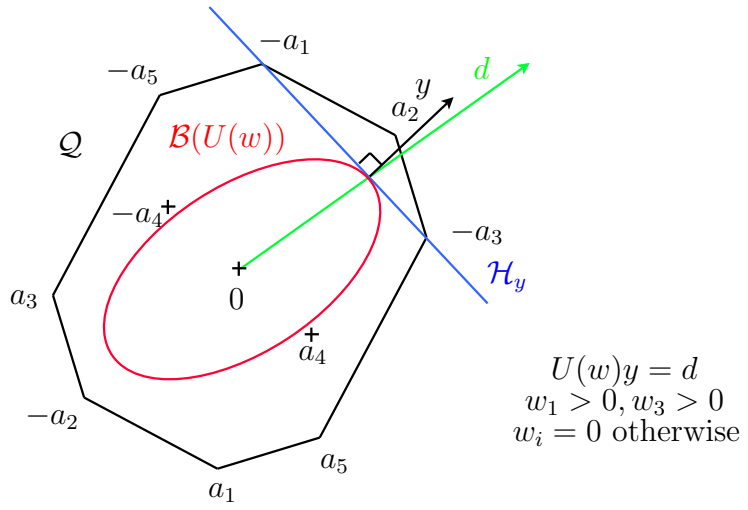



Figure 3.8: Algorithm 10 can fail to converge to the optimum.

iteration. The idea we are going to exploit now is updating $U(w)$ only slightly at every iteration, in a rank-one fashion. This corresponds to changing the weight of a specific term $a_j a_j^*$ and then adjusting all other weights *uniformly* by a certain factor, so as to keep the resulting vector of weights feasible.

In what follows we will focus on a single iteration with current weight $w$. Assume throughout that $\psi(w) < \infty$, or equivalently,

$$d \in \operatorname{range} U(w), \qquad \text{(Assumption 1)}$$

and that we are in possession of vector $y$ such that $U(w)y = d$. Suppose we update this weight to

$$w(\kappa) := \frac{w + \kappa e_j}{1 + \kappa}, \tag{3.28}$$

where $\kappa$ is a real parameter, $j \in \{1, 2, \ldots, m\}$ is to be determined later and $e_j$ is the $j$-th unit vector of $\mathbf{R}^m$. The smallest possible $\kappa$ for which $w(\kappa)$ is feasible is $\kappa_{min} := -w_j$. For $w(\kappa)$ to be meaningfully defined, we will further suppose that

$$w_j \neq 1. \qquad \text{(Assumption 2)}$$

This ensures both that $w(\kappa)$ varies as $\kappa$ varies and that $w(-w_j)$ is well-defined. We allow $\kappa$ to take on the value $\infty$ and naturally define $w(\infty) := e_j$. Note that the set of weights described this way forms a chord of $\Delta_m$ joining vertex $e_j$ with $w$ (see Figure 3.3.2). We chose this particular parametrization of the chord over the more natural $w(\lambda) := (1-\lambda)w + \lambda e_j,\ 0 \le \lambda \le 1$, because it turns out to yield a more compact exact line-search formula, developed in the next subsection. By linearity of $U(\cdot)$ as a function of $w$, this translates into updating $U(w)$ (for simplicity we will write just $U$) as follows:

$$U(\kappa) := U(w(\kappa)) = \frac{U + \kappa a_j a_j^*}{1+\kappa}. \tag{3.29}$$

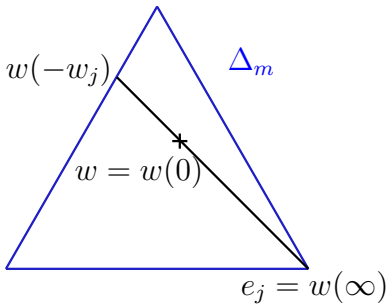



Figure 3.9: The weights $w(\kappa)$ for $\kappa \in [-w_j, \infty]$.

After we update $w$ to $w(\kappa)$, the value $\psi^2(w) = \langle d, y\rangle$ changes to

$$\psi^2(\kappa) := \psi^2(w(\kappa)) = \begin{cases} \langle d, y(\kappa)\rangle & \text{if} \quad y(\kappa) \text{ solves } U(\kappa)y(\kappa) = d, \\ \infty & \text{if} \quad d \notin \operatorname{range} U(\kappa). \end{cases}$$

Of course, we consider only updates decreasing the objective value at every iteration and hence the second case does not apply for $\kappa$ we actually end up using.

Since $U$ changes in a simple and highly structured way (rank-one update and scaling), it can be expected that $y(\kappa)$ should be obtainable from $y$ with less effort

than resolving from scratch. This is indeed the case. If both $U$ and $U(\kappa)$ are nonsingular, one can use the Sherman-Morrison formula (see, for example, Section 2.1.3 of [10]) to this purpose. Loosely speaking, the formula says that the inverse of a rank-one perturbation of a nonsingular matrix results in a rank-one perturbation of the inverse:

**Fact 3.3.1** (Sherman-Morrison)**.** *If $U\colon \mathbf{E} \to \mathbf{E}^*$ is an invertible linear operator, $g \in \mathbf{E}^*$ and $\kappa \in \mathbf{R}$ such that $1 + \kappa\langle g, U^{-1}g\rangle \neq 0$, then $U + \kappa gg^*$ is invertible and*

$$(U + \kappa gg^*)^{-1} = U^{-1} - \frac{\kappa U^{-1}gg^*U^{-1}}{1 + \kappa\langle g, U^{-1}g\rangle}.$$

In the definition of $U(\kappa)$ we are dealing with a rank-one update followed by scaling. In particular, if $U = U(w)$ is invertible and $1 + \kappa\langle a_j, U^{-1}a_j\rangle \neq 0$ with $\kappa$ being a real number, the Sherman-Morrison formula implies

$$y(\kappa) := U(\kappa)^{-1}d = (1+\kappa)\left(y - \frac{\kappa U^{-1}a_j\langle a_j, y\rangle}{1 + \kappa\langle a_j, U^{-1}a_j\rangle}\right),$$

and hence

$$\psi^2(\kappa) = \langle d, y(\kappa)\rangle = (1+\kappa)\left(\langle d, y\rangle - \frac{\kappa\langle a_j, y\rangle^2}{1 + \kappa\langle a_j, U^{-1}a_j\rangle}\right). \tag{3.30}$$

In the remainder of this subsection we compute a general formula for $\psi^2(\kappa)$, one that is free of the full-rank assumption on $U$ and includes the case $\kappa = \infty$ and the situation when the expression in the denominator of (3.30) vanishes. We proceed through several auxiliary results — the first step is the following simple generalization of the Sherman-Morrison inversion identity:

**Lemma 3.3.2.** *Let $U\colon \mathbf{E} \to \mathbf{E}^*$ be a (not necessarily invertible) linear operator and assume $Uy = d$ for some $y \in \mathbf{E}$ and $d \in \mathbf{E}^*$. If for $g \in \mathbf{E}^*$ and $\kappa \in \mathbf{R}$ we let*

$$\tilde{y}(\kappa) := \begin{cases} y & \text{if} \quad \langle g, y\rangle = 0, \\ y - \frac{\kappa\langle g, y\rangle x}{1+\kappa\langle g, x\rangle} & \text{if} \quad Ux = g \quad \text{and} \quad 1 + \kappa\langle g, x\rangle \neq 0 \quad \text{for some} \quad x \in \mathbf{E}, \end{cases}$$

*then* $(U + \kappa gg^*)\tilde{y}(\kappa) = d$.

*Proof.* The first case is trivial; the statement in the second case follows from:

$$\begin{aligned}(U + \kappa gg^*)\tilde{y}(\kappa) &= Uy + \kappa\langle g, y\rangle g - \frac{\kappa\langle g, y\rangle Ux}{1 + \kappa\langle g, x\rangle} - \frac{\kappa^2\langle g, x\rangle\langle g, y\rangle g}{1 + \kappa\langle g, x\rangle} \\ &= d + \kappa\langle g, y\rangle g\left(1 - \frac{1 + \kappa\langle g, x\rangle}{1 + \kappa\langle g, x\rangle}\right) \\ &= d.\end{aligned}$$

□

**Remark 3.3.3.** *Note that if* $U$ *is self-adjoint, the value* $\langle g, x\rangle$ *in the above lemma does not depend on the particular choice of the solution of the system* $Ux = g$. *Indeed, if* $x'$ *and* $x''$ *are two such solutions, then* $\langle g, x'\rangle = \langle Ux'', x'\rangle = \langle Ux', x''\rangle = \langle g, x''\rangle$. *This is precisely one of the two arguments we used to show that (3.6) gives a valid definition of* $\|g\|_U^*$. *If we also have* $U \succeq 0$, *then* $\langle g, x\rangle = (\|g\|_U^*)^2$, *which is positive unless* $g = 0$.

The next result characterizes the family of rank-one self-adjoint perturbations of a positive semidefinite self-adjoint operator preserving positive-semidefiniteness.

**Lemma 3.3.4.** *Let* $U\colon \mathbf{E} \to \mathbf{E}^*$ *be a positive semidefinite self-adjoint operator and consider* $g \in \mathbf{E}^*$ *and a real parameter* $\kappa$.

*(i) If* $g \in \operatorname{range} U$, *then*

$$U + \kappa gg^* \succeq 0 \quad \Leftrightarrow \quad 1 + \kappa(\|g\|_U^*)^2 \geq 0.$$

*(ii) If* $g \notin \operatorname{range} U$, *then*

$$U + \kappa gg^* \succeq 0 \qquad \Leftrightarrow \qquad \kappa \geq 0.$$

*Proof.* We offer two proofs. *First proof.* The statements trivially hold if $\kappa \geq 0$. If we notice that $\|g\|_U^* = \infty$ precisely when $g \notin \operatorname{range} U$, the case with $\kappa < 0$ is essentially a restatement of the equivalence between $(i)$ and $(iii)$ of Lemma 3.2.18 with $\tau := -1/\kappa > 0$.

*Second proof.* To establish $(i)$, let $x$ be any vector satisfying $Ux = g$. Observe that if $U + \kappa gg^* \succeq 0$, we have

$$0 \leq \langle (U + \kappa gg^*)x, x\rangle = \langle Ux, x\rangle + \kappa \langle g, x\rangle^2 = \langle g, x\rangle(1 + \kappa\langle g, x\rangle).$$

This proves the direct implication if we further notice that $\langle g, x\rangle = (\|g\|_U^*)^2 > 0$ unless $g = 0$. If $g = 0$, the reverse implication is trivial. Assume therefore that $(\|g\|_U^*)^2 > 0$ and consider any $z \in \mathbf{E}$. The gauge Cauchy-Schwarz inequality (Corollary 3.2.3) gives

$$\frac{\langle g, z\rangle^2}{(\|g\|_U^*)^2} \leq \frac{(\|g\|_U^*)^2\|z\|_U^2}{(\|g\|_U^*)^2} = \langle Uz, z\rangle$$

and hence

$$\langle (U + \kappa gg^*)z, z\rangle = \langle Uz, z\rangle + \kappa\langle g, z\rangle^2 \geq 0,$$

whenever $\kappa \geq -1/(\|g\|_U^*)^2$.

For the direct implication in $(ii)$ note that $\operatorname{range} U$ consists precisely of those functionals that vanish on $\operatorname{null} U$ and hence there must exist $z \in \operatorname{null} U$ such that $\langle g, z\rangle > 0$. Therefore,

$$0 \leq \langle (U + \kappa gg^*)z, z\rangle = \kappa\langle g, z\rangle^2,$$

which gives $\kappa \geq 0$. The reverse implication is straightforward. □

**Corollary 3.3.5.** *If $w(\kappa)$ is feasible, then $\kappa \geq -1/(\|a_j\|_U^*)^2$ and in particular $-w_j \geq -1/(\|a_j\|_U^*)^2$. If $w_j > 0$, then $(\|a_j\|_U^*)^2 \leq 1/w_j$.*

*Proof.* Observe that $w(\kappa)$ being feasible implies $1+\kappa \geq 1-w_j > 0$ and $U(\kappa) \succeq 0$ and hence $U+\kappa a_j a_j^* = (1+\kappa)U(\kappa) \succeq 0$. Now use Lemma 3.3.4 with $g = a_j$. □

**Remark 3.3.6.** *Note that if we subscribe to the convention that*

$$\kappa \times \infty = \begin{cases} -\infty & \text{if} \quad \kappa < 0, \\ 0 & \text{if} \quad \kappa = 0, \text{ and} \\ +\infty & \text{if} \quad \kappa > 0, \end{cases}$$

*the second case of Lemma 3.3.4 gets subsumed by the first.*

**Proposition 3.3.7.** *Let $U\colon \mathbf{E} \to \mathbf{E}^*$ be a positive semidefinite self-adjoint operator and assume $Uy = d$ for some $y \in \mathbf{E}$ and $0 \neq d \in \mathbf{E}^*$.*

*(i) If $0 \neq g \in \operatorname{range} U$ then for $\kappa \geq -1/(\|g\|_U^*)^2$ the operator $U+\kappa gg^*$ is positive semidefinite and*

$$\|d\|^*_{U+\kappa gg^*} = \begin{cases} \sqrt{(\|d\|_U^*)^2 - \frac{\kappa\langle g,y\rangle^2}{1+\kappa(\|g\|_U^*)^2}} & \text{if} \quad \kappa > -\frac{1}{(\|g\|_U^*)^2}, \\ \|d\|_U^* & \text{if} \quad \kappa = -\frac{1}{(\|g\|_U^*)^2},\ \langle g,y\rangle = 0, \\ \infty & \text{if} \quad \kappa = -\frac{1}{(\|g\|_U^*)^2},\ \langle g,y\rangle \neq 0. \end{cases} \tag{3.31}$$

*Moreover,*

$$\|d\|^*_{U+\kappa gg^*} \to \begin{cases} \frac{\sqrt{(\|d\|_U^*)^2(\|g\|_U^*)^2 - \langle g,y\rangle^2}}{\|g\|_U^*} & \text{as} \quad \kappa \to \infty, \\ \infty & \text{as} \quad \kappa \downarrow -\frac{1}{(\|g\|_U^*)^2} \quad \text{if} \quad \langle g,y\rangle \neq 0. \end{cases}$$

*(ii) If $g \notin \operatorname{range} U$, then for $\kappa \geq 0$ the operator $U+\kappa gg^*$ is positive semidefinite and*

$$\|d\|^*_{U+\kappa gg^*} = \|d\|_U^* \quad \text{for all} \quad \kappa \geq 0. \tag{3.32}$$

*Proof.* Consider statement $(i)$ and note that $g \neq 0$ implies $\|g\|_U^* > 0$. Positive semidefiniteness of $U + \kappa gg^*$ follows from part $(i)$ of Lemma 3.3.4. If $\tilde{y}(\kappa)$ and $x$ are as in Lemma 3.3.2, then $(\|d\|^*_{G+\kappa gg^*})^2 = \langle d, \tilde{y}(\kappa)\rangle$ and $(\|g\|_U^*)^2 = \langle g, x\rangle$, and hence the first two cases of (3.31) follow. Assume now that $\kappa = -1/(\|g\|_U^*)^2$ and $\langle g, y\rangle \neq 0$. We will show that this implies $d \notin \text{range}(U + \kappa gg^*)$, and hence $\|d\|^*_{U+\kappa gg^*} = \infty$, by demonstrating that $x' := x/\langle g, y\rangle$ satisfies $\langle d, x'\rangle = 1$ and $\langle (U + \kappa gg^*)x', x'\rangle = 0$ and then appealing to Proposition 3.2.5. Indeed,

$$\langle d, x'\rangle = \frac{\langle d, x\rangle}{\langle g, y\rangle} = \frac{\langle Uy, x\rangle}{\langle g, y\rangle} = \frac{\langle Ux, y\rangle}{\langle g, y\rangle} = 1$$

and

$$\begin{aligned}
\langle (U + \kappa gg^*)x', x'\rangle &= \tfrac{1}{\langle g,y\rangle^2}\langle (U + \kappa gg^*)x, x\rangle \\
&= \tfrac{1}{\langle g,y\rangle^2}(\langle g, x\rangle + \kappa\langle g, x\rangle^2) \\
&= \tfrac{(\|g\|_U^*)^2}{\langle g,y\rangle^2}(1 + \kappa(\|g\|_U^*)^2) \\
&= 0.
\end{aligned}$$

The proof of the limit statements is straightforward. To establish $(ii)$, fix arbitrary nonnegative $\kappa$ and note that whenever some $\tilde{y}(\kappa)$ satisfies $(U + \kappa gg^*)\tilde{y}(\kappa) = d$, we have $\kappa\langle g, \tilde{y}(\kappa)\rangle g = d - U\tilde{y}(\kappa) \in \text{range}\, U$. This is possible if and only if $\kappa\langle g, \tilde{y}(\kappa)\rangle = 0$ and $U\tilde{y}(\kappa) = d$. It therefore follows that $\|d\|^*_{U+\kappa gg^*} = \langle d, \tilde{y}(\kappa)\rangle^{1/2} = \|d\|_U^*$. □

The main result of this subsection gives a complete characterization of $\psi^2(\kappa)$, generalizing (3.30).

**Theorem 3.3.8.** *Assume $y \in \mathbf{E}$ is such that $Uy = d$ ($U = U(w)$) and let us*

*establish the following simplified notation*[2]*:*

$$\alpha := \langle d, y\rangle = (\|d\|_U^*)^2 = \psi^2(w), \qquad \beta := \langle a_j, y\rangle, \qquad \gamma := (\|a_j\|_U^*)^2.$$

*(i) If* $a_j \in \operatorname{range} U$ *and* $-1 < -w_j \le \kappa \le \infty$*, then the operator* $U(\kappa) = (U + \kappa a_j a_j^*)/(1+\kappa)$ *is positive semidefinite and self-adjoint and* $\psi^2(\kappa) = (\|d\|_{U(\kappa)}^*)^2$ *can be written explicitly in terms of* $\alpha, \beta, \gamma$ *and* $\kappa$ *as follows:*

$$\psi^2(\kappa) = \begin{cases} (1+\kappa)\left(\alpha - \frac{\kappa\beta^2}{1+\kappa\gamma}\right) & \text{if} \quad \infty > \kappa > -1/\gamma, \\ (1+\kappa)\alpha & \text{if} \quad \kappa = -1/\gamma = -w_j,\ \beta = 0, \\ \infty & \text{if} \quad \kappa = -1/\gamma = -w_j,\ \beta \neq 0, \\ \infty & \text{if} \quad \kappa = \infty,\ \alpha\gamma > \beta^2, \\ \frac{\alpha}{\gamma} & \text{if} \quad \kappa = \infty,\ \alpha\gamma = \beta^2. \end{cases} \tag{3.33}$$

*Moreover,* $\psi^2$ *enjoys the following continuity/barrier properties:*

$$\psi^2(\kappa) \to \begin{cases} \infty & \text{as} \quad \kappa \downarrow -w_j \quad \text{if} \quad \beta \neq 0,\ -w_j = -\frac{1}{\gamma}, \\ \infty & \text{as} \quad \kappa \to \infty \quad \text{if} \quad \alpha\gamma > \beta^2, \\ \frac{\alpha}{\gamma} & \text{as} \quad \kappa \to \infty \quad \text{if} \quad \alpha\gamma = \beta^2. \end{cases} \tag{3.34}$$

*(ii) If* $a_j \notin \operatorname{range} U$*, and* $0 \le \kappa \le \infty$*, then the operator* $U(\kappa)$ *is positive semidefinite and self-adjoint and* $\psi^2(\kappa)$ *can be written as follows:*

$$\psi^2(\kappa) = \begin{cases} (1+\kappa)\alpha & \text{if} \quad \infty > \kappa \ge 0, \\ \infty & \text{if} \quad \kappa = \infty. \end{cases} \tag{3.35}$$

*Moreover,* $\psi^2(\kappa) \to \infty$ *as* $\kappa \to \infty$*.*

[2]The symbols $\alpha, \beta$ and $\gamma$ are not meant to concur with the notation used in Chapter 2. For example, $\alpha$ is not related to quality of any ellipsoidal rounding and $\gamma$ does not refer to a Lipschitz constant.

*Proof.* Let us start with part $(i)$ and observe that $-1/\gamma \leq -w_j$ (Corollary 3.3.5). The first three cases of (3.33) follow from Proposition 3.3.7 used with $g = a_j$ since

$$\psi^2(\kappa) = (\|d\|^*_{U(\kappa)})^2 = (1+\kappa)(\|d\|^*_{U+\kappa a_j a_j^*})^2.$$

The first limit case of (3.34) corresponds to a case from Proposition 3.3.7 while the other two can be easily derived by taking the limit in the first expression of (3.33).

It remains to analyze the $\kappa = \infty$ cases. First observe that $U(\infty) = a_j a_j^*$ and that $\alpha\gamma \geq \beta^2$ by the gauge Cauchy-Schwarz inequality (Corollary 3.2.3)

$$\beta^2 = \langle a_j, y\rangle^2 \leq (\|a_j\|^*_U)^2 \|y\|^2_U = \gamma\alpha, \tag{3.36}$$

with equality if and only if either $a_j$ or $-a_j$ is a nonnegative multiple of $Uy = d$ (i.e. $a_j$ and $d$ are collinear). Consider the equality case and assume $d = \tau a_j$. Since the ellipsoid $\mathcal{B}(a_j a_j^*)$ corresponds to the line segment $[-a_j, a_j]$, it must be the case that $\|d\|^*_{a_j a_j^*} = |\tau|$. This can be also seen without referring to the geometrical picture as follows: If $y'$ is such that $(a_j a_j^*)y' = d$, then $\tau = \langle a_j, y'\rangle$ and

$$(\|d\|^*_{a_j a_j^*})^2 = \langle d, y'\rangle = \langle \tau a_j, y'\rangle = \tau^2.$$

If $\langle a_j, y\rangle^2 = \beta^2 = \alpha\gamma > 0$, we can write

$$\|d\|^*_{U(\infty)} = \tau = \left|\frac{\tau\langle a_j, y\rangle}{\langle a_j, y\rangle}\right| = \frac{\langle d, y\rangle}{|\langle a_j, y\rangle|} = \frac{\alpha}{|\beta|},$$

and hence

$$\psi^2(\infty) = (\|d\|^*_{U(\infty)})^2 = \frac{\alpha^2}{\beta^2} = \frac{\alpha^2}{\alpha\gamma} = \frac{\alpha}{\gamma}.$$

In the remaining case $a_j$ and $d$ are not collinear and thus $d \notin \text{range}(U(\infty))$, implying $\psi^2(\infty) = \infty$.

The first statement of part $(ii)$ is a consequence of part $(ii)$ of Proposition 3.3.7. The second statement can be proved in complete analogy to the fourth case of (3.33). This is because $d \in \operatorname{range} U$ and $a_j \notin \operatorname{range} U$ and hence $d$ and $a_j$ can not be collinear, implying $\alpha\gamma > \beta^2$. □

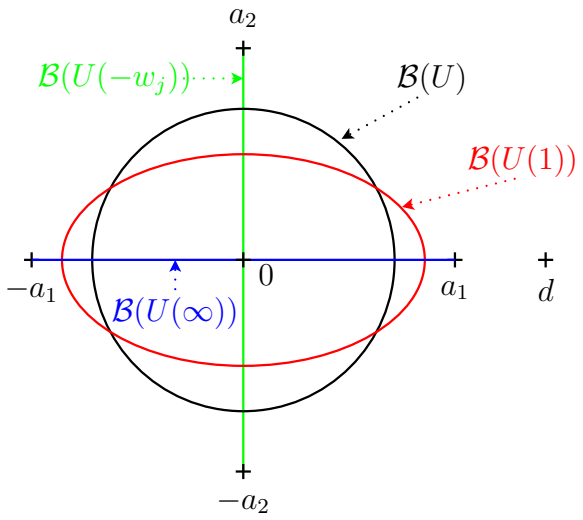



Figure 3.10: Geometry of line-search (Example 3.3.9).

**Example 3.3.9.** Consider the following simple example with $\mathbf{E} = \mathbf{R}^2$ (hence $n = 2$) and $m = 2$ (see Figure 3.10). Let $a_1 = (\sqrt{2}, 0)^T, a_2 = (0, \sqrt{2})^T$ and $d = (2, 0)$, with the current weights being $w_1 = w_2 = \frac{1}{2}$. This means that $U := U(w) = w_1 a_1 a_1^T + w_2 a_2 a_2^T = I$ and hence $\mathcal{B}(U)$ is the unit ball in $\mathbf{R}^2$. Consider updating the weight of $a_j = a_1$ and note that in this case

$$\alpha = \langle d, y \rangle = \langle d, U^{-1} d \rangle = \langle d, d \rangle = 4,$$

$$\beta = \langle a_j, y \rangle = \langle a_j, U^{-1} d \rangle = 2\sqrt{2},$$

and

$$\gamma = (\|a_j\|_U^*)^2 = 2.$$

Let us analyze the following choices for the update parameter $\kappa$:

1. $\kappa := -w_j = -\frac{1}{2}$. Note that $U(\kappa) = (U + \kappa a_j a_j^T)/(1+\kappa) = a_2 a_2^T = \left(\begin{smallmatrix} 0 & 0 \\ 0 & 2 \end{smallmatrix}\right)$ and hence $\mathcal{B}(U(\kappa))$ is the one dimensional ellipsoid corresponding to the line segment joining $-a_2$ and $a_2$. Also note that $-w_j = -\frac{1}{\gamma}$ and hence by (3.33) we have

$$(\|d\|^*_{U(\kappa)})^2 = \psi^2(\kappa) = \infty.$$

It is geometrically clear that this should be case since the vector $d$ "sticks out" of the span of $\mathcal{B}(U(\kappa))$.

2. $-\frac{1}{\gamma} < \kappa < \infty$. In this case $\mathcal{B}(U(\kappa))$ remains being a full-dimensional ellipsoid. If the weight on $a_j$ is increased (corresponding to $\kappa > 0$), then $\mathcal{B}(U(\kappa))$ prolongs in the $x$-axis direction and shrinks in the $y$-axis direction (see $\mathcal{B}(U(1))$ in Figure 3.10), meaning that the norm of $d$ as measured by $U(\kappa)$ decreases. Indeed, by (3.33) we get

$$(\|d\|^*_{U(\kappa)})^2 = \psi^2(\kappa) = (1+\kappa)\left(\alpha - \frac{\kappa\beta^2}{1+\kappa\gamma}\right) = 2 + \frac{2}{1+\kappa},$$

for $\kappa \geq 0$. On the other hand, if the weight on $a_j$ is decreased (corresponding to $\kappa < 0$), then $\mathcal{B}(U(\kappa))$ shrinks in the $x$-axis direction and expands in the $y$-axis direction, meaning that the norm of $d$ as measured by $U(\kappa)$ increases.

3. $\kappa = \infty$. This choice leads to $\mathcal{B}(U(\kappa))$ corresponding to the line segment joining $-a_1$ and $a_1$. Geometrically, the norm of $d$ should therefore drop to $2/\sqrt{2} = \sqrt{2}$. Let us verify this by computation. Since $\alpha\gamma = 8 = \beta^2$, formula (3.33) tells us that

$$(\|d\|^*_{U(\kappa)})^2 = \psi^2(\kappa) = \frac{\alpha}{\gamma} = \frac{4}{2} = 2.$$

This corresponds to the optimal choice of $\kappa$ minimizing the value of $\psi^2(\kappa) = (\|d\|^*_{U(\kappa)})^2$.

**The choice of $j$.** We need to address two questions. First, how do we pick the index $j$? Once this is chosen, we want to find the steplength $\kappa$ minimizing $\psi^2(\kappa)$.

Our initial motivation for the choice of $j$ can be drawn from the multiplicative weight-update rule. At every iteration of Algorithm 10, the weights $w_i$ are multiplied by the factor $|\langle a_i, y\rangle|$ and then re-normalized. If this value is relatively large (or small) for particular $i$, the corresponding weight is being updated by a relatively large (or small) factor and is likely to have a substantial effect. It therefore makes sense to consider

$$j^+ := \arg\max_i |\langle a_i, y\rangle| \qquad \text{and} \qquad j^- := \arg\min_{w_i>0} |\langle a_i, y\rangle|. \tag{3.37}$$

Notice that $\varphi(y) = |\langle a_{j^+}, y\rangle|$ and that either $a_j \in \partial\varphi(y)$ or $-a_j \in \partial\varphi(y)$, depending on whether $\varphi(y) = \langle a_j, y\rangle$ or $\varphi(y) = \langle -a_j, y\rangle$.

### 3.3.3 Line search

In this subsection we consider the following line-search problem:

$$\kappa^* := \arg\min\{\psi^2(\kappa) \ : \ \kappa_{min} = -w_j \le \kappa \le \infty\}. \tag{3.38}$$

Note that if $a_j \notin \operatorname{range} U$ then $w_j = 0$ by Proposition 3.2.9. In this case, however, the line-search problem is trivial with the optimal step size being $\kappa^* = 0$ (see Theorem 3.3.8). We will therefore henceforth assume that $a_j \in \operatorname{range} U$.

#### General line-search formula

Our main result in this subsection is Theorem 3.3.10, in which we give a closed-form formula for the solution of (3.38). We then specialize this formula for $j = j^+$ and $j = j^-$, as defined in (3.37).

We will assume, as in Theorem 3.3.8, that $w_j \neq 1$ and $d \in \operatorname{range} U$. The first assumption is in place to ensure that $w(\kappa)$ does not degenerate into describing a single point for all values of $\kappa$ while the second ensures that $\psi^2(0) = \alpha < \infty$. Note that we also have $\alpha > 0$ (because $d \neq 0$). Recall that we assume throughout the chapter that the vectors $a_1, \ldots, a_m$ are all nonzero, and in particular, $a_j \neq 0$. Since also $a_j \in \operatorname{range} U$, we have $\gamma > 0$. Also recall that $\alpha\gamma \geq \beta^2$ (3.36), with equality if and only if $a_j$ and $d$ are collinear.

**Theorem 3.3.10.** *If $a_j \in \operatorname{range} U$, $w_j \neq 1$ and $\alpha, \beta$ and $\gamma$ are as in Theorem 3.3.8, the solution of the line-search problem (3.38) is*

$$\kappa^* = \begin{cases} \kappa_{min} & \text{if } \beta = 0 \text{ or } \gamma \leq 1, \\ \max\{\kappa_{min}, \kappa_1\} & \text{if } \gamma > 1 \text{ and } \alpha\gamma > \beta^2, \\ \infty & \text{if } \gamma > 1 \text{ and } \alpha\gamma = \beta^2, \end{cases} \tag{3.39}$$

*where*

$$\kappa_1 := -\frac{1}{\gamma} + \frac{|\beta|\sqrt{\gamma - 1}}{\gamma\sqrt{\alpha\gamma - \beta^2}}. \tag{3.40}$$

*Moreover, if $-1/\gamma = -w_j$ then $\gamma > 1$ and $\max\{\kappa_{min}, \kappa_1\} = \kappa_1$.*

*Proof.* First note that since $a_j \in \operatorname{range} U$, the function $\psi^2$ is given by (3.33). Let us start by analyzing the (simpler) case when $-1/\gamma < -w_j$, eliminating two of the subcases in (3.33). In view of the behavior of $\psi^2(\kappa)$ as $\kappa$ approaches infinity, we may assume that

$$\psi^2(\kappa) = (1 + \kappa)\left(\alpha - \frac{\beta^2\kappa}{1 + \gamma\kappa}\right) = \frac{1 + \kappa}{1 + \gamma\kappa}[(\alpha\gamma - \beta^2)\kappa + \alpha], \tag{3.41}$$

and work with $\kappa \in [-w_j, \infty)$. If we discover that the infimum is attained "at" $\infty$, we will set $\kappa^* = \infty$. In order not to get lost in the many subcases to follow, let us do some branching of the argument:

1. If $\beta = 0$ then $\psi^2(\kappa) = (1+\kappa)\alpha$, which is nondecreasing, and we can set $\kappa^* = \kappa_{min}$.

2. Assume that $\beta \neq 0$ and notice that

$$(\psi^2)'(\kappa) = \alpha - \beta^2 \frac{\gamma\kappa^2 + 2\kappa + 1}{(1+\gamma\kappa)^2} = \frac{\gamma(\alpha\gamma - \beta^2)\kappa^2 + 2(\alpha\gamma - \beta^2)\kappa + \alpha - \beta^2}{(1+\gamma\kappa)^2}. \tag{3.42}$$

   (a) Let us first consider the degenerate case when the numerator in the expression above fails to be a quadratic. If $\alpha\gamma = \beta^2$, looking at (3.41) we see that $\psi^2$ is increasing if $\gamma < 1$ and hence we can choose $\kappa^* = \kappa_{min}$. If $\gamma = 1$ then $\psi^2$ is constant on $[-w_j, \infty]$ and any choice of $\kappa^*$ is optimal. Finally, if $\gamma > 1$ then $\kappa^* = \infty$.

   (b) Assume that $\alpha\gamma > \beta^2$. The discriminant of the (convex) quadratic in the numerator of (3.42) is $D = 4(\alpha\gamma - \beta^2)\beta^2(\gamma - 1)$. This is nonpositive if $\gamma \leq 1$, in which case the derivative of $\psi^2$ is nonnegative on $(-1/\gamma, \infty) \supset [-w_j, \infty)$. We can therefore choose $\kappa^* = \kappa_{min}$. Henceforth suppose $\gamma > 1$ and let us write down the roots of the quadratic:

$$\kappa_{1,2} = \frac{-(\alpha\gamma - \beta^2) \pm |\beta|\sqrt{(\gamma-1)(\alpha\gamma - \beta^2)}}{\gamma(\alpha\gamma - \beta^2)}.$$

   Notice that

$$\kappa_2 = -\frac{1}{\gamma} - \frac{|\beta|\sqrt{\gamma-1}}{\gamma\sqrt{\alpha\gamma - \beta^2}} < -\frac{1}{\gamma} < -\frac{1}{\gamma} + \frac{|\beta|\sqrt{\gamma-1}}{\gamma\sqrt{\alpha\gamma - \beta^2}} = \kappa_1.$$

   This implies that $\psi^2$ is decreasing on $(-1/\gamma, \kappa_1)$ and then increasing on $(\kappa_1, \infty)$. Since we consider only $\kappa \geq -w_j$, it is clear that $\kappa^* = \max\{\kappa_{min}, \kappa_1\}$.

It remains to analyze the situation with $-1/\gamma = -w_j$. In this case we proceed as above, except we have to take into account also the second and third expression in (3.33) defining $\psi^2$. If $\beta = 0$ then $\psi^2(\kappa) = (1+\kappa)\alpha$ on $[-w_j, \infty)$ and hence we conclude, as above, that $\kappa^* = \kappa_{min}$. Assume henceforth that $\beta \neq 0$. Now because $\psi^2(\kappa) \to \infty = \psi^2(-w_j)$ as $\kappa \downarrow -w_j$, we may proceed exactly as in the detailed analysis above, keeping in mind that $\gamma > 1$, which is a consequence of the assumption $-1 < -w_j = -1/\gamma$. In case 2a this leads to $\kappa^* = \infty$, while in case 2b we now know that $-w_j = -1/\gamma < \kappa_1$ and hence $\kappa^* = \kappa_1$. □

**Line search with $j^+$ and $j^-$**

If we assume that $j$ is chosen to be either $j^+$ or $j^-$, as defined in (3.37), we can get a refined version of the optimal line-search formula. Let us first observe that $\langle a_{j^-}, y\rangle^2 \leq \alpha \leq \langle a_{j^+}, y\rangle^2 = \varphi^2(y)$, which is a simple consequence of the definitions of $j^+$ and $j^-$ and the frequently used identity $\sum w_i \langle a_i, y\rangle^2 = \langle U(w)y, y\rangle = \langle d, y\rangle = \psi^2(w) = \alpha$. Indeed, the above inequalities say that the weighted average of the numbers $\langle a_i, y\rangle^2$ with positive weights $w_i$ cannot be smaller than their minimum or bigger than their maximum. If there is equality in any of the two inequalities, then $\langle a_i, y\rangle^2 = \alpha = \varphi^2(y)$ for all $i$ for which $w_i > 0$, which is equivalent to the optimality condition $(ii')$ of Proposition 3.2.23. So unless the current vector of weights $w$ is optimal, we have

$$\langle a_{j^-}, y\rangle^2 < \alpha < \langle a_{j^+}, y\rangle^2 = \varphi^2(y). \tag{3.43}$$

Consider now the following cases:

1. Assume $j = j^+$. First notice that $\alpha \leq \beta^2$, with equality if and only if $w$ is optimal. The Cauchy-Schwarz inequality $\alpha\gamma \geq \beta^2$ then implies $\gamma \geq 1$ and

hence $\gamma = 1$ implies optimality. Assume therefore that $\gamma > 1$, which excludes the first case in (3.39), and consider two subcases:

(a) Case $\alpha\gamma > \beta^2$. By (3.39) we have $\kappa^* = \max\{\kappa_{min}, \kappa_1\}$. However, we can say a bit more. Noting that $\alpha \leq \beta^2$ is equivalent to $\kappa_1 \geq 0$, we obtain $\kappa^* = \kappa_1$.

(b) Case $\alpha\gamma = \beta^2$. Formula (3.39) implies $\kappa^* = \infty$. We claim that the next iterate (after taking the "infinite" step) will be optimal. Indeed, $U^+ := U(\kappa^*) = a_j a_j^*$ and if we let $y^+$ satisfy $U^+ y^+ = d$, then

$$\sqrt{\alpha^+} := \|d\|_{U^+}^* = \langle d, y^+\rangle^{1/2} = \langle a_j a_j^* y^+, y^+\rangle^{1/2} = |\langle a_j, y^+\rangle| = \frac{1}{\varphi^*}.$$

The last equality follows from $\text{bdry}\,\mathcal{Q} \ni \varphi^* d = \varphi^* \langle a_j, y^+\rangle a_j$ because $\{a_j, -a_j\} \subset \text{bdry}\,\mathcal{Q}$ and hence it must be the case that $|\varphi^* \langle a_j, y^+\rangle| = 1$.

2. Assume $j = j^-$. First note that $\beta^2 = \langle a_{j^-}, y\rangle^2 \leq \alpha$ with equality if and only if $w$ is optimal.

   If $\gamma \leq 1$ then (3.39) implies $\kappa^* = \kappa_{min}$. If $\gamma > 1$, we get $\beta^2 \leq \alpha < \alpha\gamma$ and consequently $\kappa^* = \max\{\kappa_{min}, \kappa_1\}$. Moreover, it is easy to show that $\beta^2 \leq \alpha$ is equivalent to $\kappa_1 \leq 0$, which leads to the observation that $\kappa^* \leq 0$. If the current iterate is not optimal, then $\beta^2 < \alpha$ and thus $\kappa^* < 0$.

We have arrived at the following conclusion:

**Corollary 3.3.11.** *Under the assumptions of Theorem 3.3.10 the following hold.*

1. *If $j = j^+$ then*

$$\kappa^* = \begin{cases} \kappa_1 \geq 0 & \text{if } \gamma > 1 \text{ and } \alpha\gamma > \beta^2, \\ \infty & \text{if } \gamma > 1 \text{ and } \alpha\gamma = \beta^2. \end{cases} \tag{3.44}$$

*Moreover, it is always the case that $\alpha \leq \beta^2$, with equality if and only if $w$ is optimal. This happens, in particular, if $\gamma = 1$. The new iterate after the $\kappa^* = \infty$ step is taken is optimal.*

2. *If $j = j^-$ then*

$$\kappa^* = \begin{cases} \kappa_{min} & \text{if } \gamma \leq 1, \\ \max\{\kappa_{min}, \kappa_1\} \leq 0 & \text{if } \gamma > 1. \end{cases} \tag{3.45}$$

*Moreover, it is always the case that $\beta^2 \leq \alpha$, with equality if and only if $w$ is optimal.*

### 3.3.4 An algorithm with "increase" steps only

In this subsection we design and analyze an algorithm which at every iteration uses the choice $j = j^+ = \arg\max_i |\langle a_i, y\rangle|$, where $y$ is some vector satisfying $Uy = d$, and updates $U$ to $U(\kappa) = (U + \kappa a_j a_j^*)/(1 + \kappa)$, using the optimal step size $\kappa^*$ described by Corollary 3.3.11. Since this particular choice of $j$ always leads to nonnegative value of the optimal step size parameter, strictly positive if $w$ is not optimal, we see from the definition of $w(\kappa)$ (3.28) that the weight $w_j$ will increase while all other weights decrease uniformly to account for this. This explains the choice of the terminology "increase" step.

Since the initial iterate $w_0$ used in Algorithm 11 has all components positive (all are equal to $\frac{1}{m}$), all weights stay positive throughout the algorithm. In other words, the method proceeds through the interior of the feasible region. One important consequence of this is that the iterate matrices $U$ never lose rank and hence stay positive definite throughout the algorithm. This implies that a system of the form $Uy = d$ will always have a unique solution, which is the first step towards an

implementable code. Of course, numerical instabilities might occur in situations when certain weights get close to zero and, as a result, $U$ becomes nearly rank-deficient (see (3.2.9)). In this work we do not present any strategies for dealing with this linear algebra issue and instead focus on the optimization-theoretic results. Let us remark that it is unlikely that there will be problems with solving $Uy = d$ for a general position of the vector $d$.

The analysis of Algorithm 11 is based on a result which says that a certain approximation of the optimal step size gives a sufficient decrease in the value of $\psi^2$ (see Lemma 3.3.12 below). Let us first describe a motivational heuristic, leading us to the discovery of a suitable approximately optimal step size.

**A step-size heuristic**

Let

$$\hat{\delta} := \frac{|\beta|}{\sqrt{\alpha}} - 1, \tag{3.46}$$

which can be also written as $\beta^2 = \alpha(1+\hat{\delta})^2$, and assume $\hat{\delta} > 0$. By Theorem 3.2.25, the current iterate $w$ is $\hat{\delta}$-optimal for $(P3)$. To see this, we just need to translate our simplified notation (using $\alpha, \beta$ and $\gamma$) to the symbols used in that theorem: $\varphi(y) = |\beta|$ and $\psi(w) = \sqrt{\alpha}$. We now see that the definition of $\hat{\delta}$ implies $\varphi(y) = (1+\hat{\delta})\psi(w)$, which is precisely the universal $\hat{\delta}$-approximate optimality condition (3.27), implying $\psi(w) \leq (1+\hat{\delta})\psi^*$.

Assume, just for the sake of the motivational heuristic to follow, that $\alpha\gamma > \beta^2$. This excludes the "next-iterate is optimal" case from the description of the optimal step size of Corollary 3.3.11 and implies $\kappa^* = \kappa_1$. We claim that in the situation when $\gamma$ is large, $\kappa^*$ is reasonably well approximated by $\hat{\delta}/\gamma$. Indeed, the ratio

$\kappa_1/(\hat{\delta}/\gamma)$ converges to 1 (from above) as $\gamma$ approaches infinity:

$$\frac{\kappa_1}{\frac{\hat{\delta}}{\gamma}} = \frac{-\frac{1}{\gamma} + \frac{|\beta|\sqrt{\gamma-1}}{\gamma\sqrt{\alpha\gamma-\beta^2}}}{\frac{\frac{|\beta|}{\sqrt{\alpha}}-1}{\gamma}} = \frac{|\beta|\sqrt{\frac{\gamma-1}{\alpha\gamma-\beta^2}}-1}{|\beta|\sqrt{\frac{1}{\alpha}}-1} \downarrow 1 \qquad \text{as} \qquad \gamma \to \infty.$$

Note that the convergence is "from above" because $(\gamma-1)/(\alpha\gamma-\beta^2) > 1/\alpha$, which follows from $\beta^2 > \alpha$.

**Sufficient decrease**

Being motivated by the optimal step-size approximation discussed above, we now show that if the condition for $\delta$-approximate optimality of Theorem 3.2.25 is not met, then by taking the step $\delta/\gamma$, we can reduce the (square of the) objective value by at least $\alpha\delta^2/\gamma$. This will play a central role in the analysis of our algorithm. Note that by taking a finite step ($\kappa \neq \infty$), the function $\psi^2$ decreases by

$$\varepsilon(\kappa) := \psi^2(0) - \psi^2(\kappa) = \frac{\beta^2\kappa(1+\kappa)}{1+\gamma\kappa} - \alpha\kappa. \tag{3.47}$$

If $\kappa = \infty$ is used, this formula should be understood in the limit sense – see (3.34).

**Lemma 3.3.12** (Sufficient decrease)**.** *If $|\beta| \geq (1+\delta)\sqrt{\alpha}$ for some $\delta \geq 0$ and we let $\kappa := \frac{\delta}{\gamma}$, then*

$$\varepsilon(\kappa^*) \geq \varepsilon(\kappa) \geq \alpha\frac{\delta^2}{\gamma}.$$

*Proof.*

$$\begin{aligned}\varepsilon(\kappa^*) \geq \varepsilon(\kappa) = \frac{\beta^2\kappa(1+\kappa)}{1+\gamma\kappa} - \alpha\kappa &\geq \frac{(1+\delta)^2\alpha\frac{\delta}{\gamma}(1+\frac{\delta}{\gamma}) - (1+\delta)\alpha\frac{\delta}{\gamma}}{1+\delta} \\ &= \frac{\alpha\delta}{\gamma}[(1+\delta)(1+\tfrac{\delta}{\gamma}) - 1] \\ &\geq \frac{\alpha\delta^2}{\gamma}.\end{aligned}$$

The last inequality follows from the estimate $1 + \frac{\delta}{\gamma} \geq 1$. ∎

**Remark 3.3.13.** *Observe that the condition of the above lemma is satisfied with equality for $\delta = \hat{\delta}$ defined in (3.46).*

**Even better decrease**

It turns out that if we take into account also some other choices of $j$, we can possibly achieve an even bigger decrease in $\psi^2$ than that guaranteed by the above lemma. Let $\beta^i := \langle a_i, y\rangle$ and $\gamma^i := \langle a_i, U^{-1}a_i\rangle$ (for all $i$), so that $\beta^j = \beta$ and $\gamma^j = \gamma$. Also let $\delta^i := (|\beta^i|/\sqrt{\alpha}) - 1$ and $\mathcal{I}$ be the set of those indices $i$ for which $\delta^i \geq 0$. Note that $j = j^+ \in \mathcal{I}$. Observe that the argument of the above lemma can be repeated to show that

$$\varepsilon(\kappa^*) \geq \varepsilon(\delta^i/\gamma^i) \geq \alpha\frac{(\delta^i)^2}{\gamma^i} = \frac{(|\beta^i| - \sqrt{\alpha})^2}{\gamma^i}, \qquad \forall\, i \in \mathcal{I}. \tag{3.48}$$

Note that this is the lower bound on the change in $\psi^2$ when using $a_i$ instead of $a_j$ and the corresponding approximately optimal step size. While the specific choice $i = j$ guarantees sufficient decrease, we might be able to do better by *optimizing* over the set $\mathcal{I}$ rather than by picking the *feasible* solution $i = j \in \mathcal{I}$. This leads us to defining

$$i^* := \arg\max_{i\in\mathcal{I}} \frac{(|\beta^i| - \sqrt{\alpha})^2}{\gamma^i}.$$

Certainly, the *decrease guaranteed* by $i^*$ is at least as good as the decrease guaranteed by $j$. This does not mean, however, that the *actual decrease*, by taking the optimal step, will be bigger. This has to be taken into account when implementing this strategy in an algorithm. If we decide to use this improvement, we have to deal with the issue of the actual computation of the values $\gamma^i$, which are needed to find $i^*$. By doing the computation from scratch at every iteration, we will need $O(mn^2)$ arithmetic operations: $O(n^2)$ for solving each of the at most $m$ equations

$Ux = a_i$ $(i \in \mathcal{I})$, assuming we maintain the Cholesky factorization of $U$ from iteration to iteration. Alternatively, we can solve for $U_0^{-1}a_i$ for all $i$ at the beginning of the algorithm, which takes $O(mn^2)$ operations if we assume the availability of the Cholesky factors of $U_0$, and subsequently update the solutions as we modify the (Cholesky factorization of the) matrix. The work per iteration will drop to $O(mn)$, which is of the same order as the work needed to calculate $j$.

**Algorithm 11 (Inc)** Solving $(P3)$ using increase steps only.

**Input:** $a_1, \ldots, a_m \in \mathbf{E}^*,\ d \in \mathbf{E}^*,\ \delta > 0$;

**Initialize:** $k = 0,\ w_0 = (1/m)e_m,\ U_0 = \frac{1}{m}\sum_i a_i a_i^*,\ y_0 = U_0^{-1}d$;

**Iterate:**

  $\alpha_k = \langle d, y_k \rangle,\ j = \arg\max_i |\langle a_i, y_k \rangle|,\ g_k = a_j$;

  $\beta_k = \langle g_k, y_k \rangle,\ \gamma_k = \langle g_k, U_k^{-1} g_k \rangle$;

  $\delta_k = \frac{|\beta_k|}{\sqrt{\alpha_k}} - 1$;

  **if** $\delta_k \leq \delta$

    **terminate**;

  **else**

    **if** $\alpha_k \gamma_k = \beta_k^2$

      $\kappa^* = \infty,\ U_{k+1} = g_j g_j^*,\ w_{k+1} = \frac{w_k + \kappa^* e_j}{1+\kappa^*} = (\underbrace{0, \ldots, 0}_{1 \ldots j-1}, \underbrace{1}_{j}, \underbrace{0, \ldots, 0}_{j+1 \ldots m})$;

      $k \leftarrow k+1,$ **terminate;** $\quad (w_k$ is optimal)

    **else**

      $\kappa^* = -\frac{1}{\gamma_k} + \frac{|\beta_k|\sqrt{\gamma_k - 1}}{\gamma_k \sqrt{\alpha_k \gamma_k - \beta_k^2}},\ U_{k+1} = \frac{U_k + \kappa^* g_k g_k^*}{1+\kappa^*},\ w_{k+1} = \frac{w_k + \kappa^* e_j}{1+\kappa^*}$;

      $y_{k+1} = (1+\kappa^*)\left(y_k - \frac{\beta_k \kappa^* U_k^{-1} g_k}{1 + \gamma_k \kappa^*}\right)$;

      $k \leftarrow k+1$;

  **end if**

**Output:** $w_k$ satisfying $\|d\|_{U_k}^* = \sqrt{\alpha_k} \leq (1+\delta)\psi^*$ and $U_k = U(w_k)$

### A crucial assumption

**Assumption 3.3.14.** *The values $\gamma_k$ generated by Algorithm 11 are bounded above by some constant $\Gamma$.*

As we shall see, the parameter $\Gamma$ appears in the complexity bounds. It would therefore be good to be able to estimate its size. We will deal with this issue in Subsection 3.3.6.

### Quick and dirty analysis

Let us first offer a rough analysis of Algorithm 11, leading to a performance guarantee of $O(\Gamma\delta^{-2}\ln m)$ iterations of a first-order method, followed by a more refined analysis with the guarantee $O(\Gamma(\ln\Gamma+\ln\ln m+\delta^{-1}))$. For the quick result observe that because

$$\varphi^* d \in \mathcal{Q} \subseteq \sqrt{m}\mathcal{B}(U_0), \tag{3.49}$$

we have $\sqrt{\alpha_0} = \|d\|_{U_0}^* \leq \sqrt{m}/\varphi^*$. Now assume that Algorithm 11 produces $K+1$ iterates with $K+1 \geq \lceil \Gamma\delta^{-2}\ln m\rceil$. The termination criterion of line 7 then implies that $\delta_k > \delta$ for $k = 0, 1, \ldots, K$. Since $|\beta_k| = (1+\delta_k)\sqrt{\alpha_k}$ for all $k \leq K$, Lemma 3.3.12 and Assumption 3.3.14 imply

$$\alpha_k - \alpha_{k+1} \geq \alpha_k \frac{\delta_k^2}{\gamma_k} > \alpha_k \frac{\delta^2}{\Gamma}.$$

Repeated use of this inequality gives $\alpha_{K+1} < \alpha_0(1-\delta^2/\Gamma)^{K+1}$ and hence

$$\begin{aligned}\psi^2(w_{K+1}) = \alpha_{K+1} &< \alpha_0(1-\delta^2/\Gamma)^{K+1} \\ &\leq \alpha_0 e^{-(K+1)\delta^2/\Gamma} \leq \alpha_0 e^{-\ln m} \leq \tfrac{m}{(\varphi^*)^2}\tfrac{1}{m} = (\psi^*)^2,\end{aligned}$$

which contradicts the fact that $\psi^*$ is the optimal value of problem (P3).

**Refined analysis**

The following is the central result of this chapter:

**Theorem 3.3.15.** *Under Assumption 3.3.14, Algorithm 11 produces a $\delta$-approximate solution of (P3) (and hence by Theorem 3.2.25 of (P1), (D1), (P2) and (D2)) in at most*

$$2\Gamma\left(\ln\Gamma + \ln\ln m + \frac{8}{\delta}\right)$$

*iterations.*

*Proof.* Let $L_k := \ln\sqrt{\alpha_k}$ and $L^* = \ln\psi^*$ and notice that $\sqrt{\alpha_k} \leq (1+\delta_k)\psi^*$. By taking logarithms,

$$\varepsilon_k' := L_k - L^* \leq \ln(1+\delta_k). \tag{3.50}$$

Also, $\beta_0^2 = \max_i \langle a_i, y_0\rangle^2 \leq \sum_i \langle a_i, y_0\rangle^2 = m\langle U_0 y_0, y_0\rangle = m\alpha_0 = m\frac{\beta_0^2}{(1+\delta_0)^2}$, whence

$$\delta_0 \leq \sqrt{m} - 1 \qquad \text{and} \qquad \varepsilon_0' \leq \ln(1+\delta_0) \leq \tfrac{1}{2}\ln m. \tag{3.51}$$

By Lemma 3.3.12, $\varepsilon(\kappa^*) = \alpha_k - \alpha_{k+1} \geq \alpha_k\delta_k^2/\gamma_k \geq \alpha_k\delta_k^2/\Gamma$ and therefore

$$\alpha_{k+1} \leq \alpha_k(1 - \delta_k^2/\Gamma). \tag{3.52}$$

By taking logarithms in (3.52) and using (3.50),

$$L_k - L_{k+1} \geq -\tfrac{1}{2}\ln(1-\delta_k^2/\Gamma) \geq \tfrac{1}{2}\delta_k^2/\Gamma \geq \tfrac{1}{2\Gamma}\ln(1+\delta_k^2) \geq \tfrac{1}{2\Gamma}\ln(1+\delta_k), \tag{3.53}$$

with the last inequality true whenever $\delta_k \geq 1$. Combining (3.50) and (3.53) yields

$$\varepsilon_{k+1}' \leq \varepsilon_k'(1 - \tfrac{1}{2\Gamma}),$$

for all $k$ with $\delta_k \geq 1$. We will now bound the number of iterations for which $\delta_k \geq 1$. The last inequality together with (3.51) gives

$$\varepsilon_k' \leq \varepsilon_0'(1-\tfrac{1}{2\Gamma})^k \leq \tfrac{1}{2}\ln m\exp(-\tfrac{k}{2\Gamma}). \tag{3.54}$$

Due to (3.54) and $\varepsilon'_k \geq \varepsilon'_k - \varepsilon'_{k+1} = L_k - L_{k+1} \geq \frac{1}{2}\delta_k^2/\Gamma \geq \frac{1}{2}\Gamma^{-1}$, the largest $k$ for which $\delta_k \geq 1$ must satisfy $\Gamma^{-1} \leq \ln m \exp(-\frac{k}{2\Gamma})$, leading to the bound

$$k \leq 2\Gamma(\ln \Gamma + \ln \ln m). \tag{3.55}$$

So one can obtain a solution within the factor of 2 of the optimum in $O(\Gamma(\ln \Gamma + \ln \ln m))$ iterations of Algorithm 11.

Following the "halving" argument of Khachiyan [15], we can bound the number of additional iterations needed to obtain the desired $\delta$-approximate solution. Suppose $\delta_k \leq 1$, and let $h(\delta_k)$ be the smallest integer $h$ such that $\delta_{k+h} \leq \delta_k/2$. Whenever $\delta_{k+h} \geq \delta_k/2$, we also have

$$\varepsilon'_{k+h} - \varepsilon'_{k+h+1} \geq \tfrac{1}{2}\delta_{k+h}^2/\Gamma \geq \tfrac{1}{8}\delta_k^2/\Gamma,$$

which says that the gap in (3.50) must at every such iteration decrease by at least $\frac{1}{8}\delta_k^2/\Gamma$. However, the original gap is of size at most $\varepsilon'_k \leq \ln(1+\delta_k) \leq \delta_k$, and hence the number of iterations needed for halving $\delta_k$ is bounded above by

$$h(\delta_k) \leq \frac{\delta_k}{\frac{1}{8}\delta_k^2/\Gamma} = \frac{8\Gamma}{\delta_k}.$$

In order to get below $\delta$, we need to "halve" $l$-times where $l$ is obtained from $\delta_k/2^l \leq \delta$, that is $l = \lceil \log_2 \delta_k/\delta \rceil$, where $k$ is the first iteration for which $\delta_k \leq 1$. The total number of additional iterations required to achieve the desired $\delta$-approximate solution is at most

$$\sum_{i=0}^{l-1} h(\delta_k/2^i) \leq 8\Gamma \sum_{i=0}^{l-1} \frac{1}{\delta_k/2^i} = \frac{8\Gamma}{\delta_k} 2^{\lceil \log_2 \delta_k/\delta \rceil} \leq \frac{16\Gamma}{\delta}.$$

□

### 3.3.5 An algorithm with both "increase" and "decrease" steps

In the previous subsection we have analyzed an algorithm which at every iteration works with $j = j^+$ (an "increase" step). A consequence of this choice is that the optimal step-size parameter $\kappa^*$ is always nonnegative, implying that $w_j$ is being increased while all other weights are decreased uniformly (and hence at a slower rate than the rate of increase of $w_j$) in due compensation. Starting from $w_0 = (\frac{1}{m}, \ldots, \frac{1}{m})$, Algorithm 11 keeps all weights positive until termination. In an optimal solution $w$, however, the weights can be positive only for points $a_i$ lying on a face (say $\mathcal{F}$) of $\mathcal{Q}$ containing the point $\varphi^* d$ — the intersection of $\mathcal{Q}$ and the half-line emanating from the origin in the direction $d$ (see Figure 3.7). Note that in the case when $m \gg n$, it is to be expected that many more points will have zero weights rather than positive weights, at optimality. It therefore seems intuitive that if the incorporation of "decrease" and/or "drop" steps could speed up the algorithm considerably.

In this subsection we propose and analyze an algorithm in which we allow also for "decrease" and "drop" iterations — steps which decrease $w_j$, respectively drop it to zero ($\kappa = -w_j$). The idea is as follows. At every iteration we consider both $j = j^+$ and $j = j^-$. We make the latter choice if the predicted decrease is better (this corresponds to $\delta^- \geq \delta^+$ in Algorithm 12), except when this leads to a drop step reducing the rank of $U$ (this happens when $-\frac{1}{\gamma} = -w_j = \kappa$). Otherwise we choose $j = j^+$.

There are several reasonable alternative rules for deciding among $j^+$ and $j^-$. For example, we could base our decision on comparing the *actual decrease* as opposed to the *decrease predicted* by $\delta^+$ and $\delta^-$. We could also forbid taking drop

steps altogether, allowing only for decrease steps, etc.

Let us start with a twin result to Lemma 3.3.12 which essentially says that if we choose $j = j^-$ and it happens that $\kappa^*$ is not a drop step, then by taking this step we are guaranteed sufficient decrease in the (square of the) objective function:

**Lemma 3.3.16.** *Assume $j = j^-$.*

*(i) If $|\beta| \leq (1-\delta)\sqrt{\alpha}$ for some $0 \leq \delta < 1$ and $\kappa := -\frac{\delta}{\gamma} \geq -w_j$, then*

$$\varepsilon(\kappa^*) \geq \varepsilon(\kappa) \geq \alpha\frac{\delta^2}{\gamma}.$$

*(ii) If $\kappa^* = \kappa_1$ and $\delta := 1 - \frac{|\beta|}{\sqrt{\alpha}}$, then $\kappa := -\frac{\delta}{\gamma} \geq -w_j$.*

*Proof.* For part $(i)$ notice that the assumption $\kappa \geq -w_j = \kappa_{min}$ ensures feasibility of the line-search parameter $\kappa$. Also observe that $1-\frac{\delta}{\gamma} \geq 1-w_j > 0$ by Assumption 2 in Subsection 3.3.2. We now proceed as in Lemma 3.3.12:

$$\begin{aligned}
\varepsilon(\kappa^*) \geq \varepsilon(\kappa) &= \frac{\beta^2\kappa(1+\kappa)}{1+\gamma\kappa} - \alpha\kappa \\
&\geq \frac{-(1-\delta)^2\alpha\frac{\delta}{\gamma}(1-\frac{\delta}{\gamma}) + (1-\delta)\alpha\frac{\delta}{\gamma}}{1-\delta} \\
&= \frac{\alpha\delta}{\gamma}[1-(1-\delta)(1-\tfrac{\delta}{\gamma})] \\
&\geq \frac{\alpha\delta^2}{\gamma}.
\end{aligned}$$

The last inequality follows from the estimate $0 < 1-\frac{\delta}{\gamma} \leq 1$. Let us now prove $(ii)$. Because $\kappa^* = \kappa_1$ is feasible for the line-search problem, we must have $\kappa_1 \geq -w_j$. However, using the inequality $\beta^2 \leq \alpha$ it can be argued by simple algebra that $-\frac{\delta}{\gamma} \geq \kappa_1$ (see (3.40) for the definition of $\kappa_1$). □

**Theorem 3.3.17.** *Under Assumption 3.3.14, Algorithm 12 produces a $\delta$-approximate solution of (P3) (and hence by Theorem 3.2.25 of (P1), (D1),*

**Algorithm 12 (IncDec)** Solving $(P3)$ using both increase and decrease steps.

**Input:** $a_1, \ldots, a_m \in \mathbf{E}^*,\ d \in \mathbf{E}^*,\ \delta > 0$;

**Initialize:** $k = 0,\ w_0 = (1/m)e_m,\ U_0 = \frac{1}{m}\sum_i a_i a_i^*,\ y_0 = U_0^{-1} d$;

**Iterate:**

$\alpha_k = \langle d, y_k \rangle$;

$j^- = \arg\min_i \{|\langle a_i, y_k\rangle| \ :\ w_k^{(i)} > 0\},\ \beta^- = \langle a_{j^-}, y_k\rangle,\ \delta^- = 1 - \frac{|\beta^-|}{\sqrt{\alpha_k}}$;

$j^+ = \arg\max_i |\langle a_i, y_k\rangle|,\ \beta^+ = \langle a_{j^+}, y_k\rangle,\ \delta^+ = \frac{|\beta^+|}{\sqrt{\alpha_k}} - 1$;

**if** $\delta^+ \le \delta$ **then terminate**; **end if**

**if** $\delta^+ < \delta^-$

$j = j^-,\ g_k = a_j,\ \beta_k = \beta^-,\ \gamma_k = \langle g_k, U_k^{-1} g_k\rangle,\ \delta_k = \delta^-$;

**if** $\gamma_k > 1$ **then** $\kappa = \max\left\{-\frac{1}{\gamma_k} + \frac{|\beta_k|\sqrt{\gamma_k - 1}}{\gamma_k\sqrt{\alpha_k\gamma_k - \beta_k^2}}, -w_k^{(j)}\right\}$;

**else** $\kappa = -w_k^{(j)}$; **end if**

**if** $\kappa = -w_k^{(j)} = -\frac{1}{\gamma_k}$ **then jump to** 14, **end if**

**else**

$j = j^+,\ g_k = a_j,\ \beta_k = \beta^+,\ \gamma_k = \langle g_k, U_k^{-1} g_k\rangle,\ \delta_k = \delta^+$;

**if** $\alpha_k\gamma_k > \beta_k^2$ **then** $\kappa = -\frac{1}{\gamma_k} + \frac{|\beta_k|\sqrt{\gamma_k - 1}}{\gamma_k\sqrt{\alpha_k\gamma_k - \beta_k^2}}$;

**else** $\kappa = \infty$; (the next iterate is optimal) **end if**

**end if**

$w_{k+1} = \frac{w_k + \kappa e_j}{1+\kappa},\ U_{k+1} = \frac{U_k + \kappa g_k g_k^*}{1+\kappa},\ y_{k+1} = (1+\kappa)\left(y_k - \frac{\beta_k \kappa U_k^{-1} g_k}{1 + \gamma_k \kappa}\right)$;

$k \leftarrow k + 1$;

**Output:** $w_k$ satisfying $\|d\|_{U_k}^* = \sqrt{\alpha_k} \le (1+\delta)\psi^*$ and $U_k = U(w_k)$

*(P2) and (D2)) in at most*

$$m + 4\Gamma\left(\ln\Gamma + \ln\ln m + \frac{8}{\delta}\right)$$

*iterations.*

*Proof.* Due to Lemma 3.3.16, the argument is identical to the proof of Theorem 3.3.15. The difference is that we need to bound the number of drop iterations because these do not guarantee any positive decrease (but do not increase the objective either). Note that either the current point $a_j$ is dropped for the first time (there are a maximum of $m$ such occurrences), or it has been dropped before, in which case we can pair it up with the previous iteration that increased the weight $w_j$ from zero to a positive value. This algorithm therefore needs at most $m$ plus twice the number of iterations guaranteed by Theorem 3.3.15. □

**Remark 3.3.18.** *The* $\ln\ln m$ *factor in the complexity estimates of Algorithms 11 and 12 can be replaced by* $\ln\ln n$ *if we pre-compute a rounding of* $\mathcal{Q}$ *with* $\frac{1}{\alpha} = O(\sqrt{n})$ *and use the corresponding matrix as* $U_0$*. This can be done in* $O(n^2 m \log m)$ *arithmetic operations (see [22]).*

### 3.3.6 Bounding the unknown constant

The performance guarantees of Algorithms 11 and 12 depend on the assumption that the squared norms of the points $a_j$ encountered throughout the iterations are bounded from above by some constant $\Gamma$. It is therefore highly desirable to invest some time into exploring our options of theoretical and/or practical justification of this assumption.

How large can $\Gamma$ be? Notice that we know from Corollary 3.3.5 that for any $j$

with positive weight $w_j$, the value

$$\gamma_j := (\|a_j\|^*_{U(w)})^2$$

can be bounded from above by a function of $w_j$:

$$\gamma_j \leq \tfrac{1}{w_j}. \tag{3.56}$$

If $w_j = 0$, as is the case when we perform an "add" step in Algorithm 12, we do not have an upper bound on $\gamma_j$. If we maintain all weights positive, as in Algorithm 11, then $\Gamma$ can certainly be bounded by the reciprocal of the smallest weight $w_j$ encountered throughout the algorithm. This leads to the idea of modifying our methods so as to keep all weights above a certain positive constant.

**Bounding the weights away from zero: theoretical implications**

Motivated by the above discussion, let us explicitly require that all weights be bounded away from zero by $\frac{\varepsilon}{m}$, with $\varepsilon \in [0, 1]$ being a small constant *independent of the dimensions* of the problem. Note that setting $\varepsilon = 1$ implies that all weights are equal to $\frac{1}{m}$.

It seems to be intuitively sound to expect that if we restrict the set of feasible points of problem $(P3)$ by requiring $w_i \geq \frac{\varepsilon}{m}$ for all $i$, the optimal value of the modified problem, which we will call $(P3_\varepsilon)$, should be close to the optimal value of $(P3)$. Also, as $\varepsilon$ gets smaller, the optimal value of $(P3_\varepsilon)$ should approach that of $(P3)$. We will formalize these ideas in the remainder of this subsection. Let

$$\Delta^\varepsilon_m := \{w \in \Delta_m \ : \ w_i \geq \tfrac{\varepsilon}{m}, \ i = 1, 2, \ldots, m\}$$

and consider the following problem

$$(P3_\varepsilon) \qquad \boxed{\psi^*_\varepsilon := \min_w \{\psi(w) \ : \ w \in \Delta^\varepsilon_m\}.}$$

We claim that the value $\psi_\varepsilon^*$ is close to $\psi^*$ for small $\varepsilon$:

**Theorem 3.3.19.** *For the optimal values $\psi^*$ and $\psi_\varepsilon^*$ of (P3) and (P3$_\varepsilon$), respectively, we have*

$$\psi_\varepsilon^* \leq \frac{1}{(1-\frac{m-1}{m}\varepsilon)^{1/2}}\psi^*. \tag{3.57}$$

To prove this we will need an auxiliary result.

**Lemma 3.3.20.** *For any $x \in \mathbf{E}$,*

$$\max_{w\in\Delta_m^\varepsilon} \|x\|_{U(w)} \geq (1-\tfrac{m-1}{m}\varepsilon)^{1/2}\varphi(x).$$

*Proof.* Assume $\varphi(x) = |\langle a_j, x\rangle|$ and let $w'$ be a vector of weights with $w'_j = 1-\frac{m-1}{m}\varepsilon$ and $w'_i = \frac{1}{m}\varepsilon$ for all other $i$. Then

$$\max_{w\in\Delta_m^\varepsilon} \|x\|_{U(w)} \geq \|x\|_{U(w')} = \Big(\sum w'_i\langle a_i, x\rangle^2\Big)^{1/2} \geq (w'_j)^{1/2}|\langle a_j, x\rangle|.$$

□

*Proof.* (theorem)

$$\begin{aligned}
\frac{1}{\psi_\varepsilon^*} = \left[\min_{w\in\Delta_m^\varepsilon} \|d\|_{U(w)}^*\right]^{-1} &= \max_{w\in\Delta_m^\varepsilon} 1/\|d\|_{U(w)}^* \\
&= \max_{w\in\Delta_m^\varepsilon} \min_{\langle d,x\rangle=1} \|x\|_{U(w)} \\
&= \min_{\langle d,x\rangle=1} \max_{w\in\Delta_m^\varepsilon} \|x\|_{U(w)} \\
&\geq \min_{\langle d,x\rangle=1} (1-\tfrac{m-1}{m}\varepsilon)^{1/2}\varphi(x) \\
&= (1-\tfrac{m-1}{m}\varepsilon)^{1/2}\varphi^* = (1-\tfrac{m-1}{m}\varepsilon)^{1/2}\frac{1}{\psi^*}.
\end{aligned}$$

The exchange of the maximum and minimum can be justified by using Hartung's minimax theorem [12]. □

**Remark 3.3.21.** *For $\varepsilon = 1$, inequality (3.57) states that $\psi(w_0) \leq \sqrt{m}\psi^*$, where $w_0$ is the vector of all weights equal to $\frac{1}{m}$. This we have already seen before as a consequence of the rounding property (3.49) of $U_0 = U(w_0)$.*

**Corollary 3.3.22.** *If $\varepsilon \leq \frac{1}{2\tau}(\sqrt{\tau(\tau+4)} + \tau - 2)$ for some positive parameter $\tau$ (necessarily, $\tau \geq \frac{1}{2}$), then $\psi_\varepsilon^* \leq (1+\tau\varepsilon)\psi^*$. In particular, if $\varepsilon \leq \frac{1}{2}(\sqrt{5}-1)$, then $\psi_\varepsilon^* \leq (1+\varepsilon)\psi^*$.*

*Proof.* The condition on $\varepsilon$ is equivalent to the last inequality in $(1 - \frac{m-1}{m}\varepsilon)^{-1/2} \leq (1-\varepsilon)^{-1/2} \leq (1+\tau\varepsilon)$. □

**Bounding the weights away from zero: algorithmic implications**

It is not trivial to see how one would go about modifying our algorithms to efficiently solve $(P3_\varepsilon)$. The requirement of keeping the weights above some positive threshold value $\frac{\varepsilon}{m}$ does not seem to be cheap to maintain. Let us briefly explain why.

One possible approach to solving $(P3_\varepsilon)$ using our methodology would involve dividing the operator $U$ into two parts, keeping one fixed, ensuring that the weights are kept above $\frac{\varepsilon}{m}$. The other is a variable part, consisting of the remaining portion of the total weight. That is, we write

$$U = \sum_{i=1}^{m} \tfrac{\varepsilon}{m} a_i a_i^* + \sum_{i=1}^{m} w_i' a_i a_i^* = U_\varepsilon + U(w'),$$

where $\sum_i w_i' = 1-\varepsilon,\ w_i' \geq 0$; that is, $w' \in (1-\varepsilon)\Delta_m$. One would now update only the variable part, similarly as in the previous analysis:

$$U(\kappa) = U_\varepsilon + \frac{U(w') + \kappa a_j a_j^*}{1 - \varepsilon + \kappa}. \tag{3.58}$$

Notice that we no longer have $1+\kappa$ in the denominator, and this would need to be accounted for by reworking the relevant analysis. The main problem with this

approach, however, is that (3.58) no longer constitutes a simple enough update of the operator $U$. It is certainly not a rank-one-and-scaling update as before. This means that it could be hard to be able to use the information from the previous iteration (for example, the Cholesky factor of $U$ and the solution $y$ of $Uy = d$) to solve the new system $U(\kappa)y = d$. If we need to solve this from scratch, it requires $O(n^3)$ arithmetic operations (assuming $U(\kappa)$ is assembled from $U_\varepsilon$ and $U(w')$ via (3.58), which takes only $O(n^2)$ arithmetic operations), which is worse than the previous $O(n^2)$ work. However, the per-iteration arithmetical complexity of Algorithms 11 and 12 is $O(mn)$, which will dominate the work above in the case when $m \geq n^2$. The critical saving would then come from the fact that we do not have to form the new matrix from scratch, which would otherwise require $O(mn^2)$ arithmetic operations.

While in this thesis we do not show any details of a direct algorithm of this type for solving $(P3_\varepsilon)$, we believe that the ideas we have just described could be turned into a provably working algorithm, albeit one with a considerably higher computational effort per iteration.

### The average of the gammas

As a possible alternative to the conservative strategy of keeping *all* weights above a certain positive threshold value throughout the algorithm, let us briefly discuss if it is possible to instead select a *particular* $j$ so that $\gamma_j$ is of a reasonable size. Let us start with the following simple observation:

**Lemma 3.3.23.**

$$\sum_{w_i>0} w_i\gamma_i = \operatorname{rank} U(w).$$

*Proof.* Assume first $U := U(w)$ is invertible. Then

$$\begin{aligned}\sum_{w_i>0} w_i\gamma_i = \sum_i w_i\langle a_i, U(w)^{-1}a_i\rangle &= \sum_i w_i \operatorname{trace}[\langle a_i, U(w)^{-1}a_i\rangle] \\ &= \sum_i w_i \operatorname{trace}[a_i a_i^* U(w)^{-1}] \\ &= \operatorname{trace}\left[\left(\sum_i w_i a_i a_i^*\right) U(w)^{-1}\right] \\ &= \operatorname{trace} I = \dim \mathbf{E}^* = n,\end{aligned}$$

where $I\colon \mathbf{E}^* \to \mathbf{E}^*$ is the identity operator. The general case is handled by transforming it to the nonsingular case above. Indeed, let $\mathcal{X}$ be a subspace of $\mathbf{E}$ for which $U(w)$, viewed as a map from $\mathcal{X}$ onto $\operatorname{range} U(w)$, is invertible and notice that $\dim \operatorname{range} U(w) = \operatorname{rank} U(w)$. □

Let us illustrate the lemma with an example:

**Example 3.3.24.** Assume $U := U(w)$ is of rank 1 and let $w_1 = 1$; all other weights being zero. Since $U = a_1 a_1^*$, the solution set of the system $Ux = a_1$ consists precisely of the vectors $x$ satisfying $\langle a_1, x\rangle = 1$. However, $\sum_{w_i>0} w_i\gamma_i = \gamma_1 = \langle a_1, x\rangle = 1 = \operatorname{rank} U(w)$.

The above lemma implies that there is always some index $i$ such that $\gamma_i = O(n)$. However, we already have a procedure for picking $j$, and it does not take $\gamma_j$ into consideration. It would be interesting to see if it is possible to devise a procedure that would guarantee *both* a sufficient decrease in the objective function *and* a reasonable bound on $\gamma_j$. Let us remark that the "even better decrease" strategy for choosing $j$ given in (3.48) *is* biased towards choosing one with small $\gamma_j$.

Note that, as a corollary of the above lemma, we get the following, albeit

somewhat weaker, bound on $\gamma_j$:

$$\gamma_j \leq \frac{\operatorname{rank} U(w)}{w_j}. \tag{3.59}$$

**An alternative proof of the bound on $\gamma_j$**

Consider the concave quadratic $x \mapsto 2\langle a_j, x\rangle - \langle U(w)x, x\rangle$ and observe that its maximizers are precisely the points $x$ for which $U(w)x = a_j$. If $x_j$ is any such point then

$$\begin{aligned}
\gamma_j = \langle a_j, x_j\rangle &= \max_x \left\{2\langle a_j, x\rangle - \langle U(w)x, x\rangle\right\} \\
&= \max_x \left\{2\langle a_j, x\rangle - \sum_{i=1}^m w_i \langle a_i, x\rangle^2\right\} \\
&\leq \max_x \left\{2\langle a_j, x\rangle - w_j\langle a_j, x\rangle^2\right\} \\
&= \max_\tau \left\{2\tau - w_j\tau^2\right\} = \tfrac{1}{w_j},
\end{aligned}$$

yielding another proof of (3.56). The author wishes to thank Yurii Nesterov for this elegant proof.

## 3.4 Interpretation

We have seen in Theorem 3.2.24 (resp. Theorem 3.2.25) that by solving (resp. approximately solving) problem $(P3)$, we have simultaneously solved (resp. approximately solved) also problems $(P1)$, $(D1)$, $(D'1)$, $(P2)$ and $(D2)$. Moreover, the former theorem mentions how to explicitly construct feasible points for the above problems given a feasible point of $(P3)$. We can therefore in principle rewrite our algorithms, which were motivated by problem $(P3)$, in terms of iterates feasible for each of the above problems.

For example, if $\{w_k\}$ is a sequence of iterates produced by Algorithm 11 and $y_k \in \mathbf{E}$ satisfy $U(w_k)y_k = d$, then $\{v_k\}$ defined by $v_k^{(i)} := w_k^{(i)}\langle a_i, y_k\rangle$, $i = 1, 2, \ldots, m$, is a sequence of points feasible for $(D2)$. Is there a natural way to interpret these iterates in the context of problem $(D2)$?

### 3.4.1 $(P3)$: The Frank-Wolfe algorithm on the unit simplex

We will start with an alternative interpretation of our last two algorithms as applied to the main problem of this chapter:

$$\boxed{\psi^* := \min_w \{\|d\|^*_{U(w)} \ : \ w \in \Delta_m\}.} \qquad (P3)$$

The Frank-Wolfe algorithm [8] is a method for solving smooth convex minimization problems over a polytope given as a convex hull of points. At each iteration the objective function is replaced by its linear approximation at the current point. After this, one finds a vertex of the feasible region minimizing the linear approximation — this is a simple enumeration problem. The next iterate is then obtained by performing a line search on the line segment joining the current point and the vertex obtained using the enumeration procedure described above. The line search can be modified by allowing for Wolfe's "away steps" [35] — steps in the direction opposite to that towards the vertex maximizing the linear approximation.

It is straightforward to show, using the formula for the derivative of $\psi^2$ established in Proposition 3.2.19, that Algorithm 11 can be interpreted as a Frank-Wolfe method using the former version of line search (the decrease and drop steps of Algorithm 12 correspond to Wolfe's away steps). Indeed, the linear approximation

of $\psi^2$ at point $w$ for which $U(w)$ is invertible is

$$\begin{aligned}\psi^2(w) + D\psi^2(w)(w' - w) &= \psi^2(w) - \langle U(w' - w)y, y\rangle \\ &= \psi^2(w) + \langle U(w)y, y\rangle - \langle U(w')y, y\rangle,\end{aligned}$$

where $y = U(w)^{-1}d$. The linearized subproblem can therefore be written as

$$\min_{w' \in \Delta_m} -\langle U(w')y, y\rangle = \max_{w' \in \{e_1,\ldots,e_m\}} \sum_i w_i' \langle a_i, y\rangle^2.$$

Notice that $w = e_j$ where $j = \arg\max_i |\langle a_i, y\rangle|$ solves the above problem. The Frank-Wolfe line search now corresponds to the problem of minimizing $\psi^2(w(\kappa))$ for $\kappa \in [0, \infty]$ since $w(k) = (w + \kappa e_j)/(1 + \kappa)$ parameterizes the line segment joining $w$ and $e_j$. Notice that although in our line search we allow $-w_j \leq \kappa < 0$, the optimal steplength $\kappa^*$ is always nonnegative (Corollary 3.3.11).

For problems where the feasible region is a unit simplex and where the objective function enjoys certain regularity properties such as strong convexity (our function does not satisfy them), it is known that the Frank-Wolfe algorithm with away steps converges linearly [35], [11].

Methods analogous to Algorithm 11 (also interpretable as performing Frank-Wolfe iterations), for computing the minimum volume enclosing ellipsoid of a centrally symmetric body, were proposed by Khachiyan [15], Todd and Yildirim [33]. The method of Todd and Yildirim is a modification of Khachiyan's algorithm using away steps and has been later analyzed by Ahipaşaoğlu, Todd and Sun [1] who established its linear convergence. These algorithms, although perhaps without modern convergence analysis, were much earlier independently developed in the statistical community in the context of optimal design by Fedorov [7], Wynn [36], Atwood [3], Silvey [30], and others.

### 3.4.2 $(P2)$: An ellipsoid method for LP

Here we will consider problem $(P2)$:

$$\boxed{\frac{1}{\varphi^*} = \max_z \{\langle d, z\rangle \ : \ z \in \mathcal{Q}^\circ\}.} \qquad (P2)$$

Recall that for all $w \in \Delta_m$ the polar of the ellipsoid $\mathcal{B}(U(w))$ contains the polar of $\mathcal{Q}$, that is, $\mathcal{B}^0(U(w)) \supset \mathcal{Q}^0$ (Proposition 3.2.10). We also know that

$$\max\{\langle d, y\rangle \ : \ y \in \mathcal{B}^0(U(w))\} = \|d\|^*_{U(w)} \geq \psi^* = \tfrac{1}{\varphi^*}.$$

Let us fix some $w \in \Delta_m$ and let $U := U(w)$. Also let $y$ be such that $U(w)y = d$.

In one iteration of Algorithm 11 (or Algorithm 12) we update $U$ in a rank-one-and-scaling fashion to $U(\kappa)$ so as to minimize the value of $\psi(\kappa) = \|d\|^*_{U(\kappa)}$. The geometry of this update is rather revealing (see Figure 3.11). Loosely speaking, we choose the step-size parameter $\kappa$ so as to "push" the polar ellipsoid $\mathcal{B}^0(U(\kappa))$ by the supporting hyperplane $\mathcal{H}_d(\kappa) := \{z \ : \ \langle d, z\rangle = \|d\|^*_{U(\kappa)}\}$ as far as possible towards $z^*$, the optimal point of $(P2)$. This is reminiscent of the correspondence established by Todd and Yildirim [33] between Khachiyan's ellipsoidal rounding algorithm [15] and the deepest cut ellipsoid method using two-sided symmetric cuts.

Note that $y/\|d\|^*_U$ lies in the intersection of $\mathcal{B}(U)$ and $\mathcal{H}_d := \mathcal{H}_d(0)$ and that $z := y/\varphi(y)$ is on the boundary of $\mathcal{Q}^0$ and hence is feasible for $(P2)$. This is the current iterate from the perspective of problem $(P2)$. We see our method produces a sequence of points on the boundary of $\mathcal{Q}^0$.

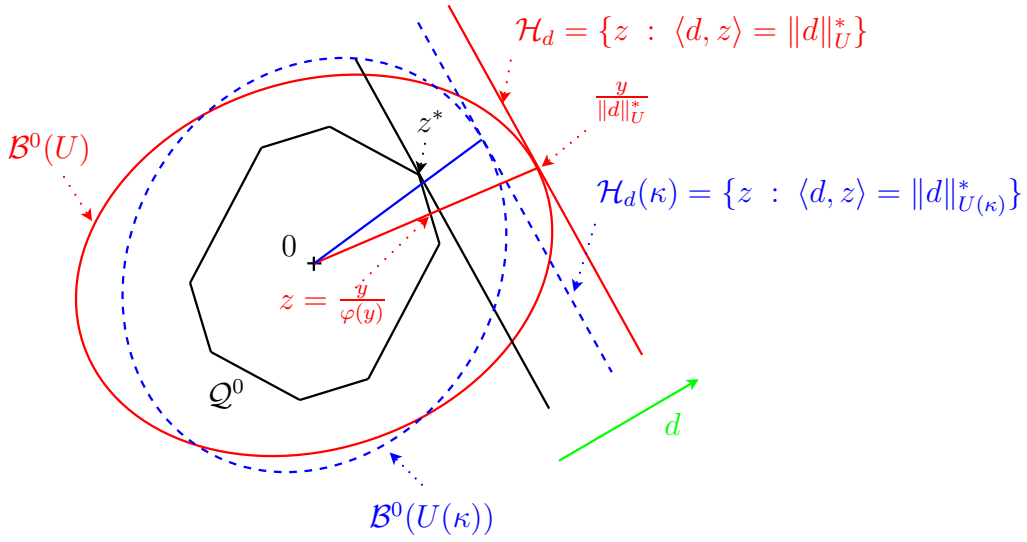



Figure 3.11: The polar algorithm.

### 3.4.3 $(D2)$: An Iteratively Reweighted Least Squares Algorithm

Recall problem $(D2)$:

$$\boxed{\min_v\{\|v\|_1 \ : \ Av = d, \ v \in \mathbf{R}^m\}.} \tag{$D2$}$$

By Lemma 3.2.12, if $w$ is feasible for $(P3)$ and if $y \in \mathbf{E}$ is such that $U(w)y = d$, then $v \in \mathbf{R}^m$ defined by $v_i := w_i\langle a_i, y\rangle$, $i = 1, 2, \ldots, m$, is feasible for $(D2)$ and, moreover,

$$\|v\|_1 \leq \|d\|_{U(w)}^*. \tag{3.60}$$

If we let $W := \mathrm{Diag}(w)$ (i.e. $W$ is the diagonal matrix with the entries of vector $w$ on its diagonal), then, assuming $U(w)$ is invertible, the above definition of $v = v(w)$ can be written as

$$v(w) = WA^*y = WA^*(U(w))^{-1}d = WA^*(AWA^*)^{-1}d. \tag{3.61}$$

We claim that if $w_i > 0$ for all $i$, which is the case in Algorithm 11, then the point $v$ above can be obtained as the (unique) minimizer of a certain $\ell_2$ projection problem. For $w \in \operatorname{rint} \Delta_m$ and $W = \operatorname{Diag}(w)$ consider

$$\boxed{\min_v \{\|W^{-1/2}v\|_2 \ : \ Av = d, \ v \in \mathbf{R}^m\}.} \qquad (D'2)$$

This problem arises from $(D2)$ if we replace the $\ell_1$-norm by the $\ell_2$-norm preconditioned by the inverse of a positive definite diagonal matrix with unit trace. Since the set of minimizers does not change if we further replace the objective function by the quadratic $\frac{1}{2}\|W^{-1/2}v\|_2^2$, the (necessary and sufficient) KKT conditions for $(D'2)$ are

$$W^{-1}v \in \operatorname{range} A^*, \quad Av = d,$$

from which we readily see that the (unique) minimizer of $(D'2)$ coincides with $v(w)$ as defined in (3.61):

$$v^*(w) = WA^*(AWA^*)^{-1}d = v(w).$$

Algorithm 11, as applied to $(D2)$, can therefore be interpreted as follows. At every iteration we maintain a vector of positive weights $w$ which defines a Euclidean norm on $\mathbf{R}^m$ by $v \to \|W^{-1/2}v\|_2$. We then "find" the smallest feasible vector in this norm, update the weights and repeat. The weight $w$ is updated to $w(\kappa)$ as in (3.28). As we have discussed before, the arithmetic complexity of every iteration is only $O(mn)$, which is the work needed to compute $A^*x$ for a given vector $x$.

**Two remarks**

Let us make two additional observations. First, if we wish to define

$$j := \arg\max_i |\langle a_i, y\rangle|$$

in terms of $v^*(w)$, then it is the index for which

$$|[W^{-1}v^*(w)]_j| = \|W^{-1}v^*(w)\|_\infty.$$

Second, notice that

$$\begin{aligned}\|W^{-1/2}v^*(w)\|_2 &= \langle W^{-1/2}WA^*(AWA^*)^{-1}d, W^{-1/2}WA^*(AWA^*)^{-1}d\rangle^{1/2}\\ &= \langle d, (AWA^*)^{-1}AW^{1/2}W^{1/2}A^*(AWA^*)^{-1}d\rangle^{1/2}\\ &= \langle d, (AWA^*)^{-1}d\rangle^{1/2}\\ &= \|d\|^*_{U(w)},\end{aligned}$$

and hence inequality (3.60) can be written as

$$\|v^*(w)\|_1 \leq \|W^{-1/2}v^*(w)\|_2.$$

**The first iterate**

Let $w_0 = (\frac{1}{m}, \ldots, \frac{1}{m})$ denote, as usual, the first iterate of Algorithm 11 (resp. Algorithm 12). Then if $W_0 := \mathrm{Diag}(w_0) = \frac{1}{m}I_m$, we get

$$v_0 := v(w_0) = W_0A^*(AW_0A^*)^{-1}d = A^*(AA^*)^{-1}d.$$

It is easy to see that this is the shortest feasible vector in the $\ell_2$ norm. Since $\|v\|_1 \geq \|v\|_2 \geq \frac{1}{\sqrt{m}}\|v\|_1$ for all $v \in \mathbf{R}^m$, then if $v^*$ is any minimizer of $(D2)$, we have

$$\|v_0\|_1 \leq \sqrt{m}\|v_0\|_2 \leq \sqrt{m}\|v^*\|_2 \leq \sqrt{m}\|v^*\|_1.$$

This shows that the initial iterate $v_0$ is $(\sqrt{m}-1)$-approximate minimizer of $(D2)$.

## 3.5 Applications

In this section we apply the methods of this chapter to two problems both of which can be expressed in the form $(P3)$. The first one is the *truss topology design* — a

civil engineering application. The second is statistical in nature — the computation of a $c$-optimal design.

### 3.5.1 Truss topology design

A *truss* is a construction composed of a network of bars linked to one another such as a crane, scaffolding, bridge, wire-model, etc. One can think of a truss as a graph in two or three dimensions. The graph-theoretic terminology then translates as follows: arcs are called bars, vertices are called nodes.

The nodes of a truss are of two categories: *free nodes* and *rigid nodes*. The rigid nodes are attached to some force-absorbing object such as a wall or the ground. Free nodes are subjected to an external force — a *load*. As a result of the load, the free nodes get displaced and bars joining them stretched or squeezed until the structure assumes an equilibrium position in which the internal tensions in the bars compensate for the external forces. A loaded truss therefore stores a certain amount of potential energy called *compliance*. The more there is of this stored energy, the more sensitive the truss becomes to additional loads and/or load variations. It is therefore desirable to design trusses with as small a compliance as possible, given a collection of loads. In this example we will only describe the situation with a fixed vector of loads acting at the free nodes. It is certainly interesting to also consider the case of load scenarios, or perhaps of a dynamic load. These problems are much harder and are out of the scope of this thesis.

#### The problem

The problem we will consider is the following: *Given a set of free and rigid nodes, a set of possible bar locations, a total weight limit on the truss and a vector of*

*external forces acting on the free nodes, design a truss, i.e. give the locations of the bars and their weights, which is capable of holding the given load and has minimum compliance.*

**Correspondence with the setting of problem** $(P3)$

The actual derivation of the model can be found, for example, in [4]. Let us describe the parameters of the model in terms of the notation of $(P3)$.

The matrix $U(w) = \sum_{i=1}^{m} w_i a_i a_i^T$ is the *bar-stiffness matrix*. The vector $w$ corresponds to the weights of the individual tentative bars, normalized so that the total weight of the bars is 1. Let $p$ be the number of the free nodes. Then we have $a_i \in \mathbf{R}^n$ where either $n = 2p$ or $n = 3p$, depending on whether we have a $2d$ or a $3d$ truss. The system

$$U(w)y = d$$

corresponds to the equilibrium equation between the vector of forces $d$ acting at the free nodes and the vector of displacements $y$ of the free nodes. The compliance is one half of the objective function of problem $(P3)$ squared:

$$Compliance = \tfrac{1}{2}(\|d\|^*_{U(w)})^2.$$

We see that problem $(P3)$ corresponds exactly to the truss topology design problem.

**Three examples**

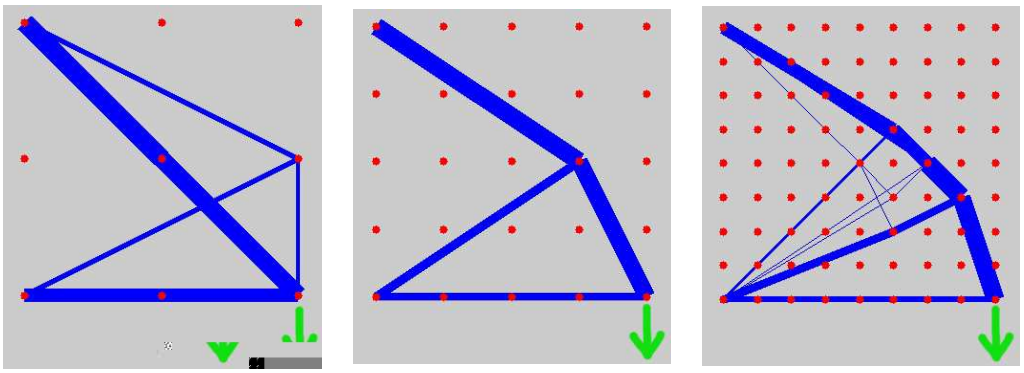
**Example 3.5.1.** A unit vertical download force is applied to the right-bottom node of each of the following three $2d$ trusses:

(a) A $3 \times 3$ truss with 3 fixed nodes attached to a wall (the nodes on the left) and 6 free nodes. Hence $n = 2 \times 6 = 12$. We allow for tentative bars to be

placed among any pair of nodes, with the exception of pairs where there is "overlap" with a chain of other smaller bars. For example, we do not allow placing a bar on the diagonal since this consists of 2 smaller tentative bars already. The number of such potential bars is $m = 28$.

(b) A $5 \times 5$ truss with 5 fixed nodes. We have $n = 2 \times 5 \times 4 = 40$ — the number of free nodes times 2. The number of tentative bars is $m = 400$.

(c) A $9 \times 9$ truss with 9 fixed nodes. In this case $n = 144$ and $m = 2040$.



(a) (b) (c)

Figure 3.12: Three optimal trusses.

Figure 3.12 displays the (approximately) optimal trusses computed with Algorithm **IncDec**. Bars of small weight were removed from the figure. The author wishes to thank Michal Kočvara for sharing his MATLAB code for producing the pictures of the trusses.

Figure 3.13 lists the performance of our **IncDec** method applied to the three problems of Example 3.5.1, with two different accuracy requirements. All computations were done in MATLAB. Let us note that the small $3 \times 3$ problem was solved by the implementation of the simplex method in MATLAB in 0.5 seconds,

| Truss | $n$ | $m$ | $\epsilon = 10^{-1}$ Time | Iteration # | $\epsilon = 10^{-4}$ Time | Iteration # |
|---|---|---|---|---|---|---|
| $3 \times 3$ | 12 | 28 | 0.07 | 413 | 0.07 | 435 |
| $5 \times 5$ | 40 | 200 | 0.15 | 676 | 1.39 | 7850 |
| $9 \times 9$ | 144 | 2040 | 10.77 | 4450 | 367 | 158601 |

Figure 3.13: Performance of Algorithm 12 on three TTD problems.

the medium $5 \times 5$ problem in 0.78 seconds, while the large $9 \times 9$ problem could not be solved by the simplex method within 30 minutes. An interior-point algorithm, however, solved the problem to high accuracy in 0.96 seconds and only 14 iterations.

### 3.5.2 Optimal design of statistical experiments

The presentation of this subsection is largely based on that in Pukelsheim [26]. See also Fedorov [7] and Silvey [30].

Consider the following situation. An experimenter observes a certain scalar quantity $y$ which is assumed to depend *linearly* on a vector $x \in \mathbf{R}^n$ of conditions under his control (a *regression* vector) and a vector of parameters $\theta \in \mathbf{R}^n$ of interest to him. The observation and/or the model is subject to an additive error $e$:

$$y = x^T\theta + e. \tag{3.62}$$

We will assume that the regression vector $x$ can be chosen from among a finite collection of vectors $a_1, \ldots, a_m$, which correspond to the vectors defining $\mathcal{Q}$ — the central object of this chapter. We therefore identify $\mathbf{E}$ and $\mathbf{E}^*$ with $\mathbf{R}^n$.

The statistician wants to estimate a certain function of the parameter $\theta$ and,

in order to do so, decides to observe the outcome under conditions $x_1, \ldots, x_l$. This is called an *experimental design* of sample size $l$. The goal is to construct a design leading to an *unbiased linear estimator*, optimal in a certain sense. Since we restrict the choice of the regression vectors to the finite set $\{a_1, \ldots, a_m\}$, any design can be described by assigning frequencies to the vectors $a_i$. Due to the constraint on the number of observations and the resulting combinatorial structure of feasible frequencies, this approach is usually hard to tackle theoretically. One can instead assign a *weight* $w_i$ to each vector $a_i$, representing the portion of the entire experiment to be spent under the conditions corresponding to this regression vector.

Let us assume that the errors $e_j$ are independent random variables with mean zero and (unknown) constant variance $\sigma^2$ (a *nuisance* parameter). The Fisher information matrix of a design assigning weight $w_i$ to point $a_i$ is given by $U(w) = \sum_i w_i a_i a_i^T$. Now consider the following cases:

- If we wish to minimize the sum of variances of estimators of the individual parameters $\theta_i$, this amounts to the problem of minimizing the trace of $U(w)^{-1}$. This criterion is referred to as *A-optimality*.

- If the goal is to minimize the variance of the (best unbiased linear estimator) of a linear function of the parameter, say $c'\theta$, it turns out that we need to find $w \in \Delta_m$ minimizing $c^T U(w)^{-1} c = (\|c\|^*_{U(w)})^2$. If we let $d := c$, this is equivalent to our main problem $(P3)$ and is referred to as the *c-optimality* criterion in the statistical literature.

- If we wish to minimize the volume of the confidence ellipsoid for $\theta$, this corresponds to the problem of maximizing the determinant of $U(w)$. This is

called the *D-optimality* criterion.

The main problem of this chapter is therefore equivalent to finding the minimum variance unbiased linear estimator of a linear function of the parameter in a statistical linear model with moment assumptions and independent errors.

## BIBLIOGRAPHY


[1] D. Ahipaşaoğlu, P. Sun, and M. J. Todd. Linear convergence of a modified Frank-Wolfe algorithm for computing minimum-volume enclosing ellipsoids. Technical Report TR1452, Cornell University, School of Operations Research and Information Engineering, 2006.

[2] S. Arora, E. Hazan, and S. Kale. The multiplicative weights update method: a meta-algorithm and applications. Technical report, Princeton University, 2005.

[3] C. L. Atwood. Optimal and efficient design of experiments. *The Annals of Mathematical Statistics*, 40:1570–1602, 1969.

[4] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

[5] J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization*. Advanced Books in Mathematics. Canadian Mathematical Society, 2000.

[6] F. A. Chudak and V. Eleutério. Improved approximation schemes for linear programming relaxations of combinatorial optimization problems. In *IPCO'05*, Berlin, 2005.

[7] V. V. Fedorov. *Theory of Optimal Experiments*. Academic Press, New York, 1972.

[8] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

[9] J.-L. Goffin. On convergence rates of subgradient optimization methods. *Mathematical Programming*, 13:329–347, 1977.

[10] G. H. Golub and Ch. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[11] J. Guélat and P. Marcotte. Some comments on Wolfe's 'away step'. *Mathematical Programming*, 35:110–119, 1986.

[12] J. Hartung. An extension of Sion's minimax theorem with an application to a method for constrained games. *Pacific Journal of Mathematics*, 103:401–408, 1982.

[13] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms*. Springer-Verlag, Berlin, 1993.

[14] F. John. Extremum problems with inequalities as subsidiary conditions. In *Studies and Essays, Presented to R. Courant on his 60th Birthday January 8, 1948*, pages 187–204, New York, 1948. Wiley Interscience.

[15] L. G. Khachiyan. Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research*, 21:307–320, 1996.

[16] P. Kumar and E. A. Yıldırım. Minimum volume enclosing ellipsoids and core sets. *Journal of Optimization Theory and Applications*, 126(1):1–21, 2005.

[17] A. Nemirovski and D. Yudin. *Informational Complexity and Efficient Methods for Solution of Convex Extremal Problems*. J. Wiley and Sons, New York, 1983.

[18] Yu. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(\frac{1}{k^2})$. *Doklady AN SSSR (translated as Soviet. Math. Docl.)*, 269(3):543–547, 1983.

[19] Yu. Nesterov. *Introductory Lectures on Convex Optimization. A Basic Course*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, 2004.

[20] Yu. Nesterov. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization*, 16(1):235–249, 2005.

[21] Yu. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.

[22] Yu. Nesterov. Rounding of convex sets and efficient gradient methods for linear programming problems. *CORE Discussion Paper #2004/04*, January 2004.

[23] Yu. Nesterov. Unconstrained convex minimization in relative scale. *CORE Discussion Paper #2003/96*, November 2003.

[24] Yu. Nesterov. Smoothing technique and its applications in semidefinite optimization. *CORE Discussion Paper #2004/73*, October 2004.

[25] Yu. Nesterov and A. Nemirovski. *Interior-Point Polynomial Algorithms in Convex Programming*, volume 13 of *SIAM Studies in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1994.

[26] F. Pukelsheim. *Optimal Design of Experiments (Classics in Applied Mathematics)*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2006.

[27] J. Renegar. *A Mathematical View of Interior-Point Methods in Convex Optimization*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001.

[28] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, USA, 1997. Reprint of the 1970 original, Princeton Paperbacks.

[29] N. Z. Shor. *Minimization Methods for Nondifferentiable Functions*. Springer-Verlag, Berlin, 1985.

[30] S. D. Silvey. *Optimal Design: An Introduction to the Theory for Parameter Estimation*. Chapman and Hall, New York, 1980.

[31] M. Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8:171–176, 1958.

[32] P. Sun and R. M. Freund. Computation of minimum-volume covering ellipsoids. *Oper. Res.*, 52(5):690–706, 2004.

[33] M. J. Todd and E. A. Yıldırım. On Khachiyan's algorithm for the computation of minimum volume enclosing ellipsoids. Technical Report TR1435, Cornell University, School of Operations Research and Information Engineering, 2005.

[34] J. von Neumann and O. Morgenstern. *The Theory of Games and Economic Behavior*. Princeton University Press, Princeton, NJ, USA, 1948.

[35] P. Wolfe. Convergence theory in nonlinear programming. In J. Abadie, editor, *Integer and Nonlinear Programming*, pages 1–36, North-Holland, Amsterdam, 1970.

[36] H. P. Wynn. The sequential generation of D-optimum experimental design. *The Annals of Mathematical Statistics*, 41:1655–1664, 1970.

[37] E. A. Yıldırım. On the minimum volume covering ellipsoid of ellipsoids. *SIAM Journal on Optimization*, 17(3):621–641, 2006.

[38] F. Zhang. *Matrix Theory: Basic Results and Techniques*. Springer, New York, 1999.