"Cones and interior-point algorithms for structured convex optimization involving powers and exponentials"

Chares, Robert

Abstract

Optimization is an important field of applied mathematics with many applications in various domains, ranging from mechanical and electrical engineering to finance and operations research. In particular, convex optimization is very popular because of the availability of highly efficient methods supported by strong theoretical results. In this thesis, we study interior-point methods whose computing time is guaranteed to grow polynomially with the problem dimension. These methods can be applied to any convex problem provided a special function known as a self-concordant barrier is available for the given formulation. We demonstrate in this work that a large class of convex optimization problems are representable in a convex conic form based on the so-called power cone. This very general formulation unifies well-known problem classes such as linear and convex quadratic optimization, but also problem classes such as geometric programming or p-norm location problems. Moreover, we show that...

Document type : *Thèse (Dissertation)*

Référence bibliographique

Chares, Robert. *Cones and interior-point algorithms for structured convex optimization involving powers and exponentials.* Prom. : Glineur, François

Université catholique de Louvain
École Polytechnique de Louvain

Cones and Interior-Point Algorithms for Structured Convex Optimization involving Powers and Exponentials

Thèse présentée en vue de l'obtention du grade de
Docteur en Sciences de l'Ingénieur par

Peter Robert Chares

Promoteur: F. Glineur
Jury: V. Blondel (Président)
F. Jarre
Y. Nesterov
C. Roos
A. Sartenaer

Acknowledgements

A Ph.D. is a work that cannot be accomplished without the help and support of others.

First, I would like to express my sincere gratitude to François Glineur. Being his first Ph.D. student, this work was certainly an exciting experience for both of us. I am very grateful for guiding me through these four years. His inexhaustible surge of ideas helped me a lot in progressing in my research.

I also would like to thank Yurii Nesterov. I have had the great honor of being part of his group at CORE and witnessing moments of his magic in numerous scientific meetings. His deep insights into the field of optimization were a constant source of inspiration. I wish to thank Annick Sartenaer for accepting to be a member of my doctoral committee. Her refreshing way of questioning certain issues of this work helped me improve this thesis.

I would like to thank Professors Cornelis Roos and Florian Jarre for accepting to be members of the jury and enhancing substantially the quality of this work. Many thanks to Professor Vincent Blondel for being the president of the jury and Professor Auguste Laloux for presiding the ceremony of the public defense.

The work on this thesis has been financially supported by research grants from Université catholique de Louvain (FSR) and Fonds National de la Recherche Scientifique (FRIA) and Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. I would like to express my gratitude to Erling Andersen, Cédric Druck, Gert Wanka and Bernd Inhester for helping me with various research proposals.

I am very happy to have spent my doctoral studies at CORE, an exceptional place to follow research and develop scientific ideas. I wish to thank Catherine and Mady for kindly helping me with all kinds of administrative matters and Laurent for very efficient computer-related assistance. There were many people at CORE with whom I have shared lots of discussions and coffee breaks. A warm ”thank you” for these nice moments goes to François, Michel, Peter M., Ruslan, Ingmar, Eissa, João, Kjetil, Joël, Jean-Sébastien, Santanu, Rafael, Peter R. and Nicolas.

Finally, I would like to express my deep gratitude to my family: my parents for their unconditional support over the years, my wonderful wife Katerina for her love and patience and my lovely daughter Antigone for having the patience to enter my life after the private defense of this thesis.

Contents

List of notations

Vector spaces and sets

$\mathcal{E}$	a general vector space,
$\mathcal{E}^*$	the dual vector space to $\mathcal{E}$,
$\mathbb{R}$	the set of real numbers,
$[a, b]$	the closed interval of real numbers between a and b,
(a, b)	the open interval of real numbers between a and b,
$\mathbb{R}^n$	the vector space of n-dimensional vectors,
$\mathcal{S}^n$	the vector space of symmetric $n \times n$-matrices,
x_i	the i-th component of the vector $x \in \mathbb{R}^n$,
$x^{(k)}$	the k-th element in a family or sequence of vectors,
$\mathbb{R}^n_+$	the cone of nonnegative n-dimensional vectors,
$\mathbb{R}^n_{++}$	the cone of strictly positive n-dimensional vectors,
$\mathbb{L}^n$	the second-order cone,
$\mathcal{S}^n_+$	the cone of positive semidefinite matrices,
$\mathcal{S}^n_{++}$	the cone of positive definite matrices,
$\operatorname{dom} F$	the effective domain of a function $F : \mathcal{E} \to \mathbb{R}$, $\operatorname{dom} F = \{x \in \mathcal{E} : F(x) < \infty\}$,
$\operatorname{epi}(F)$	the epigraph of a function $F : \mathcal{E} \to \mathbb{R}$, $\operatorname{epi}(F) = \{(x, t) \in \mathcal{E} \times \mathbb{R} : F(x) \leq t\}$,
$\operatorname{int}(\mathcal{Q})$	the interior of a set $\mathcal{Q}$,
$\operatorname{ri}(\mathcal{Q})$	the relative interior of a set $\mathcal{Q}$,
$\operatorname{cl}(\mathcal{Q})$	the closure of a set $\mathcal{Q}$,
$\partial(\mathcal{Q})$	the boundary of a set $\mathcal{Q}$,
$\mathcal{C}^k$	the vector space of k-times continuously differentiable functions.

Functions

$\log(x)$	the natural logarithm of a positive number x,
$D^kF(x)[h_1,\dots,h_k]$	the k-th directional derivative of F at the point $x \in \operatorname{dom} F$ in the direction $(h_1,\dots,h_k)$,
$\nabla F(x,y)$	the gradient of F at $(x,y) \in \operatorname{dom} F$,
$F'_x(x,y)$	the partial derivative of F with respect to the variable x, if $x \in \mathbb{R}^n$ is a vector, then $F'_x(x,y) \in \mathbb{R}^n$ denotes the vector of partial derivatives of F with respect to x,
$\nabla^2 F(x,y)$	the Hessian of F at $(x,y) \in \operatorname{dom} F$,
$F''_{xx}(x,y)$	the second derivatives of F only with respect to x.
$\lvert\lvert x\rvert\rvert_2$	the Euclidean norm of the point $x \in \mathbb{R}^n$: $\lvert\lvert x\rvert\rvert_2 = (\sum_{i=1}^n x_i^2)^{1/2}$,
$\lvert\lvert x\rvert\rvert_p$	the p-norm ($p \geq 1$) of the point $x \in \mathbb{R}^n$: $\lvert\lvert x\rvert\rvert_p = (\sum_{i=1}^n \lvert x_i\rvert^p)^{1/p}$,
$\lvert\lvert x\rvert\rvert_S$	the norm of the point x, induced by the positive definite matrix S: $\lvert\lvert x\rvert\rvert_S = (x^TSx)^{1/2}$.

Linear algebra

$M \in \mathbb{R}^{m,n}$	a matrix with m rows and n columns,
$M_{i,j}$	for a matrix $M \in \mathbb{R}^{m,n}$, its component in the i-th row and j-th column,
M_j	for a matrix $M \in \mathbb{R}^{m,n}$, the j-th column,
I	the identity matrix,
$\texttt{blkdiag}_N(X)$	the block diagonal matrix containing N copies of the matrix X,
$\texttt{rep}_N(X)$	the N times (horizontal) repetition of a matrix X, i.e. $\texttt{rep}_N(X) = [X,\dots,X]$.

CHAPTER 1

Introduction

1.1 Motivation and goals

Optimization is the field of applied mathematics where one wants to minimize or maximize an objective function in several variables. In real-life applications the variables correspond to parameters that have to be chosen within some range. The set of all combinations of these parameter choices is called the *set of feasible solutions* or *feasible set*. For example a variable could be the thickness of a bar in a mechanical construction, or the amount of money invested in a certain asset of a portfolio. Our goal is to find an optimal choice of the variables in the set of feasible solutions in the sense that it optimizes the objective function. The objective could be for example a measure of stability of a construction or the return of an investment.

These three ingredients – variables, feasible set and objective function – determine a generic optimization problem. Unfortunately, general optimization problems are difficult to solve. For almost all problem classes it is impossible to write down analytically a closed-form optimal solution. Instead, iterative methods have to be employed in order to generate a sequence of iterates that eventually converge to an optimal solution. Since the algorithm should stop after a finite amount of time, it typically only provides an *approximation* for an optimal solution. But even worse, that approach of approximating an optimal solution might fail. In fact, it can be shown (see [46]) that even for rather well-behaved problem classes (with an objective that is Lipschitz continuous and a feasible set that is the n-dimensional unit box), any method that makes only use of function values (no derivatives) has an exponential worst-case complexity. For example if the number of variables is $n = 10$ (which is rather small for applications), the objective has a Lipschitz constant of $L = 2$ and we require a low absolute accuracy in terms of the objective value of $\epsilon = 0.01$, then using a computer that can evaluate 10^{10} times the objective function per second, it could still take more than 300 years to solve the problem. Similar results hold when derivatives are available.

Of course, this result is of worst-case nature. Very often methods are much faster on many instances than predicted by their worst-case complexity bounds, like for example the simplex method for linear programming. However, the example illustrates that too few assumptions in terms of the formulation of the problem and too little information used in the design of the algorithm might result in bad performance for some optimization problems. On the other hand, if we provide more information about the concrete problem (like for example derivatives of an objective function), we implicitly restrict ourselves to a smaller, less general, class of problems (such as the class of problems with differentiable objective) and expect better algorithms to become available. In other words, there is a trade-off between generality and theoretical and/or practical performance. Unfortunately, in some situations we might not have much choice when it comes to the methods, because the problem is extremely large or the function values are the output of some numerical simulation. In these cases first- and second-order methods are either not practical or simply not applicable because we have only very limited information available about the problem. As a consequence we cannot expect a method for these problems to give a solution with high accuracy in a reasonable amount of time.

In this thesis we follow the opposite approach: we consider the class of second-order methods, because up to now these methods have the best known theoretical properties. Within this class we focus on interior-point methods which are naturally defined for convex optimization problems. We argue that the restriction to these problems is rewarding. Indeed, there is a huge amount of literature on convex optimization (see e.g. [3],[6],[5],[9],[24],[15],[66]) showing the great scope of potential applications, ranging from mechanical and electrical engineering, to finance, network design, location problems and many more. Moreover, even if the original problem might not be convex, often it is possible to approximate it well by some convex problem (see e.g. [64],[45],[25]).

The main objective of this thesis is to establish a unified algorithmic framework for a principal problem class with the following three properties: 1) any instance of the chosen problem class should be solvable with a complexity that is polynomial in the problem size, 2) the chosen problem class should be sufficiently general in the sense that many sets and functions are representable in terms of that basic formulation, and 3) the framework should exploit structure to overcome the two main drawbacks which are inherent to the class of interior-point methods: its high memory storage and the high cost per iteration, due to the evaluation and storage of the gradient and Hessian of a multivariate nonlinear function, and the computation of search directions by solving a linear system at each iteration.

1.2 Some historical remarks

As we outlined above, optimization can be applied in a multitude of areas in every day's life. This fact has been known now for a long time. The beginning of the 20th century saw increasing research activity mainly on the theoretical side of optimization (see e.g. [32],[38],[60]) like duality theory and optimality conditions.

In the late 1940's, research activity on algorithms for optimization problems became stronger, driven by an increasing interest from the side of applications.

In 1947 Dantzig ([13, 14]) proposed the simplex method for solving linear optimization problems. At that time the main criteria for evaluating the quality of algorithms were the convergence to an optimal solution and finiteness of the algorithm. The simplex method goes from one vertex of the polyhedron of feasible solutions to a neighbor that improves the objective function. As there is a finite number of vertices on the boundary of the feasible set and because an optimal solution must be situated on one of them, it is clear that both above conditions are satisfied by the algorithm. That is why this method is popular and still very successfully used today.

In the late 60's and early 70's a new concept for evaluating algorithms has been introduced: the *complexity* of an algorithm does not only take into account the convergence to an optimal solution and its finiteness, but it also shows how the number of iterations depends on the size of the problem. In 1972 Klee and Minty ([34]) showed by an example that the worst-case complexity of the simplex method is in fact exponential in the size of the problem. Already in the 1950's an increased activity started in the area of nonlinear optimization ([21],[29]) which culminated in the seminal work of Fiacco and McCormick ([17]). They proposed in 1968 the method of sequential unconstrained minimization as a technique to tackle nonlinear constrained problems. Later ([1]) their method was even shown to be polynomial for linear programming problems. In 1970 Shor ([59]) proposed the ellipsoid method, used later by Khachiyan in 1979 ([33]) to prove for the first time polynomial complexity of a method for solving linear programs. In the same year Nemirovski and Yudin ([43]) proposed a framework for solving the general class of convex optimization. Among others, they showed that the complexity of solving this class is in the best case $\mathcal{O}(n \log(1/\epsilon))$, where n is the number of the variables and ϵ the desired accuracy. This result is the main tool for showing *optimality of a method for convex optimization problems.*

In 1984 started the interior-point revolution with the seminal work of Karmarkar ([31]) who proposed another practically efficient polynomial-time interior-point method for linear programming. In 1994 Nesterov and Nemirovski ([52]) generalized this result to convex optimization problems. They introduced the notion of self-concordant functions and barriers. In 1997 and 1998 Nesterov and Todd ([53],[54]) extended the framework of polynomial interior-point methods to the conic setting and proposed symmetric primal-dual interior-point methods for a class of convex problems that nicely generalizes the previously known primal-dual methods for linear programming to the convex case. Since the polynomial interior-point breakthrough during the 1990's there have been thousands of publications on interior-point methods for specific classes of convex optimization problems and improvements or extensions of the original results [52, 53, 54].

During the 90's started some further research activity on the field of interior-point methods for general nonlinear (and nonconvex) optimization problems (see e.g. [18], [65]). These methods share similar ideas with the methods from the convex optimization community, in particular the use of barrier and/or penalty functions to treat nonlinear constraints and the use of primal-dual optimality conditions.

1.3 Thesis overview and main contributions

This thesis is organized as follows. Chapter 2 provides the necessary background on convex optimization and in particular conic optimization. We present the key results about path-following interior-point methods to solve generic convex optimization problems.

One contribution in this chapter is the less commonly encountered generalization of known results for convex optimization to the situation where linear equality constraints are present (Sections 2.3 - 2.5). It is known that the presence of linear equalities can always be circumvented by removing some of the variables. However, we kept the original formulations with equality constraints and gave insight on the overall effect on Newton's method (in Section 2.3) and on path-following methods (in Section 2.4 and Section 2.5). Furthermore, with the aim to make the primal-dual conic frameworks accessible for a large class of convex problems we propose and analyze a primal-dual predictor-corrector method that exhibits a polynomial algorithmic complexity. We stress here that this method does not make use of a pair of barriers for the primal and dual cone that are conjugate to each other. For that reason this nonsymmetric primal-dual method is applicable whenever there is a self-concordant barrier for the dual cone at hand[1].

The main contribution of Chapter 3 is the proof of self-concordance of a new barrier for the power cone with parameter $\nu = 3$. As a direct consequence of this result we obtain a self-concordant barrier for the epigraph of convex power functions with an improved self-concordance parameter. The previously known barrier had a parameter that was clearly non-optimal, which was stated already by Nesterov in [46, Section 4.3.5.4] (however, the obtained parameter value is still not optimal, e.g. for the power cone with $\alpha = \frac{1}{2}$). In view of that observation we mention numerical tests which suggest that a scaling of the barrier for the power cone is possible which results in a self-concordance parameter between 2 and 3. In a similar vein we compute numerically the universal barrier for the three-dimensional p-cone. Our numerical tests suggest that the self-concordance parameter is situated between 2 and 3. The self-concordance proof for the new barrier for the power cone and its generalizations as well as its direct implications (improved self-concordance parameter of convex powers) can be found in [10], a paper entitled *New self-concordant barriers for the power cone.*

The third chapter provides the stepping stone for the rest of the thesis, in that we give in Chapter 4 a clear description of the scope of the power cone by listing power cone representable sets and functions. It turns out that many convex constraints involving powers, exponentials and logarithms are representable using the power cone. The conic reformulation has the advantage that it allows the design of polynomial-time interior-point methods. In order to benefit additionally from the primal-dual framework, we compute the dual cones of the power cone itself and of a limit of the power cone. Knowledge of these dual cones is essential when one wants to use the primal-dual predictor-corrector method proposed earlier in Section 2.5. We present two important problem classes (that is, generalized location problems and geometric programs) that can be cast in conic form using power

[1] The primal variant of the proposed method has been presented earlier by Nesterov [49] for the case where the primal cone is proper, i.e. where there are no primal free variables present.

cones, with numerical results that compare the dual and primal-dual interior-point methods to other solvers for these problem classes. We show that the dual and primal-dual interior-point method are competitive with respect to the number of iterations and reliability, but not with respect to the overall computation time. This relatively disappointing result can be partially explained by the fact that the other compared solvers do not rely on methods with polynomial complexity. Instead, they solve directly the original problem which has the effect that the cost of one single iteration is much lower compared to the conic reformulation, where many artificial variables have to be introduced. On the other hand, there are situations where nonlinear solvers fail even for tiny instances, as we show at the end of the fourth chapter for problems involving mixed powers. The results from Section 4.4.1 and Section 4.5.1 in combination with a complete complexity proof of the proposed path-following method have been published in a paper entitled *An interior-point method for the location problem with mixed norms using a conic formulation* in [9].

Based on the numerical results in Chapter 4 we present in Chapter 5 a new framework that can be embedded in interior-point methods with the aim to reduce the cost per iteration. The underlying motivation is a result which says that the partial minimization of a self-concordant barrier preserves the property of self-concordance. Based on this result we propose a two-level interior-point scheme where in each outer iteration we solve the partial minimization subproblem approximately. Using this approximate solution we compute a direction that can be thought of as a Newton direction in an affine subspace that is approximately tangent to the surface of partial minimizers. We show that this sequence of approximately tangent subspaces approaches the minimizer of the current centering problem. Eventually, as soon as the Newton decrement for both the partial minimization subproblem and the outer problems in the sequence of subspaces are small, we can conclude that the current iterate is a good approximation for the overall minimizer. We show that polynomial complexity is preserved when embedding partial minimization into an interior-point scheme, even if the partial minimization can only be done approximately. In this sense the partial minimization framework benefits from the self-concordance of the barrier for the extended reformulation and the reduction of the problem size by restricting to the subspaces. Moreover, we demonstrate that this new technique also works in practical implementations, where we observe a reduction in the cost per iteration, and also in the total number of iterations. The results on partial minimization from Chapter 5 are contained in [11], which covers the generalized implicit barrier theorem, the entire framework of approximate partial minimization, examples of potential applications and the numerical results.

CHAPTER 2

Convex optimization

In this chapter we consider convex optimization problems. We start with the basic concepts of convexity for sets and functions (Section 2.1). Later we present different classes of convex optimization problems, first convex problems without any constraints (Section 2.2), then convex problems with linear equality constraints (Section 2.3), which turns out to be essentially equivalent to the class of unconstrained problems. In Section 2.4 we consider convex problems with inequality constraints. Finally, in Section 2.5 we present a unified format for convex optimization problems: convex problems in conic form.

The first two sections are to a large extent a collection of known results that can be found in the standard literature on convex optimization (e.g. [56],[6],[2]) and on interior-point methods (e.g. [52],[46],[55]). The results in Sections 2.3 - 2.5 are generalizations of previously known results to the case where linear equality constraints are present in the model. Instead of removing these constraints (e.g. by Gaussian elimination) we explicitly keep them in the model and phrase all the results and their proofs in terms of the original problem. In Section 2.5 we present a nonsymmetric primal-dual predictor-corrector method, whose primal variant has been proposed earlier by Nesterov [49]. Our extension is formulated for dual conic problems with linear equality constrains. Moreover, we give an explicit description of a safe step length for the primal-dual affine-scaling direction.

2.1 Convexity

In order to analyze convex optimization problems, we need to fix the concepts of *convex sets* and *convex functions.*

2.1.1 Convex sets

Let $\mathcal{E}$ be an n-dimensional vector space. Its dual space $\mathcal{E}^*$ is the space of linear functionals mapping from $\mathcal{E}$ to $\mathbb{R}$. Most of the results in this thesis are valid for general vector spaces $\mathcal{E}$. However, if nothing else is specified, we consider the special case $\mathcal{E} = \mathcal{E}^* = \mathbb{R}^n$.

Definition 2.1.1. *A set $\mathcal{C} \subseteq \mathcal{E}$ is said to be* convex *if for any pair of points $x \in \mathcal{C}$ and $y \in \mathcal{C}$ the whole line segment between these two points belongs to $\mathcal{C}$, i.e.* $\forall \lambda \in [0, 1]$

$$z := \lambda x + (1 - \lambda) y \in \mathcal{C}.$$

Examples

- $\mathcal{C} = \mathcal{E}$, $\mathcal{C} = \emptyset$,
- $\mathcal{C}$ is a half-space (for $a \neq 0$ we have $\mathcal{C} = \{x : a^T x \leq b\}$) or a hyperplane ($\mathcal{C} = \{x : a^T x = b\}$, where $a \neq 0$),
- $\mathcal{C}$ is an ellipsoid around some point $x_0 \in \mathcal{E}$, $\mathcal{C} = \{x : (x - x_0)^T S (x - x_0) \leq 1\}$, where S is a positive definite matrix,
- $\mathcal{C}$ is the set of positive semidefinite matrices (here $\mathcal{E} = S^n$ the set of symmetric matrices).

Operations that preserve convexity

In order to check whether a given set $\mathcal{C}$ is convex, one can directly try to verify Definition 2.1.1. Another way is to show that $\mathcal{C}$ is in fact the image of another set $\tilde{\mathcal{C}}$ (which is known to be convex) under a transformation that preserves convexity. Some of these convexity-preserving operations are listed below. Their proofs can be found e.g. in [56], [6] or[2].

1. **Intersection.**
   Let $\mathcal{C}_1 \subseteq \mathcal{E}$ and $\mathcal{C}_2 \subseteq \mathcal{E}$ be convex sets. Then $\mathcal{C}_1 \bigcap \mathcal{C}_2$ is convex. This generalizes to any family of convex sets $\{\mathcal{C}_\tau\}_{\tau \in I}$ for some index set I. Note that I can have infinite cardinality.

2. **Direct product.**
   Let $\mathcal{C}_1 \subseteq \mathcal{E}_1$ and $\mathcal{C}_2 \subseteq \mathcal{E}_2$ be convex, where $\mathcal{E}_i$ are two vector spaces. Then the direct product (Cartesian product) of $\mathcal{C}_1$ and $\mathcal{C}_2$ is defined as

   $$\mathcal{C}_1 \times \mathcal{C}_2 = \{(x, y) : x \in \mathcal{C}_1, y \in \mathcal{C}_2\}.$$

   It turns out that $\mathcal{C}_1 \times \mathcal{C}_2$ is convex.

3. **Compositions with affine function.**
   Let $\mathcal{C} \subseteq \mathbb{R}^n$ be convex and $\mathcal{A} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ such that $\mathcal{A}(x) = Ax + b$ for some $A \in \mathbb{R}^{m,n}$ and $b \in \mathbb{R}^m$. Then

   $$\mathcal{A}(\mathcal{C}) := \{\mathcal{A}(x), x \in \mathcal{C}\} \subseteq \mathbb{R}^m$$

   is convex.

4. **Inverse image of affine function.**
Let $\mathcal{C} \subseteq \mathbb{R}^n$ be convex and $\mathcal{A} : \mathbb{R}^p \to \mathbb{R}^n$ such that $\mathcal{A}(y) = By + c$ for some $B \in \mathbb{R}^{n,p}$ and $c \in \mathbb{R}^n$. Then
$$\mathcal{A}^{-1}(\mathcal{C}) := \{y : \mathcal{A}(y) \in \mathcal{C}\} \subseteq \mathbb{R}^p$$
is convex.

5. **Conic hull.**
Let $\mathcal{C} \subseteq \mathcal{E}$ be convex. Then its conic hull
$$\operatorname{cone}(\mathcal{C}) := \left\{(x, t) \in \mathcal{E} \times \mathbb{R}_{++} : \frac{x}{t} \in \mathcal{C}\right\}$$
is convex.

6. **Polar set.**
Let $\mathcal{C} \subseteq \mathcal{E}$ be convex. Then its polar set $\mathcal{C}^o \subseteq \mathcal{E}^*$
$$\mathcal{C}^o := \{s \in \mathcal{E}^* : \langle s, x\rangle \leq 1, \forall\, x \in \mathcal{C}\}$$
is convex.

7. **Minkowski sum.**
Let $\mathcal{C}_1 \subseteq \mathcal{E}$ and $\mathcal{C}_2 \subseteq \mathcal{E}$ be convex sets. Then the Minkowski sum
$$\mathcal{C}_1 + \mathcal{C}_2 := \{x + y, x \in \mathcal{C}_1, y \in \mathcal{C}_2\}$$
is convex.

2.1.2 Convex functions

Definition 2.1.2. *A function $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ is said to be* convex *if its domain $\mathcal{C}$ is convex and for any $x \in \mathcal{C}, y \in \mathcal{C}$ and $\lambda \in [0, 1]$ we have*
$$F(\lambda x + (1 - \lambda)y) \leq \lambda F(x) + (1 - \lambda)F(y).$$
A function $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ is said to be concave *if $-F$ is convex.*

If F is differentiable, then we implicitly assume that $\mathcal{C} = \operatorname{dom} F$ is an open set. In that case we have an alternative definition: F is convex if and only if $\mathcal{C}$ is convex and for any $x \in \mathcal{C}$ and $y \in \mathcal{C}$ it holds
$$F(y) \geq F(x) + \nabla F(x)^T (y - x). \tag{2.1}$$
If F is twice differentiable, then we have that F is convex if and only if $\mathcal{C}$ is convex and
$$\nabla^2 F(x) \succeq 0,\ \forall\, x \in \mathcal{C},$$
that is, the Hessian of F must be positive semidefinite on $\mathcal{C}$.

Definition 2.1.3. *A function $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ is said to be* strictly convex *if $\mathcal{C}$ is convex and for any $x \in \mathcal{C}, y \in \mathcal{C}$, $x \neq y$ and $\lambda \in (0, 1)$ we have*
$$F(\lambda x + (1 - \lambda)y) < \lambda F(x) + (1 - \lambda)F(y).$$

Just like in the case of convexity (as opposed to strict convexity) we can phrase Definition 2.1.3 in terms of the derivatives of F (provided that F is differentiable, of course). If F is differentiable, then we have that F is strictly convex if and only if $\mathcal{C}$ is convex and for any $x \in \mathcal{C}, y \in \mathcal{C}$, $x \neq y$ we have

$$F(y) > F(x) + \nabla F(x)^T(y - x).$$

Similarly, if F is twice differentiable then we have that if

$$\nabla^2 F(x) \succ 0, \; \forall\, x \in \mathcal{C}$$

then F is strictly convex. Note that the condition $\nabla^2 F(x) \succ 0, \forall\, x \in \mathcal{C}$ is *not necessary* for strict convexity of F. For example $F(x) = x^4$ is strictly convex but $F^{''}(0) = 0$.

Examples of convex functions

- quadratic functions $F(x) = x^T A x + a^T x + \alpha$, where $A \in \mathcal{S}^n_+$, $\operatorname{dom} F = \mathbb{R}^n$,
- exponential $F(x) = \exp(x)$, $\operatorname{dom} F = \mathbb{R}$,
- logarithm $F(x) = -\log(x)$, $\operatorname{dom} F = \mathbb{R}_{++}$,
- entropy function $F(x) = x\log(x)$, $\operatorname{dom} F = \mathbb{R}_{++}$ (note that we can include the point $x = 0$ in the domain by defining $F(0) = 0$),
- $F(x) = |x|^p$, for $p \geq 1$ or $p \leq 0$ (with $\operatorname{dom} F = \mathbb{R}_+$ if $p \geq 1$ and $\operatorname{dom} F = \mathbb{R}_{++}$ if $p \leq 0$),
- $F(x) = ||x||$, where $||\cdot||$ is *any* norm on $\mathcal{E}$, $\operatorname{dom} F = \mathcal{E}$,
- $F(x) = \max_{i=1\ldots n}\{x_i\}$, $\operatorname{dom} F = \mathbb{R}^n$.

Operations that preserve convexity

Similar to the case of convex sets there are certain operations that preserve convexity of functions. This means that in order to show that a given function F is convex it suffices to show that F is the image of a convex function under one of these transformations. We list some of these transformations below. Their proof can be found e.g. in [56] or [6].

1. **Nonnegative weighted sum.**
   Let $F_i : \mathcal{C}_i \subseteq \mathcal{E} \to \mathbb{R}, i = 1, \ldots, m$ be convex functions and $\alpha_i \geq 0, i = 1, \ldots, m$. Then
   $$F(x) = \sum_{i=1}^{m} \alpha_i F_i(x)$$
   is convex on $\bigcap_{i=1}^{n} \mathcal{C}_i$.

2. **Nonnegative weighted sum of separable convex functions.**
Let $F_i : \mathcal{C}_i \subseteq \mathcal{E}_i \to \mathbb{R}, i = 1, \ldots, m$ be convex functions and $\alpha_i \geq 0, i = 1, \ldots, m$. Then
$$F(x) = \sum_{i=1}^{m} \alpha_i F_i(x_i)$$
is convex on $\mathcal{C}_1 \times \ldots \times \mathcal{C}_m$.

3. **Composition with affine function.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be convex, $B \in \mathbb{R}^{n,p}$ and $c \in \mathbb{R}^n$. Then
$$\tilde{F}(y) = F(By + c)$$
is convex on $\operatorname{dom} \tilde{F} = \{y \in \mathbb{R}^p : By + c \in \mathcal{C}\}$.

4. **Restriction to affine subspace.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be convex, and $\mathcal{L} = \{x \in \mathbb{R}^n : Ax = b\}$ an affine subspace, where $A \in \mathbb{R}^{m,n}$ and $b \in \mathbb{R}^m$ for $m < n$. Then the restriction of F to $\mathcal{L}$, $F|_{\mathcal{L}} : (\mathcal{C} \bigcap \mathcal{L}) \to \mathbb{R}$ such that
$$F|_{\mathcal{L}}(x) = F(x), \text{ for } x \in \mathcal{L}$$
is convex.

5. **Perspective.**
Let $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ be convex. Then
$$G(x, t) = tF(x/t)$$
is convex on $\operatorname{dom} G = \{(x, t) \in \mathcal{E} \times \mathbb{R}_{++} : x/t \in \mathcal{C}\}$.

6. **Pointwise supremum.**
Let $F_\tau : \mathcal{C}_\tau \subseteq \mathcal{E} \to \mathbb{R}$ with $\tau \in I$, for some index set I, be a family of convex functions. Then
$$F(x) = \max_{\tau \in I} F_\tau(x)$$
is convex with domain $\bigcap_{\tau \in I} \mathcal{C}_\tau$. Note that I can have infinite cardinality.

7. **Conjugate function.**
Let $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ (not necessarily convex). Then its conjugate
$$F_*(s) = \sup_{x \in \mathcal{C}} \{\langle s, x \rangle - F(x)\}$$
is convex on $\operatorname{dom} F_*$, which is the set of all points $s \in \mathcal{E}^*$ such that the supremum above is finite.

8. **Partial minimization.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$, such that $(x, y) \mapsto F(x, y)$, be convex and bounded from below. Then the partial minimization of F with respect to y
$$G(x) = \inf_{y \in \mathcal{Q}(x)} F(x, y),$$
where $\mathcal{Q}(x) = \{y : (x, y) \in \mathcal{C}\}$, is convex on $\operatorname{dom} G = \{x : \mathcal{Q}(x) \neq \emptyset\}$.

9. **Composition with non-decreasing convex functions.**
   Let $F_1 : \mathcal{C}_1 \subseteq \mathbb{R} \to \mathbb{R}$ be nondecreasing and convex, $F_2 : \mathcal{C}_2 \subseteq \mathbb{R}^n \to \mathbb{R}$ be convex. Then
   $$F(x) := F_1(F_2(x))$$
   is convex on $\operatorname{dom} F = \{x \in \mathcal{C}_2 : F_2(x) \in \mathcal{C}_1\}$.

There are strong links between convex sets and convex functions, relying on the notion of epigraphs.

Definition 2.1.4. *The* epigraph *of a function $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ is defined as the set*
$$\operatorname{epi}(F) := \{(x, t) \in \mathbb{R}^n \times \mathbb{R} : x \in \mathcal{C} \textit{ and } F(x) \leq t\}.$$

A function F is convex if and only if its epigraph is convex:

$$F \text{ convex} \Leftrightarrow \operatorname{epi}(F) \text{ convex}.$$

2.1.3 Convex optimization

Convex optimization problems consist in the minimization of a convex objective function F over a convex set $\mathcal{C}$. There are immediately two advantages of convex optimization problems over general nonlinear problems.

1. Any locally optimal solution is also globally optimal, i.e. if
   $$F(x^*) \leq F(x) \; \forall x \in \operatorname{dom} F \bigcap \mathcal{C} \bigcap \mathcal{N}(x^*)$$
   holds for some neighborhood $\mathcal{N}(x^*)$ around $x^* \in \mathcal{C}$ then
   $$F(x^*) \leq F(x) \; \forall x \in \operatorname{dom} F \bigcap \mathcal{C}.$$
   Therefore for convex optimization problems we do not have to distinguish between local and global solutions. A globally optimal solution for a convex problem will be simply called *optimal solution.*

2. The Karush-Kuhn-Tucker optimality conditions for a local (and thus global) optimal solution are not only necessary (under the hypothesis that some constraint qualification is satisfied), but also *sufficient.* This fact will be essentially used in Sections 2.2 and 2.3, where we describe algorithms that directly strive for points satisfying these optimality conditions.

The proofs for the above statements can be found in the standard references for convex optimization, such as [56] or [6].

We discuss in this chapter the following hierarchy of convex optimization problems.

1. Unconstrained optimization problems.

2. Equality-constrained optimization problems.

3. Constrained optimization problems.

4. Conic optimization problems.

2.2 Unconstrained optimization

2.2.1 Problem statement

Let us consider the following unconstrained convex problem with a convex objective $F : \operatorname{dom} F \subseteq \mathcal{E} \to \mathbb{R}$,

$$\min_{x \in \operatorname{dom} F} F(x). \tag{2.2}$$

Note that $\operatorname{dom} F$ might be different from $\mathbb{R}^n$, hence (2.2) might have some *implicit* constraints. For example the function

$$F(x) = -\log(x) - \log(1-x)$$

is only defined on the open interval $\operatorname{dom} F = (0,1)$. Throughout this chapter we implicity assume that (2.2) (and the subsequent principal problems, i.e. (2.18) in Section 2.3, (2.32) in Section 2.4 and the pair $(P), (D)$ in Section 2.5) are solvable.

Optimality conditions and Newton direction

Let F be continuously differentiable on its domain. We implicitly assume here that $\operatorname{dom} F$ is open (and hence full-dimensional). The point x^* is optimal for (2.2) if and only if

$$\nabla F(x^*) = 0. \tag{2.3}$$

The aim is to find a point x^* that satisfies the optimality conditions. However, these conditions are usually nonlinear in x^* and therefore it is typically difficult to find analytically a solution for (2.3). One way to compute a point that at least approximately satisfies (2.3) is the use of iterative methods that start at some initial guess x_0 and move along a search direction h that reduces the function value. Let $x \in \operatorname{dom} F$. Then we wish to find $h \in \mathcal{E}$ such that $x + h \in \operatorname{dom} F$ and

$$F(x+h) < F(x).$$

Definition 2.2.1. *Let $x \in \operatorname{dom} F$ and $h \in \mathcal{E}$. The direction h is called a* direction of descent *if and only if*

$$\nabla F(x)^T h < 0. \tag{2.4}$$

The name "direction of descent" is justified in view of the following arguments. Let $x \in \operatorname{dom} F$ and h such that $(x+h) \in \operatorname{dom} F$. If $\nabla F(x)^T h \geq 0$, then we have according to (2.1)

$$F(x+h) \geq F(x) + \nabla F(x)^T h \geq F(x),$$

which means that the function value at $(x+h)$ is higher than the one at x. This is why we have to enforce $\nabla F(x)^T h < 0$ if we want to achieve a decrease in the function value of F.

The geometric interpretation of a descent direction is that h must make an acute angle with the negative gradient of F at x. In that sense $-\nabla F(x)$ (which is known to be the direction of steepest descent with respect to the Euclidean norm) behaves as a direction of reference. However, note that (2.4) does not guarantee a decrease when going from x to $x+h$. In fact, it does not even guarantee feasibility of $x+h$. It only indicates that h is a good *direction*.

However, we see that it is possible to find a step size $\bar{\alpha}$ such that $F(x+\bar{\alpha}h) < F(x)$. Since $\nabla F(x)^T h < 0$ and F is continuously differentiable, it is possible to find a parameter $\bar{\alpha} > 0$ such that

$$(\nabla F(x+\alpha h))^T h < 0$$

holds for all $0 < \alpha < \bar{\alpha}$. On the other hand, according to the mean value theorem, we have that

$$F(x+\bar{\alpha}h) = F(x) + (\nabla F(x+\theta\bar{\alpha}h))^T h$$

for some $\theta \in (0,1)$, which means the coefficient $\alpha := \theta\bar{\alpha}$ is situated between 0 and $\bar{\alpha}$. Therefore we conclude that

$$F(x+\bar{\alpha}h) = F(x) + \underbrace{(\nabla F(x+\alpha h))^T h}_{<0} < F(x).$$

Newton's method

Newton's method is an iterative method for finding a solution to a nonlinear system of equations

$$G(x) = 0,$$

where $G : \mathcal{C}_G \subseteq \mathbb{R}^n \to \mathbb{R}^m$ is a nonlinear function in x, defined on some domain $\mathcal{C}_G$. Newton's method linearizes G

$$G(x+\Delta x) \approx G(x) + J(x)\,\Delta x,$$

where $J(x)$ denotes the Jacobian of G, and computes a step Δx such that

$$G(x) + J(x)\,\Delta x = 0.$$

We can use Newton's method to solve the system of equations that is given by the optimality conditions (2.3) in order to find an optimal solution for (2.2). We get $G(x) = \nabla F(x)$ and $J(x) = \nabla^2 F(x)$. The Newton direction is therefore the solution to the linear system

$$\nabla^2 F(x)\,\Delta x = -\nabla F(x). \tag{2.5}$$

Similarly, we can consider directly the Taylor model of F at $x \in \operatorname{dom} F$, i.e.

$$f(\Delta x) := F(x) + \nabla F(x)^T \Delta x + \frac{1}{2}\Delta x^T \nabla^2 F(x) \Delta x.$$

The optimality conditions for the quadratic function f are

$$\nabla f(\Delta x) = 0,$$

(see (2.3)), where

$$\nabla f(\Delta x) = \nabla F(x) + \nabla^2 F(x)\Delta x.$$

We see that this leads exactly to (2.5).

Properties of the Newton direction

Let F be convex and $x \in \operatorname{dom} F$ such that $\nabla F(x) \neq 0$ and $\nabla^2 F(x)$ is nonsingular. That means x is not optimal for (2.2) and the Newton direction $\Delta x \neq 0$ is defined according to (2.5). Moreover, Δx is a direction of descent, because

$$\nabla F(x)^T \Delta x = -\Delta x^T \nabla^2 F(x) \Delta x < 0.$$

We have used here convexity of F implying $\nabla^2 F(x) \succ 0$.

A very important property of the Newton direction is its affine invariance. Suppose we have a nonsingular matrix S and a constant vector t. Let us apply an affine change of variables (for example a scaling and translation of the variables) $x = Sy + t$. Then we can consider the Newton direction for the new (convex) function

$$\tilde{F}(y) := F(Sy + t).$$

It is clear that $\nabla \tilde{F}(y) = S^T \nabla F(Sy+t)$ and $\nabla^2 \tilde{F}(y) = S^T \nabla^2 F(Sy+t)\, S$. Therefore the Newton direction for $\tilde{F}$ in terms of y becomes

$$\begin{aligned}
\Delta y &= -\left[\nabla^2 \tilde{F}(y)\right]^{-1} \nabla \tilde{F}(y) \\
&= -\left[S^T \nabla^2 F(Sy+t) S\right]^{-1} S^T \nabla F(Sy+t) \\
&= -S^{-1} \nabla^2 F(x)^{-1} \nabla F(x) \\
&= S^{-1} \Delta x,
\end{aligned}$$

where Δx is the Newton direction for F in terms of x. This means that if $x = Sy+t$, then also x^+ and y^+ are related in the same affine way, i.e.

$$x^+ = x + \Delta x = Sy + t + S\Delta y = S\,(y + \Delta y) + t = S\,y^+ + t.$$

In other words, the Newton directions are independent of the choice of the coordinate system.

Closely related to the above observation is the fact that the Newton direction is also invariant under a change of the inner product. Indeed, if the inner product is changed from $\langle x, y\rangle$ to $\langle x, y\rangle_S := \langle Sx, y\rangle$, where S is a positive definite matrix, then the gradient of F at x changes from $\nabla F(x)$ to $S^{-1}\nabla F(x)$ and the Hessian changes from $\nabla^2 F(x)$ to $S^{-1}\nabla^2 F(x)$ (for a reference, see [55, Theorems 1.2.1 and 1.3.1]). As a consequence we get as the "new" Newton direction in terms of the inner product $\langle \cdot, \cdot \rangle_S$:

$$-\left[S^{-1}\nabla^2 F(x)\right]^{-1} S^{-1}\nabla F(x) = -\nabla^2 F(x)^{-1} \nabla F(x),$$

which is exactly the Newton direction Δx in terms of the inner product $\langle \cdot, \cdot \rangle$.

The following theorem provides the standard result of quadratic convergence of Newton's method in close proximity to an optimal solution x^*.

Theorem 2.2.2. *Let F be twice continuously differentiable. We assume that there exist an optimal solution x^* such that $\nabla F(x^*) = 0$ and a constant $l > 0$ such that*

$\nabla^2 F(x^*) \succeq l \cdot I$ (where I denotes the identity matrix) and that the Hessian of F is Lipschitz continuous with constant M, i.e.

$$||\nabla^2 F(x) - \nabla^2 F(y)|| \leq M||x-y||, \; \forall x, y \in \operatorname{dom} F.$$

Let x such that

$$||x - x^*|| < \frac{2l}{3M}.$$

Then the full Newton step

$$x^+ = x + \Delta x,$$

where Δx is the solution of (2.5), is feasible, i.e. $x^+ \in \operatorname{dom} F$. Moreover, the method converges quadratically:

$$||x^+ - x^*|| \leq \frac{M||x-x^*||^2}{2(l - M||x-x^*||)}.$$

Proof. e.g. [46, Theorem 1.2.5]. □

Since $||x - x^*|| < \frac{2l}{3M} \left(< \frac{l}{M}\right)$, it follows $3M||x-x^*||^2 < 2l||x-x^*||$ (by multiplying both sides with the positive number $3M||x-x^*||$), which is equivalent to

$$M||x-x^*||^2 < 2 \underbrace{(l - M||x-x^*||)}_{>0} ||x-x^*||.$$

Dividing both sides of the inequality by the positive number $2(l - M||x-x^*||)$ yields

$$\frac{M||x-x^*||^2}{2(l - M||x-x^*||)} < ||x-x^*||.$$

In view of Theorem 2.2.2 we see that as soon as x is close enough to an optimal solution x^*, then the method converges monotonically towards x^*. Moreover, the method converges quadratically, since

$$||x^+ - x^*|| \leq \frac{M||x-x^*||^2}{2(l - M||x-x^*||)} < \frac{3M}{2l}||x-x^*||^2.$$

That means if we have in the current iteration an optimality gap of $\epsilon = ||x - x^*||$, then the optimality gap in the the next iteration will be $\mathcal{O}(\epsilon^2)$.

However, Theorem 2.2.2 can only be applied in a rather small neighborhood around the optimal solution. On the other hand, since the Newton direction Δx is a direction of descent, by introducing a step size parameter α that is sufficiently small, we can always guarantee an actual decrease in the function value when going from x to $x + \alpha\Delta x$.

Another drawback of Theorem 2.2.2 is that the region of quadratic convergence seems to depend on the choice of the norm $||\cdot||$ (or underlying inner product $\langle\cdot,\cdot\rangle$). However, we have shown before that the Newton directions are invariant under a change of the inner product, and therefore the region of quadratic convergence should *not* depend on the chosen inner product (for a thorough discussion we refer the reader to [46, Section 4.1.2]).

Let us summarize the damped Newton method for minimizing a convex function F (Algorithm 1).

At this moment there are some issues that are still unclear for Algorithm 1.

Algorithm 1 Standard damped Newton method for minimizing a convex function

```
repeat
    1) solve (2.5) to get the Newton direction $\Delta x$,
    2) update $x^+ = x + \alpha\Delta x$, where $\alpha$ is a suitable stepsize such that $F$ decreases,
until stopping criterion is met
```

1. What is a good stopping criterion that guarantees proximity to an optimal solution?

2. How to choose the step size parameter α so that the iterates form a sequence that converges to an optimal solution?

3. Can we find a closed form description of the region of quadratic convergence?

We answer these three questions in Section 2.2.3.

2.2.2 Self-concordant functions

We present now a family of convex functions that are particularly well-suited for Newton's method. Recall that the third assumption on F in Theorem 2.2.2 requires a uniform absolute bound on the variation of the Hessian of F. In fact, this assumption becomes the defining property of self-concordant functions, and it can be viewed as a *relative* bound of the third derivative of F in terms of the second derivative of F, at any point $x \in \operatorname{dom} F$, in any direction $h \in \mathbb{R}^n$. For a detailed discussion, read [46, Section 4.1.2].

Definition 2.2.3. *A closed convex function $F \in \mathcal{C}^3$ (three times continuously differentiable) with open domain $\mathcal{C}$ is called* self-concordant *if*

$$|D^3F(x)[h,h,h]| \leq 2D^2F(x)[h,h]^{3/2}, \tag{2.6}$$

for all $x \in \operatorname{dom} F$ and for all $h \in \mathbb{R}^n$.
A self-concordant function F is called nondegenerate *if its Hessian $\nabla^2 F$ is nonsingular for all $x \in \operatorname{dom} F$.*

Theorem 2.2.4. *If F is self-concordant and its domain does not contain a straight line, then F is nondegenerate.*

Assumption:

In the rest of this chapter when we speak of self-concordant functions we implicitly assume that they are nondegenerate.

The assumption of nondegeneracy guarantees that the Newton directions are defined everywhere. Moreover, since the Hessian is nonsingular and positive semidefinite it must be positive definite and therefore F is strictly convex. It follows that if x^* is a minimizer of the nondegenerate self-concordant function F, then it is in fact unique.

Examples of self-concordant functions

- Affine functions $F(x) = a^T x + b$ are self-concordant (since their second and third derivatives are constant and equal to 0), but not nondegenerate,
- Convex quadratic functions $F(x) = x^T Ax + a^T x + \alpha$, where $A \succeq 0$, are self-concordant because $D^2F(x) = A \succeq 0, D^3F(x) = 0$. If $A \succ 0$, then F is even nondegenerate.
- $F(x) = -\log(x)$, because $D^2F(x)[h,h] = \frac{h^2}{x^2}$, $D^3F(x)[h,h,h] = -2\frac{h^3}{x^3}$.

Operations that preserve self-concordance

Similar to the class of convex functions, we have operations that preserve the property of self-concordance (compare to Section 2.1.2). The operations listed below can be found for example in [46]. Since a self-concordant function F has to be differentiable, we implicitly always assume that its domain is open.

1. **Weighted sum.**
Let $F_i : \mathcal{C}_i \subseteq \mathcal{E} \to \mathbb{R}, i = 1, \ldots, m$ be self-concordant functions and $\alpha_i \geq 1, i = 1, \ldots, m$. Then
$$F(x) = \sum_{i=1}^{m} \alpha_i F_i(x)$$
is self-concordant on $\operatorname{dom} F = \bigcap_{i=1}^{n} \mathcal{C}_i$. Note that the coefficients have to be greater than or equal to 1, as opposed to 0 in Section 2.1.2 (Proof, see [52, Proposition 2.1.1(ii)]).

2. **Weighted sum of separable functions.**
Let $F_i : \mathcal{C}_i \subseteq \mathcal{E}_i \to \mathbb{R}, i = 1, \ldots, m$ be self-concordant functions and $\alpha_i \geq 1, i = 1, \ldots, m$. Then
$$F(x) = \sum_{i=1}^{m} \alpha_i F_i(x_i)$$
is self-concordant on $\mathcal{C}_1 \times \ldots \times \mathcal{C}_m$ (Proof, see [52, Proposition 2.1.1(iii)]).

3. **Composition with affine function.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be self-concordant, $\mathcal{A} : \mathbb{R}^p \to \mathbb{R}^n$ such that $\mathcal{A}(y) = By + c$ for $B \in \mathbb{R}^{n,p}$ and $c \in \mathbb{R}^n$. Assume $\mathcal{A}(\mathbb{R}^p) \bigcap \mathcal{C} \neq \emptyset$. Define
$$\mathcal{C}^+ = \mathcal{A}^{-1}(\mathcal{C}) = \{y \in \mathbb{R}^p : \mathcal{A}(y) \in \mathcal{C}\} \subseteq \mathbb{R}^p.$$
Then $\tilde{F} : \mathcal{C}^+ \to \mathbb{R}$ defined as
$$\tilde{F}(y) = F(\mathcal{A}(y))$$
is self-concordant on $\operatorname{dom} \tilde{F} = \mathcal{C}^+$ (Proof, see [52, Proposition 2.1.1(i)]).

4. **Restriction to affine subspace.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be self-concordant and $\mathcal{L} = \{x \in \mathbb{R}^n : Ax = b\}$ an affine subspace, where $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $m < n$. Then the restriction

of F to $\mathcal{L}$, which we denote by $F|_{\mathcal{L}}$, is self-concordant on its domain. We define $\operatorname{dom} F|_{\mathcal{L}} = \operatorname{ri}(\mathcal{C} \bigcap \mathcal{L})$, where $\operatorname{ri}(\cdot)$ denotes the relative interior of a set. Note that we consider $\operatorname{dom} F|_{\mathcal{L}}$ as a full-dimensional object in the lower-dimensional space $\mathbb{R}^{n-m}$, embedded in $\mathbb{R}^n$. This ensures that $\operatorname{dom} F|_{\mathcal{L}}$ is an open set which we need in order to make sure that the derivatives of $F|_{\mathcal{L}}$ are well defined. We discuss this operation of restricting a self-concordant function in more detail below.

5. **Partial minimization.**
Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$, such that $(x, y) \mapsto F(x, y)$, be self-concordant and bounded from below. We assume that $\mathcal{C}$ does not contain a straight line. Then the partial minimization of F with respect to y

$$G(x) = \inf_{y \in \mathcal{Q}(x)} F(x, y),$$

where $\mathcal{Q}(x) = \{y : (x, y) \in \mathcal{C}\}$, is self-concordant on $\operatorname{dom} G = \{x : \mathcal{Q}(x) \neq \emptyset\}$ (Proof, see [50, Theorem 3]).

Note that 4. is in fact a consequence of 3. In order to see that, let us define $p = n - m$. Let $B \in \mathbb{R}^{n,p}$ be any matrix such that such that $\operatorname{range}(B) = \operatorname{null}(A)$ and let $\bar{x}$ be any particular solution to the linear system $Ax = b$, i.e. $\bar{x} \in \mathcal{L}$. Define $c = \bar{x}$. Then we can parametrize $\mathcal{L}$ in the following way:

$$\mathcal{L} = \{x \in \mathbb{R}^n : Ax = b\} = \{By + c : y \in \mathbb{R}^p\} \subseteq \mathbb{R}^n.$$

The parametrization of $\mathcal{L}$ corresponds to the elimination of $m = n - p$ variables x_i. Using this definition, we get that

$$\mathcal{C}^+ = \{y \in \mathbb{R}^p : By + c \in \mathcal{C}\} \subseteq \mathbb{R}^p.$$

In view of 3. we have that $\tilde{F} : \mathcal{C}^+ \to \mathbb{R}$, defined as

$$\tilde{F}(y) = F(By + c),$$

is self-concordant on $\mathcal{C}^+$, which is open because $\mathcal{C}$ is open.

Let us give an interpretation of the set $\mathcal{C}^+$ in our case. For $y \in \mathbb{R}^p$ we define the point $x_y = By + c \in \mathbb{R}^n$. Then $y \in \mathcal{C}^+$ if and only if $x_y \in \mathcal{C}$. The latter condition is the same as $x_y - c = By$, or equivalently $x_y - c \in \operatorname{range}(B)$. By assumption, we have $\operatorname{range}(B) = \operatorname{null}(A)$, so we get $x_y - c \in \operatorname{null}(A)$ which means $A(x_y - c) = 0$. Since $\bar{x} = c \in \mathcal{L}$, it follows

$$A(x_y - c) = Ax_y - Ac = Ax_y - b = 0,$$

or $Ax_y = b$. In other words, $x_y \in \mathcal{L}$. From above we have additionally that $x_y \in \mathcal{C}$. It follows that $y \in \mathcal{C}^+$ if and only if $x_y = By + c \in \mathcal{C} \bigcap \mathcal{L}$.

Note that we have to be careful when speaking about restrictions of self-concordant functions to affine subspaces. Even though $\mathcal{C}^+$ is open (in $\mathbb{R}^m$) the set $\mathcal{C} \bigcap \mathcal{L} \subseteq \mathbb{R}^n$ is *not* open (neither is its relative interior $\operatorname{ri}(\mathcal{C} \bigcap \mathcal{L})$). Strictly speaking functions defined on $\operatorname{ri}(\mathcal{C} \bigcap \mathcal{L})$ cannot be differentiable, as differentiability implicitly assumes that the domain is open.

Therefore, when we speak of the restriction of a self-concordant function F to the affine subspace $\mathcal{L} = \{x : Ax = b\}$, we understand it in the following way: find a matrix B such that $\text{range}(B) = \text{null}(A)$ and $c \in \mathcal{L}$. The restriction $F|_{\mathcal{L}} : \mathcal{C}^+ \to \mathbb{R}$ is defined as

$$F|_{\mathcal{L}}(y) = F(By + c)$$

on $\text{dom}\, F|_{\mathcal{L}} = \mathcal{C}^+ := \{y : By + c \in \mathcal{C}\} \subseteq \mathbb{R}^p$.

Example 2.2.5. *Let* $F : \mathcal{C} \subseteq \mathbb{R}^3 \to \mathbb{R}$ *be self-concordant with* $\mathcal{C} = \text{int}\{x \in \mathbb{R}^3 : ||x||_2 \leq 1\}$. *Let* $\mathcal{L} = \{x \in \mathbb{R}^3 : x_1 + x_2 + x_3 = 1\}$. *We see that* $\mathcal{C} \bigcap \mathcal{L}$ *is the unit ball intersected with some hyperplane. In the above notation we have that* $A = [1, 1, 1]$ *and* $b = 1$. *Moreover,* $\text{null}(A) = \{x = (x_1, x_2, x_3) : Ax = x_1 + x_2 + x_3 = 0\} \subseteq \mathbb{R}^3$.

If we define now

$$B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{bmatrix}, \qquad c = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

we can check that $\text{range}(B) = \{x \in \mathbb{R}^3 : x = By = (y_1, y_1, -y_1 - y_2),$ *for some* $y \in \mathbb{R}^2\}$. *We see immediately that* $\text{null}(A) = \text{range}(B)$. *Furthermore, we can check that* $c \in \mathcal{L}$.

By virtue of the above arguments, we have that the restriction of F *to* $\mathcal{L}$ *is in fact given by* $F(\mathcal{A}(y))$, *where* $\mathcal{A}(y) = By + c$, *that is*

$$F|_{\mathcal{L}}(y) = F(y_1, y_2, 1 - y_1 - y_2)$$

with $\text{dom}\, F|_{\mathcal{L}} = \{y \in \mathbb{R}^2 : \mathcal{A}(y) \in \mathcal{C}\}$.

Properties of self-concordant functions

For the rest of this section, when we say that F is self-concordant, we mean that $F : \mathcal{C} \subseteq \mathcal{E} \to \mathbb{R}$ is a nondegenerate self-concordant function with open domain $\mathcal{C}$.

The following theorem states that every self-concordant function is in fact a *barrier* for its domain.

Theorem 2.2.6. *Let* F *be self-concordant. For any sequence* $\{x_k\} \subset \mathcal{C}$ *converging towards* $\text{cl}(\mathcal{C}) \backslash \mathcal{C}$ *we have*

$$F(x_k) \to +\infty.$$

Proof. [46, Theorem 4.1.4]. □

Any (nondegenerate) self-concordant function F induces a family of *intrinsic inner products* in $\mathcal{E}$: for any $x \in \mathcal{C}, h_1, h_2 \in \mathcal{E}$ we define

$$\langle h_1, h_2 \rangle_x := \left\langle \nabla^2 F(x) h_1, h_2 \right\rangle.$$

Analogously, we can define the *intrinsic inner product* in the dual space $\mathcal{E}^*$: for $x \in \mathcal{C}$ and $g_1, g_2 \in \mathcal{E}^*$ we define

$$\langle g_1, g_2 \rangle_x := \left\langle g_1, \nabla^2 F(x)^{-1} g_2 \right\rangle.$$

Using the intrinsic inner products, we can define a so-called *local norm* in $\mathcal{E}$ and $\mathcal{E}^*$ with respect to $x \in \mathcal{C}$.

Definition 2.2.7. *Let F be self-concordant, $x \in \mathcal{C}$. We define the* local norm *of $h \in \mathcal{E}$ as*

$$||h||_x := {\langle h, h \rangle_x}^{1/2}, \ h \in \mathcal{E}$$

and the (dual) local norm *of $g \in \mathcal{E}^*$ as*

$$||g||_x^* := {\langle g, g \rangle_x}^{1/2}, \ g \in \mathcal{E}^*.$$

One can verify that $|| \cdot ||_x^*$ is indeed the norm which is dual to $|| \cdot ||_x$, i.e.

$$||g||_x^* = \max_{||h||_x \leq 1} \langle g, h \rangle.$$

As $|| \cdot ||_x$ and $|| \cdot ||_x^*$ are dual to each other, we automatically have the Hölder inequality with respect to the reference point $x \in \mathcal{C}$, for any $h \in \mathcal{E}$ and any $g \in \mathcal{E}^*$, i.e.

$$|\langle g, h \rangle| \leq ||g||_x^* \cdot ||h||_x.$$

If we use the local norms to measure the size of a point $h \in \mathcal{E}$ (or $g \in \mathcal{E}^*$) then these values depend obviously on the choice of the reference point $x \in \mathcal{C}$. For the same reason the neighborhoods around a fixed point $\bar{h} \in \mathcal{E}$ (with respect to the local norm) are dependent on the reference point $x \in \mathcal{C}$. If we consider the neighborhood around $x \in \mathcal{C}$ measured with the local norm with respect to the same point x, we get the so-called Dikin ellipsoids. They are important objects to describe the topology of F, as we will see later.

Definition 2.2.8. *Let F be self-concordant. For given $x \in \mathcal{C}$ the* Dikin ellipsoid *with radius $r > 0$ is defined as*

$$D(x, r) = \{y : ||y - x||_x \leq r\}.$$

We denote the interior of the Dikin ellipsoid by $D_0(x, r)$, i.e.

$$D_0(x, r) = \{y : ||y - x||_x < r\}.$$

The following result is essential for the analysis of the behavior of self-concordant functions inside $\mathcal{C}$. It says that the Dikin ellipsoid is always contained in $\mathcal{C}$.

Lemma 2.2.9. *Let F be self-concordant. For any $x \in \mathcal{C}$ and $0 < r < 1$ we have*

$$D(x, r) \subseteq \mathcal{C} = \operatorname{dom} F.$$

Proof. [46, Theorem 4.1.5]. □

For the next theorem, let us introduce some piece of notation. Let $A \in \mathcal{S}^n$, $B \in \mathcal{S}^n$. Then we denote

$$A \succeq B$$

if and only if $A - B \succeq 0$, that is the difference $A - B$ is positive semidefinite. Analogously, we write $A \succ B$ if and only if $A - B \succ 0$.

Inside the Dikin ellipsoid self-concordant functions are well-behaved. The following theorem states that the Hessian of a self-concordant function (and hence the local norm) does not vary too much inside $D(x, r)$.

Theorem 2.2.10. *Let F be self-concordant, $x \in \mathcal{C}$, $y \in D(x,r)$, $r < 1$ and $h \in \mathcal{E}$. Then*

$$(1-r) \cdot ||h||_x \leq ||h||_y \leq \frac{1}{(1-r)} \cdot ||h||_x,$$

or equivalently

$$(1-r)^2 \nabla^2 F(x) \preceq \nabla^2 F(y) \preceq \frac{1}{(1-r)^2} \nabla^2 F(x).$$

Proof. [46, Theorem 4.1.6]. □

We have the following two bounds on the variation of the gradient of F inside $D(x,r)$.

Lemma 2.2.11. *Let F be self-concordant, $x \in \mathcal{C}$, $y \in D(x,r)$, $r < 1$. Then it holds*

$$||\nabla F(y) - \nabla F(x) - \nabla^2 F(x)(y-x)||_x^* \leq \frac{r^2}{1-r}.$$

Proof. [49, Lemma 1]. □

For the following theorem we need to introduce the local norm of a linear operator.

Definition 2.2.12. *Let $M : \mathcal{E} \to \mathcal{E}^*$ be a linear operator, F self-concordant and $x \in \mathcal{C}$. Then we define the local norm of M as*

$$||M||_x := \sup_{h:||h||_x \leq 1} ||M\,h||_x^*.$$

Theorem 2.2.13. *Let F be self-concordant, $x \in \mathcal{C}$, $y \in D(x,r)$, $r < 1$. Then it holds*

$$||\nabla F(y) - \nabla F(x)||_x^* \leq \frac{r}{1-r}.$$

Proof. Let us define $y(\theta) = x + \theta(y-x)$ for $\theta \in [0,1]$. Then we have

$$\nabla F(y) - \nabla F(x) = \int_0^1 \nabla^2 F(y(\theta))\,(y-x)\,d\theta.$$

Using the above representation and subadditivity of norms, we get

$$\begin{aligned} ||\nabla F(y) - \nabla F(x)||_x^* &= ||\int_0^1 \nabla^2 F(y(\theta))\,(y-x) d\theta||_x^* \\ &\leq \int_0^1 ||\nabla^2 F(y(\theta)) d\theta\,(y-x)||_x^*. \end{aligned}$$

In other words, the norm of the integral is less than or equal to the integral of the norm. A formal proof of the inequality that we have used above, can be found for example in [55, Theorem 1.5.4]. In view of Definition 2.2.12, and using the fact that $y \in D(x,r)$ means $||y-x||_x \le r < 1$, we obtain

$$\begin{aligned}\int_0^1 ||\nabla^2 F(y(\theta))d\theta\,(y-x)||_x^* &\le \sup_{y:||y-x||_x\le 1}\int_0^1 ||\nabla^2 F(y(\theta))d\theta\,(y-x)||_x^* \\ &= \int_0^1 \sup_{y:||y-x||_x\le 1} ||\nabla^2 F(y(\theta))d\theta\,(y-x)||_x^* \\ &= \int_0^1 ||\nabla^2 F(y(\theta))||_x d\theta \cdot ||y-x||_x.\end{aligned}$$

Further, since $\theta \in [0,1]$, we have that $y(\theta) \in D(x,\bar{r})$, where $\bar{r} = \theta r \le r < 1$. Using Theorem 2.2.10, it follows

$$\nabla^2 F(y(\theta)) \preceq \frac{1}{(1-\theta r)^2}\nabla^2 F(x),$$

which implies

$$||\nabla^2 F(y(\theta))||_x \le \frac{1}{(1-\theta r)^2}||\nabla^2 F(x)||_x.$$

But since

$$||\nabla^2 F(x)||_x = \max_{||h||_x\le 1} ||\nabla^2 F(x)\; h||_x^* = \max_{||h||_x \le 1} ||h||_x = 1,$$

we conclude

$$\begin{aligned}||\nabla F(y) - \nabla F(x)||_x^* &\le \int_0^1 \frac{r}{(1-\theta r)^2}d\theta \\ &= \int_0^r \frac{r}{(1-t)^2}\,\frac{1}{r}dt \\ &= \frac{1}{r}\left[\frac{r}{1-t}\right]_{t=0}^r = \frac{1}{r}\left(\frac{r}{1-r} - r\right) \\ &= \frac{1}{r}\,\frac{r^2}{1-r} = \frac{r}{1-r}.\end{aligned}$$

□

Let us define the following convex functions that will be useful in the analysis of self-concordant functions.

$$\omega(t_1) := t_1 - \log(1+t_1),\ t_1 > -1, \qquad \omega_*(t_2) := -t_2 - \log(1-t_2),\ t_2 < 1.$$

Lemma 2.2.14. *We have for* $0 \le t_1$ *and* $0 \le t_2 < 1$

$$\omega'(t_1) = \frac{t_1}{1+t_1}, \qquad \omega_*'(t_2) = \frac{t_2}{1-t_2}, \tag{2.7}$$

$$\omega_*'(\omega'(t_1)) = t_1, \qquad \omega'(\omega_*'(t_2)) = t_2, \tag{2.8}$$

$$\omega_*(t_2) = t_2\omega_*'(t_2) - \omega(\omega_*'(t_2)), \qquad \omega(t_1) = t_1\omega'(t_1) - \omega_*(\omega'(t_1)). \tag{2.9}$$

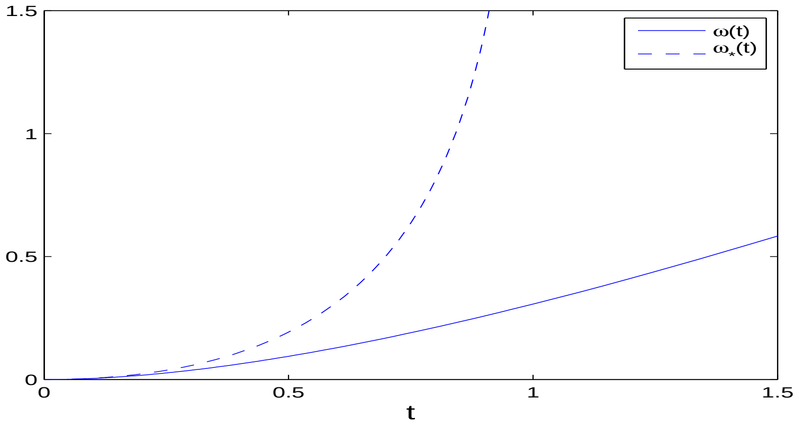


Figure 2.1: solid: $\omega(t)$, dashed: $\omega_*(t)$.

Proof. [46, Lemma 4.1.4]. □

Note that ω and ω_* are in fact conjugate to each other (see [46, Lemma 4.1.4]), that is

$$\begin{aligned}\omega_*(t_2) &= \sup_{t_1 \geq 1}\{t_1 t_2 - \omega(t_1)\},\\ \omega(t_1) &= \sup_{t_2 < 1}\{t_1 t_2 - \omega_*(t_2)\}.\end{aligned}$$

Moreover, in view of (2.8) we have that ω' and ω'_* are inverse to each other. We have for $0 \leq t_1, t_2$

$$\omega(t_1) + \omega(t_2) \leq \omega(t_1 + t_2), \tag{2.10}$$

Indeed

$$\begin{aligned}\omega(t_1) + \omega(t_2) &= t_t - \log(1 + t_1) + t_2 - \log(1 + t_2)\\ &= t_1 + t_2 - \log((1 + t_1)(1 + t_2))\\ &= t_1 + t_2 - \log(1 + t_1 + t_2 + \underbrace{t_1 t_2}_{\geq 0})\\ &\leq t_1 + t_2 - \log(1 + t_1 + t_2)\\ &= \omega(t_1 + t_2).\end{aligned}$$

With the same arguments we get for $0 \leq t_1, t_2$ such that $t_1 + t_2 < 1$

$$\omega_*(t_1) + \omega_*(t_2) \leq \omega_*(t_1 + t_2). \tag{2.11}$$

The following two theorems give convex lower and upper bounds on both the function value of F and the variation of the gradient of F around a point $x \in \mathcal{C}$.

Theorem 2.2.15. *Let F be self-concordant, $x \in \mathcal{C}$ and $y \in \mathcal{C}$. Denote $r = ||y - x||_x$. Then*

$$F(y) \geq F(x) + \langle \nabla F(x), y - x \rangle + \omega(r), \tag{2.12}$$

$$\langle \nabla F(y) - \nabla F(x), y - x \rangle \geq \frac{r^2}{1 + r}. \tag{2.13}$$

Reversely, let (2.12) or (2.13) be true for any $x \in \mathcal{C}$ and $y \in \mathcal{C}$. Then F is self-concordant.

Proof. [46, Theorem 4.1.7 and 4.1.9]. □

Theorem 2.2.16. *Let F be self-concordant, $x \in \mathcal{C}$ and $y \in D(x, r)$, $r < 1$. Then*

$$F(y) \leq F(x) + \langle \nabla F(x), y - x \rangle + \omega_*(r), \tag{2.14}$$

$$\langle \nabla F(y) - \nabla F(x), y - x \rangle \leq \frac{r^2}{1 - r}. \tag{2.15}$$

Reversely, let (2.14) or (2.15) be true for any $x \in \mathcal{C}$ and $y \in D(x, r)$, $r < 1$. Then F is self-concordant.

Proof. [46, Theorem 4.1.8 and 4.1.9]. □

2.2.3 Newton's method for minimizing self-concordant functions

We have mentioned above that the class of self-concordant functions is particularly well-suited for Newton's method. We are going to demonstrate that now. An important object that gives us a lot of information about F is the so-called Newton decrement.

Definition 2.2.17. *Let F be self-concordant and $x \in \mathcal{C}$. We define the* Newton decrement *at x as*

$$\delta_x = \langle \nabla F(x), \nabla^2 F(x)^{-1} \nabla F(x) \rangle^{1/2}.$$

In view of the definition of the local norms (see Definition 2.2.7), we have

$$\delta_x = ||\nabla F(x)||_x^*.$$

On the other hand we can also write

$$\begin{aligned} \delta_x &= \left(\nabla F(x)^T \underbrace{\nabla^2 F(x)^{-1} \nabla F(x)}_{= -\Delta x} \right)^{1/2} \\ &= \left(-\nabla F(x)^T \Delta x \right)^{1/2}, \end{aligned}$$

where Δx denotes the Newton direction. Finally, we can introduce the identity $I = (\nabla^2 F(x))^{-1}\nabla^2 F(x)$ in the last expression. We get

$$\begin{aligned}\delta_x &= \Bigg(\underbrace{-\nabla F(x)^T \nabla^2 F(x)^{-1}}_{=\Delta x^T} \nabla^2 F(x)\Delta x \Bigg)^{1/2} \\ &= \left(\Delta x^T \nabla^2 F(x)\Delta x\right)^{1/2} \\ &= ||\Delta x||_x.\end{aligned}$$

That means we have three different representations of the Newton decrement δ_x, that is

$$\delta_x = ||\nabla F(x)||_x^* = \left(-\nabla F(x)^T \Delta x\right)^{1/2} = ||\Delta x||_x.$$

If we want to find a point x that is close (in some sense) to the optimal solution x^*, there are several ways of measuring this proximity: we could compare the objective values $F(x)$ and $F(x^*)$ and if the difference $F(x) - F(x^*)$ is sufficiently small, we accept x as an approximation for x^*. Another way would be to look at the (local) norm of the error term $e = x - x^*$, i.e. $||x - x^*||_x$ and accept x if the norm is small enough.

However, we see that both measures involve the knowledge of the (unknown) optimal solution x^* (or its function value). The following theorem shows how the readily computable Newton decrement can be used to bound these optimality measures.

Theorem 2.2.18. *Let F be self-concordant and $x \in \mathcal{C}$ such that $\delta_x < 1$. Then*

$$\omega(\delta_x) \leq F(x) - F(x^*) \leq \omega_*(\delta_x), \tag{2.16}$$

$$\omega'(\delta_x) \leq ||x - x^*||_x \leq \omega_*'(\delta_x). \tag{2.17}$$

Proof. [46, Theorem 4.1.13]. □

Note that the lower bounds in (2.16) and (2.17) are valid even if $\delta_x \geq 1$.

The *upper bounds* provided by Theorem 2.2.18 are particularly useful. Unfortunately, an upper bounding in terms of the Newton decrement is only possible if δ_x is small. If $\delta_x \geq 1$, there is no immediate way to bound these distance measures. We will address this issue at the end of this section.

If the Newton decrement is small (which means we are close to x^*), we have quadratic convergence of Newton's method in terms of δ_x.

Theorem 2.2.19. *Let F be self-concordant and $x \in \mathcal{C}$ such that $\delta_x < 1$. Then the full Newton step*

$$x^+ = x + \Delta x,$$

where Δx is the solution of (2.5), is feasible and we have

$$\delta_{x^+} \leq \left(\frac{\delta_x}{1-\delta_x}\right)^2.$$

Proof. [46, Theorem 4.1.14]. □

The quadratic convergence of Newton's method can also be phrased in terms of the distance to an optimal solution x^*. We have the following theorem.

Theorem 2.2.20. *Let F be self-concordant and $x \in \mathcal{C}$ with $x^* \in D_0(x,1)$. Then the full Newton step*

$$x^+ = x + \Delta x,$$

where Δx is the solution of (2.5), is feasible and we have

$$||x^+ - x^*||_x \leq \frac{||x - x^*||_x^2}{1 - ||x - x^*||_x}.$$

Proof. [55, Theorem 2.2.3]. □

Note the similarity to the standard result on quadratic convergence (Theorem 2.2.2). The difference to Theorem 2.2.2 is that all distances are now phrased in terms of the local norm $||\cdot||_x$ with respect to the current iterate x.

Let us phrase the bound on $||x^+ - x^*||_x$ from Theorem 2.2.20 in terms of the Newton decrement δ_x.

Theorem 2.2.21. *Let F be self-concordant and $x \in \mathcal{C}$ such that $\delta_x < \frac{1}{2}$. Then we have*

$$||x^+ - x^*||_x \leq \frac{\delta_x^2}{2\delta_x^2 - 3\delta_x + 1}.$$

Proof. For $0 < \delta_x < \frac{1}{2}$ we have $\frac{\delta_x}{1-\delta_x} < 1$. In view of the right-hand side inequality of (2.17) it follows then

$$||x - x^*||_x \leq \frac{\delta_x}{1-\delta_x} < 1,$$

which means $x^* \in D_0(x,1)$. Therefore we can apply Theorem 2.2.20 and get

$$||x^+ - x^*||_x \leq \frac{||x - x^*||_x^2}{1 - ||x - x^*||_x}.$$

In view of the right-hand side inequality of (2.17) we have

$$\begin{aligned}
\frac{||x - x^*||_x^2}{1 - ||x - x^*||_x} &\leq \frac{(\omega_*(\delta_x))^2}{1 - \omega_*(\delta_x)} \\
&= \frac{\left(\frac{\delta_x}{1-\delta_x}\right)^2}{1 - \frac{\delta_x}{1-\delta_x}} = \frac{\delta_x^2}{(1-\delta_x)^2}\frac{1-\delta_x}{1-2\delta_x} \\
&= \frac{\delta_x^2}{(1-\delta_x)(1-2\delta_x)} \\
&= \frac{\delta_x^2}{1 - 3\delta_x + 2\delta_x^2}.
\end{aligned}$$

□

Renegar ([55, Theorem 2.2.5]) has established a similar bound on the error of the new iterate x^+ with respect to the local norm at the current iterate x. The bound is the following. If $\delta_x \leq \frac{1}{4}$, then

$$||x^+ - x^*||_x \leq \frac{3\delta_x^2}{(1-\delta_x)^3}.$$

Note that the upper bound from Theorem 2.2.21 is tighter than the one provided by Renegar. This can be seen in Figure 2.2, where we have plotted both upper bounds as functions of δ_x. Moreover, Theorem 2.2.21 is valid for $\delta_x < \frac{1}{2}$, while Renegar requires $\delta_x \leq \frac{1}{4}$.

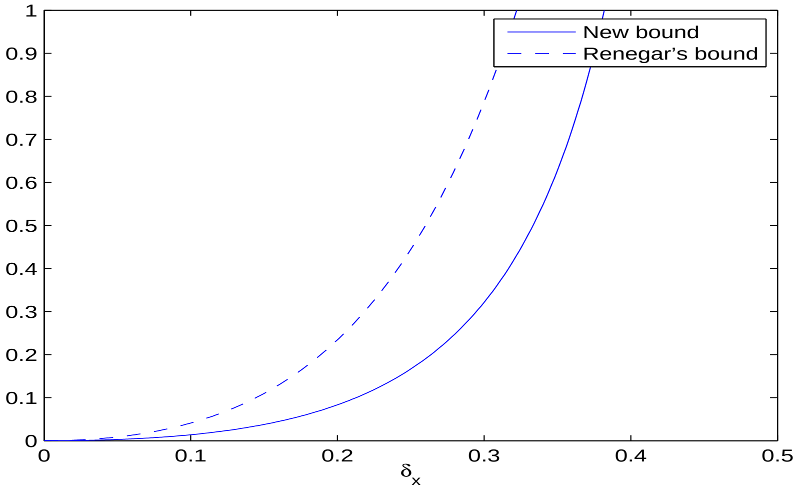


Figure 2.2: Two bounds on $||x^+ - x^*||_x$. Solid: new bound $\frac{\delta_x^2}{1-3\delta_x+2\delta_x^2}$, dashed: Renegar's bound $\frac{3\delta_x^2}{(1-\delta_x)^3}$.

Theorem 2.2.20 and Theorem 2.2.21 provide bounds on the quantity $||x^+ - x^*||_x$, which is the error at the *new* iterate $x^+ = x + \Delta x$, measured with respect to the local norm in terms of the *old* iterate x. The following result provides a bound on the distance of x^* to the new point x^+ with respect to the ”correct” local norm at x^+.

Corollary 2.2.22. *Let F be self-concordant and $x \in \mathcal{C}$ such that $\delta_x < \frac{1}{2}$. Then we have*

$$||x^+ - x^*||_{x^+} \leq \frac{\delta_x^2}{(1-\delta_x)^2(1-2\delta_x)}.$$

Proof. The inequality is an immediate consequence of Theorem 2.2.21 and the fact that $||x^+ - x||_x = ||\Delta x||_x = \delta_x < \frac{1}{2}$ implies in accordance with Theorem 2.2.10

$$||x^+ - x^*||_x \geq (1-\delta_x)||x^+ - x^*||_{x^+}.$$

□

As we have mentioned before, the Newton direction Δx is a direction of descent (unless we are already at some optimal solution), and we have argued that it should be possible to find a step size α such that we can globalize the method: from any starting point we can apply several damped Newton steps, thus decrease the function value and eventually reach the region of quadratic convergence. Once in that region, few full Newton steps suffice to compute an arbitrarily good approximation for an optimal solution. The following theorem gives an explicit description of such a step size.

Theorem 2.2.23. *Let F be self-concordant and $x \in \mathcal{C}$. We define the new iterate*

$$x^+ = x + \frac{1}{1+\delta_x} \cdot \Delta x.$$

Then $x^+ \in \mathcal{C}$ and

$$F(x^+) \leq F(x) - \omega(\delta_x).$$

Proof. [46, Theorem 4.1.12]. □

We are coming back now to the three questions we have posed right after formulating the basic version of Newton's method (Algorithm 1).

Theorem 2.2.19 and Theorem 2.2.20 settle the third question related to the description of the region of quadratic convergence of Newton's method. Theorem 2.2.19 phrases the quadratic convergence in terms of the Newton decrement δ_x, which is a quantity that we can observe. Theorem 2.2.20, on the other hand, illustrates quadratic convergence of the method in terms of the errors $||x - x^*||_x$, which might be of higher interest (as they describe the actual distance to the optimum) but are not directly accessible.

The second question, related to a suitable step size, is answered by Theorem 2.2.23 and again Theorem 2.2.19 (or Theorem 2.2.20). As long as the Newton decrement is large, which means we are far from an optimal solution (in view of the left-hand side inequalities of Theorem 2.2.18), we can also achieve a relatively large improvement in terms of the function value (see Figure 2.1). If δ_x is small (that is, $\delta_x < 1$ for Theorem 2.2.19 and $\delta_x < \frac{1}{2}$ for Theorem 2.2.20) we can do full Newton steps, i.e. we can choose as step size parameter $\alpha = 1$.

To answer the first question concerning the stopping criterion, we refer to the right-hand side inequalities of Theorem 2.2.18. If we desire an accuracy in terms of the objective value of, say $\epsilon_1 > 0$, then we can stop as soon as we have found a point $x \in \mathcal{C}$ such that $\delta_x \leq \omega_*^{-1}(\epsilon_1)$. Analogously, if we wish to find a point such that $||x - x^*||_x \leq \epsilon_2$, for some $\epsilon_2 > 0$, then we can stop the algorithm as soon as we have encountered a point $x \in \mathcal{C}$ such that $\delta_x \leq (\omega_*')^{-1}(\epsilon_2) = \omega'(\epsilon_2)$. In view of Theorem 2.2.19 and Theorem 2.2.23, it becomes clear how a globalized Newton method for minimizing a self-concordant function should look like. It consists of 2 phases.

1. **Damped phase.**
   As long as the Newton decrement is large, we reduce the function value by a nontrivial amount according to Theorem 2.2.23.

2. **Quadratically convergent phase.**
   If the Newton decrement is sufficiently small, we are in the region of quadratic convergence and we can do full Newton steps according to Theorem 2.2.19.

We only have to decide when to switch from phase 1 to phase 2. We would like to enter the second phase as soon as possible to profit from the rapid convergence to the optimal solution.

Theorem 2.2.19 can be applied as soon as $\delta_x < 1$, but this alone does not guarantee a decrease of the Newton decrement δ_x. Additionally, we need that $\delta_{x^+} < \delta_x$ to ensure monotonic (and fast) convergence. In view of Theorem 2.2.19 the latter inequality is guaranteed if

$$\left(\frac{\delta_x}{1-\delta_x}\right)^2 < \delta_x,$$

which is true whenever $\delta_x^2 < (1-\delta_x)^2\delta_x$. Since we need that $\delta_x < 1$, we get that a decrease $\delta_{x^+} < \delta_x$ is ensured as soon as $0 < \delta_x < \frac{3-\sqrt{5}}{2} = \epsilon_0$. That means at the beginning of the algorithm (as long as $\delta_x \geq \epsilon_0$) we do not necessarily have a decrease in terms of the Newton decrements. This is only guaranteed when $\delta_x < \epsilon_0$. On the other hand this condition will be met eventually because of Theorem 2.2.23 in combination with the left-hand side inequality of (2.16).

Let us summarize the overall complexity result of the globalized Newton method in the following theorem.

Theorem 2.2.24. *Let F be self-concordant, $x_0 \in \mathcal{C}$, $0 < \epsilon < \omega_*(1/2)$ and choose $\bar{\beta} \in \left(0, \frac{3-\sqrt{5}}{2}\right)$. Then we can find a point $\bar{x} \in \mathcal{C}$ such that*

$$F(\bar{x}) - F(x^*) \leq \epsilon$$

in no more than

$$N = N_1 + N_2$$

iterations, where

$$\begin{aligned} N_1 &\leq \frac{F(x_0) - F(x^*)}{\omega(\bar{\beta})} \\ N_2 &= \mathcal{O}\left(\log_2\left(\log_2\left(1/\epsilon\right)\right)\right). \end{aligned}$$

Proof. Indeed, as long as $\delta_x \geq \bar{\beta}$, we can apply damped Newton steps with $\alpha = \frac{1}{1+\delta_x}$, as described in Theorem 2.2.23. By doing so, in each iteration we can reduce the function value by $\omega(\delta_x) \geq \omega(\bar{\beta})$. That means the original optimality gap $F(x_0) - F(x^*)$ will be reduced at most

$$\frac{F(x_0) - F(x^*)}{\omega(\bar{\beta})}$$

times before $\delta_x < \bar{\beta}$. If $\epsilon \geq \omega_*(\bar{\beta})$, then we have in view of (2.16)

$$F(x) - F(x^*) \leq \omega_*(\delta_x) < \omega_*(\bar{\beta}) \leq \epsilon,$$

which means that x is an ϵ-solution and we can stop here.

Otherwise, as soon as $\delta_x < \bar{\beta} < \frac{3-\sqrt{5}}{2} =: \epsilon_0$, we switch to the full Newton method with step size $\alpha = 1$. Once we have entered phase 2, we have according to Theorem 2.2.19

$$\delta_{x^+} \leq \left(\frac{\delta_x}{1-\delta_x}\right)^2 \leq \underbrace{\frac{1}{(1-\epsilon_0)^2}}_{=:\kappa} \cdot \delta_x^2 < \underbrace{\kappa \cdot \epsilon_0}_{=1} \cdot \delta_x = \delta_x.$$

Note that $\kappa\,\epsilon_0 = 1$ because ϵ_0 is a solution to the nonlinear equation $\frac{x}{(1-x)^2} = 1$. If we denote now by $\delta_x^{(k)}$ the Newton decrement in the k-th step of phase 2, we get for all $k > 1$ that $1/(1-\delta_x^{(k)})^2 \leq \kappa$ (because $\delta_x^{(k)} < \delta_x^{(1)} < \epsilon_0$), and recursively

$$\delta_x^{(k)} \leq \kappa \cdot (\delta_x^{(k-1)})^2 \leq \kappa \cdot \left[\kappa \left(\delta_x^{(k-2)}\right)^2\right]^2 \leq (\kappa \cdot \delta_x^{(1)})^{2^{k-1}} \cdot \kappa^{-1},$$

where $\delta_x^{(1)}$ is the first Newton decrement such that $\delta_x^{(1)} < \epsilon_0$. We apply the quadratically convergent phase of Newton's method until the decrement is less than ϵ. To achieve that it suffices to make sure that

$$(\kappa \cdot \delta_x^{(1)})^{2^{k-1}} \leq \epsilon \cdot \kappa,$$

which in turn is satisfied when

$$\log_2\left((\kappa \cdot \delta_x^{(1)})^{2^{k-1}}\right) = \underbrace{2^{k-1} \cdot \log_2\left(\underbrace{\kappa \cdot \delta_x^{(1)}}_{<1}\right)}_{<0} \leq \underbrace{\log_2(\underbrace{\epsilon\kappa}_{<1})}_{<0}.$$

This is true if and only if

$$2^{k-1} \cdot \log_2\left(\frac{1}{\kappa \cdot \delta_x^{(1)}}\right) \geq \log_2\left(\frac{1}{\epsilon\kappa}\right).$$

It follows that we have to ensure

$$2^{k-1} \geq \frac{\log_2\left(\frac{1}{\epsilon\kappa}\right)}{\log_2\left(\frac{1}{\kappa \cdot \delta_x^{(1)}}\right)} = \mathcal{O}\left(\log_2\left(1/\epsilon\right)\right)$$

We conclude that it suffices to run at most

$$k = \mathcal{O}\left(\log_2\left(\log_2\left(1/\epsilon\right)\right)\right)$$

iterations in phase 2 to guarantee $\delta_x^{(k)} \leq \epsilon$. This means we need very few iterations to achieve basically any desired accuracy. For example, for $\epsilon = 10^{-10}$, we need no more than 6 iterations.

It remains to note that if we want to find a point x such that $F(x) - F(x^*) \leq \epsilon$, then in accordance with (2.16), this can be guaranteed by satisfying $\omega_*(\delta_x) \leq \epsilon$. Finally, for $\epsilon < \omega_*(\bar{\beta}) < 0.1$ we have that $\omega_*(\epsilon) < \epsilon$ (see Figure 2.1), which finishes the proof. □

Remark 2.2.25. *The bound on the number of iterations in phase 1 is rather pessimistic. In fact, Theorem 2.2.23 guarantees an absolute decrease in the function value $F(x)$ of at least $\omega(\delta_x)$ which can in fact be much larger than $\omega(\bar{\beta})$. Moreover, it could be possible to find a larger step size than $\frac{1}{1+\delta_x}$ that achieves a higher functional decrease. For example, one can implement a simple line search along the Newton direction Δx that starts with the safeguard step length of $\alpha_0 = \frac{1}{1+\delta_x}$ and that gradually increases α until the function values of F are increasing again. This results in a better practical performance of the algorithm.*

2.3 Equality constrained optimization

From now on we consider the underlying vector space $\mathcal{E}$ to be $\mathbb{R}^n$. In this section we want to extend the results from the previous section to the case where we want to minimize a convex function F subject to linear equality constraints, i.e. we consider the problem

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \; & F(x) \\ & x \in \mathcal{L} = \{x : Ax = b\}, \end{aligned} \tag{2.18}$$

where $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ is a convex function defined on some open domain $\mathcal{C}$.

We have seen in Section 2.2 that if there are no equality constraints present and if F is well-behaved (that is, F is self-concordant), we can solve the unconstrained problem (2.2) in with a guaranteed complexity (see Theorem 2.2.24). Therefore we keep the assumption that the objective F is a nondegenerate self-concordant function. We will show that we can obtain essentially the same results as in Section 2.2.

In this work we assume that there exist points $x \in \mathcal{C}$ such that $Ax = b$ (otherwise (2.18) is trivially infeasible). The latter condition is true if and only if $\operatorname{rank}(A) = \operatorname{rank}([A, b])$, where $[A, b]$ denotes the matrix where vector b is appended to the matrix A. This, in turn, is in particular satisfied if A has full row-rank. However, we want to stress here that the full row rank condition is not necessary. On the other hand, if A is rank-deficient, although there are solutions to the system, it is clear that we can always reduce $Ax = b$ to a system with fewer rows $\tilde{A}x = \tilde{b}$ with a matrix that has full row rank. For convenience we will assume in the rest of this chapter that A has full row rank. This simplifies the analysis, for example it has the advantage that we can write down explicitly a generalized definition of the Newton directions.

2.3.1 From unconstrained to equality constrained optimization

A direct application of the KKT conditions to the problem (2.18) yields the following optimality conditions. The point $x^* \in \mathcal{C}$ is optimal for (2.18) if and only if

$\exists\ \lambda^*$ such that

$$\begin{aligned} \nabla F(x^*) &= -A^T\lambda^* \\ Ax^* &= b. \end{aligned} \tag{2.19}$$

Analogously to the previous section, (2.19) is a nonlinear system of equations in the variables x^* and λ^*. Again, we might try to find a solution to the optimality conditions iteratively by linearizing (2.19) at a given feasible point $x \in \mathcal{C} \bigcap \mathcal{L}$. The only nonlinear term in (2.19) is the left-hand side expression in the first line. Its linearization becomes

$$\nabla F(x + \Delta x) \approx \nabla F(x) + \nabla^2 F(x)\,\Delta x.$$

Further, we want to restrict ourselves to points in $\mathcal{L}$. Since $Ax = b$, we must therefore impose $A\Delta x = 0$. We arrive at the following linear system (also called the *augmented system*)

$$\begin{bmatrix} \nabla^2 F(x) & A^T \\ A & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla F(x) \\ 0 \end{bmatrix}. \tag{2.20}$$

Note that the solution of (2.20) is unique, since F is assumed to be nondegenerate which implies $\nabla^2 F(x) \succ 0$ for all $x \in \mathcal{C}$. If we multiply the first equation by $\nabla^2 F(x)^{-1}$, we get

$$\Delta x = -\nabla^2 F(x)^{-1}(\nabla F(x) + A^T\lambda). \tag{2.21}$$

Replacing the above term in the second equation of (2.20) yields a reduced linear system only in terms of λ (also called the *normal equation*):

$$A\nabla^2 F(x)^{-1} A^T\,\lambda = -A\nabla^2 F(x)^{-1}\nabla F(x).$$

We have assumed that A has full row rank. Therefore the system matrix $A\nabla^2 F(x)^{-1}A^T$ is positive definite too and we get as unique solution for λ

$$\lambda = -[A\nabla^2 F(x)^{-1}A^T]^{-1}\,A\nabla^2 F(x)^{-1}\nabla F(x). \tag{2.22}$$

Replacing this expression in the (2.21) gives the unique solution for Δx.

Newton direction as a projection

Note that the Newton direction Δx is now given by the linear system (2.20), whose unique solution we have computed in (2.21) and (2.22). If we substitute λ from (2.22) in (2.21), we get

$$\begin{aligned} \Delta x &= \underbrace{\left[I - \nabla^2 F(x)^{-1} A^T [A\nabla^2 F(x)^{-1} A^T]^{-1} A\right]}_{=\mathcal{P}^F_{\mathcal{L}_0,x}} \cdot \underbrace{\left(-\nabla^2 F(x)\right)^{-1}\nabla F(x)}_{=\Delta x^{(u)}} \\ &= \mathcal{P}^F_{\mathcal{L}_0,x}\,\Delta x^{(u)}, \end{aligned}$$

where $\Delta x^{(u)}$ is the Newton direction for the *unconstrained* problem $\min_x F(x)$, i.e. the solution to (2.5), and $\mathcal{P}^F_{\mathcal{L}_0,x}$ is the projection operator onto $\mathcal{L}_0 = \{x : Ax = 0\}$, with respect to the local norm. To check the latter interpretation of $\mathcal{P}^F_{\mathcal{L}_0,x}$, let us consider the corresponding problem of projecting a given point x_0 onto $\mathcal{L}_0$ with respect to the local norm $||\cdot||_x$ for a particular point $x \in \mathcal{L}_0$, i.e.

$$\min_{y\in\mathcal{L}_0} \; ||y-x_0||_x^2 \tag{2.23}$$

The objective can be written as

$$\begin{aligned} ||y-x_0||_x^2 &= (y-x_0)^T\nabla^2F(x)(y-x_0) \\ &= y^T\nabla^2F(x)y - 2y^T\nabla^2F(x)x_0 + x_0\nabla^2F(x)x_0 \end{aligned}$$

The KKT conditions for the convex quadratic problem (2.23) become then: $\exists\,\lambda$ such that

$$\begin{aligned} 2\nabla^2F(x)\,y - 2\nabla^2F(x)\,x_0 &= -2A^T\lambda \\ Ay &= 0 \end{aligned}$$

From the first equation we get

$$\begin{aligned} y &= \nabla^2F(x)^{-1}\left[-A^T\lambda + \nabla^2F(x)\,x_0\right] \\ &= -\nabla^2F(x)^{-1}A^T\lambda + x_0. \end{aligned}$$

Replacing this term in the second equation of the optimality conditions gives

$$Ay = A[-\nabla^2F(x)^{-1}A^T\lambda + x_0] = 0.$$

Since A has full row rank, from the last equation we can derive λ and get

$$\lambda = \left[A\nabla^2F(x)^{-1}A^T\right]^{-1}Ax_0.$$

Substituting λ in x yields

$$\begin{aligned} y &= -\nabla^2F(x)^{-1}A^T\left[A\nabla^2F(x)^{-1}A^T\right]^{-1}A\,x_0 + x_0 \\ &= \left[I - \nabla^2F(x)^{-1}A^T[A\nabla^2F(x)^{-1}A^T]^{-1}A\right]x_0 \\ &= \mathcal{P}^F_{\mathcal{L}_0,x}\,x_0. \end{aligned}$$

The Newton decrement

Analogously to Section 2.2 the Newton decrement δ_x is defined as the local norm of the Newton direction Δx. The only difference is that Δx is now defined as the solution of the linear system (2.20) (as opposed to (2.5) in Section 2.2).

Definition 2.3.1. *Let F be self-concordant and $x \in \mathcal{C}$. We define the* Newton decrement *at the point x as*

$$\delta_x = ||\Delta x||_x$$

where Δx is the solution of (2.20).

Using (2.21), we get two alternative representation of δ_x.

$$\begin{aligned}\delta_x &= \left(-\Delta x^T \nabla^2 F(x) \nabla^2 F(x)^{-1} (\nabla F(x) + A^T\lambda)\right)^{1/2} \\ &= \left(-\Delta x^T (\nabla F(x) + A^T\lambda)\right)^{1/2} \\ &= \left(-\Delta x^T \nabla F(x) - \underbrace{\Delta x^T A^T}_{=0} \lambda\right)^{1/2} \\ &= \left(-\Delta x^T \nabla F(x)\right)^{1/2}.\end{aligned}$$

Alternatively, (2.21) yields

$$\begin{aligned}\delta_x &= \left((\nabla F(x) + A^T\lambda)^T \nabla^2 F(x)^{-1} (\nabla F(x) + A^T\lambda)\right)^{1/2} \\ &= ||\nabla F(x) + A^T\lambda||_x^*.\end{aligned}$$

Let us compare the Newton decrement of Definition 2.3.1 (from the equality constrained problem (2.18)) to the Newton decrement of Definition 2.2.17 (from the unconstrained problem (2.2)).

Lemma 2.3.2. *Let F be self-concordant and $x \in \mathcal{C}$ such that $Ax = b$. Denote by δ_x the Newton decrement for equality constrained problem (2.18), and $\delta_x^{(u)}$ the Newton decrement of the unconstrained problem (2.2). Then*

$$\delta_x \leq \delta_x^{(u)}.$$

Proof. The Newton decrement for (2.18) is given by

$$\delta_x = \left(-\nabla F(x)^T \Delta x\right)^{1/2},$$

where Δx is the solution of (2.20). We get

$$\begin{aligned}\delta_x^2 &= -\nabla F(x)^T \Delta x \\ &= \underbrace{\nabla F(x)^T \nabla^2 F(x)^{-1} \nabla F(x)}_{=\left(\delta_x^{(u)}\right)^2} + \nabla F(x)^T \nabla^2 F(x)^{-1} A^T \lambda \\ &= \left(\delta_x^{(u)}\right)^2 - \nabla F(x)^T \nabla^2 F(x)^{-1} A^T [A \nabla^2 F(x)^{-1} A^T]^{-1} A \nabla^2 F(x)^{-1} \nabla F(x).\end{aligned}$$

The second term is non-positive because the matrix in the middle $[A\nabla^2 F(x)^{-1} A^T]^{-1}$ is positive semidefinite. We conclude that

$$\delta_x^2 \leq \left(\delta_x^{(u)}\right)^2,$$

which implies the result. □

The above Lemma certainly makes intuitive sense: the Newton decrements $\delta_x^{(u)}$ and δ_x are defined as the (local) norms of the unconstrained ($\Delta x^{(u)}$) and constrained (Δx) Newton directions. Since Δx is confined to the subspace null$(A) = \{x : Ax = 0\}$, the point $x + \Delta x$ should be closer to x than $x + \Delta x^{(u)}$, which does not need to be in $\mathcal{L} = \{x : Ax = b\}$.

Variable elimination and link to the unconstrained case

We can also derive the optimality conditions in another way. We see that (2.18) is the minimization of the restriction of a self-concordant function F to the affine subspace $\mathcal{L} = \{x \in \mathbb{R}^n : Ax = b\}$. In view of Section 2.2.2 we can find $B \in \mathbb{R}^{n,n-m}$ such that $\text{range}(B) = \text{null}(A)$ and $c \in \mathcal{L}$ and define the unconstrained problem

$$\min_{y \in \mathcal{C}^+} F|_{\mathcal{L}}(y) = F(By + c), \tag{2.24}$$

with $\operatorname{dom} F|_{\mathcal{L}} = \mathcal{C}^+ = \{y \in \mathbb{R}^{n-m} : By + c \in \mathcal{C}\}$, which is in fact equivalent to (2.18). The optimality conditions for (2.24) are

$$\nabla F|_{\mathcal{L}}(y) = 0.$$

Using the definition of $F|_{\mathcal{L}}$, we get

$$\nabla F|_{\mathcal{L}}(y) = B^T \nabla F(By + c).$$

That means a point y is optimal for (2.24) if and only if $B^T \nabla F(By + c) = 0$, or in other words $\nabla F(By + c) \in \text{null}(B^T)$. Since $\text{range}(B) + \text{null}(B^T) = \mathbb{R}^n$, we conclude that $\text{range}(B) = \text{null}(A)$ is equivalent to $\text{range}(A^T) = \text{null}(B^T)$. Therefore, $y \in \mathbb{R}^p$ is optimal for (2.24) if and only if $\nabla F(By + c) \in \text{range}(A^T)$, which means we can find multipliers $\lambda \in \mathbb{R}^m$ such that

$$\nabla F(By + c) = -A^T \lambda.$$

Moreover, for any $y \in \mathcal{C}^+$ we have

$$A(By + c) = \underbrace{AB}_{=0} y + \underbrace{Ac}_{=b} = b,$$

using the fact that $\text{range}(B) = \text{null}(A)$ is the same as saying $A\,B = 0 \in \mathbb{R}^{m,n-m}$. That means if we define $x = By + c$ for any $y \in \mathcal{C}^+$ the above optimality conditions are exactly the same as (2.19).

Yey another way of deriving the Newton system (2.20) is to look at the linearization of the optimality conditions of the unconstrained problem (2.24), i.e.

$$\nabla F|_{\mathcal{L}}(y) + \nabla^2 F|_{\mathcal{L}}(y)\,\Delta y = 0.$$

We have computed the gradient $\nabla F|_{\mathcal{L}}$ above. The Hessian becomes

$$\nabla^2 F|_{\mathcal{L}}(y) = B^T \nabla^2 F(By + c) B.$$

Replacing the expressions for $\nabla F|_{\mathcal{L}}$ and $\nabla^2 F|_{\mathcal{L}}$ in the linearized optimality conditions yields

$$B^T \left[\nabla F(By + c) + \nabla^2 F(By + c) B \Delta y\right] = 0.$$

In other words, we need that $\nabla F(By + c) + \nabla^2 F(By + c) B \Delta y \in \text{null}(B^T) = \text{range}(A^T)$. That means we can find $\lambda \in \mathbb{R}^m$ such that

$$\nabla F(By + c) + \nabla^2 F(By + c) B \Delta y = -A^T \lambda.$$

If we define $\Delta x = B\Delta y$, we get by construction $A\Delta x = AB\Delta y = 0$ (since $AB = 0$ in view of the assumption range(B) = null(A)). Denoting $x = By + c \in \mathcal{C}$, we arrive at the following system of equations

$$\begin{aligned} \nabla^2 F(x)\Delta x + A^T\lambda &= -\nabla F(x) \\ A\,\Delta x &= 0, \end{aligned}$$

which is exactly (2.20).

In view of these observations we could stop here and refer to the results of Section 2.2, since (2.18) is equivalent to (2.24) and everything can be phrased in terms of an unconstrained problem of minimizing a self-concordant function (whose derivatives we can compute). However, we have decided to keep the equality constrained formulation (2.18) in its explicit form for several reasons.

1. We do not want a potential user to be occupied with the process of eliminating variables, i.e. with the task of finding a matrix B such that range(B) = null(A) (we do assume, however, that a particular solution $c \in \mathbb{R}^n$ to the linear constraints is known). Moreover, the computation of B might slow down the total computation time and destroy sparsity.
2. We prefer the equality constrained formulation (2.18) since they explicitly contain the variables x that have an actual meaning for the original application (from where (2.18) has arisen).
3. Later (Chapter 5) we are going to exploit structure in the matrix A and in the domain of the objective function. This structure gets lost when the variables are eliminated as in (2.24).

As we have seen in (2.24), the equality constrained problem (2.18) is equivalent to the unconstrained minimization of the composition of F with an affine operator $\mathcal{A}(y) = By + c$, where $B \in \mathbb{R}^{n,n-m}$ such that range(B) = null(A) and $c \in \mathcal{L} = \{x : Ax = b\}$. The following result shows the link between the Newton decrements for both problems.

Let us change a bit our notation here. For the particular solution $x = c \in \mathcal{L}$ we denote by $f_x(y)$ the objective function of the unconstrained problem (2.24), that is

$$f_x(y) = F(By + x).$$

In view of the calculations above the derivatives of f_x are given by:

$$\begin{aligned} \nabla f_x(y) &= B^T\,\nabla F(By + x), \\ \nabla^2 f_x(y) &= B^T\,\nabla^2 F(By + x)\,B, \end{aligned}$$

and the optimality conditions for (2.24) are

$$\nabla f_x(y) = 0.$$

The Newton direction and Newton decrement of f_x at the point $y = 0$ (which corresponds to the particular $x \in \mathbb{R}^n$) are given by

$$\begin{aligned} \Delta y_0 &= -\nabla^2 f_x(0)^{-1}\,\nabla f_x(0) = -[B^T\,\nabla^2 F(x)\,B]^{-1}\,B^T\nabla F(x), \\ \delta_{y_0} &= ||\Delta y_0||_{\nabla^2 f_x(0)} = \left(-\nabla f_x(0)^T\Delta y_0\right)^{1/2}. \end{aligned}$$

Lemma 2.3.3. *Let F be self-concordant and $x \in \mathcal{C}$ such that $Ax = b$. Let δ_{y_0} be the Newton decrement for the unconstrained problem (2.24) at the point $y = 0$, and δ_x the Newton decrement for the equality constrained problem (2.18) at the point x. Then we have*

$$\delta_x = \delta_{y_0}.$$

Proof. We have seen above that $B\Delta y_0 \in \operatorname{null}(A)$ because $AB\Delta y_0 = 0$.

Further, let us define

$$\lambda = -\left[AA^T\right]^{-1} A \cdot \left[\nabla F(x) + \nabla^2 F(x) B\Delta y_0\right],$$

we see that it holds

$$\nabla^2 F(x) B\Delta y_0 + A^T\lambda = -\nabla F(x).$$

The latter statement is true because of the following fact. Let us define

$$M = \begin{bmatrix} A \\ B^T \end{bmatrix} \in \mathbb{R}^{n,n}.$$

Since $AB = 0$ implies that $\operatorname{rank}(M) = \operatorname{rank}(A) + \operatorname{rank}(B) = m + (n - m) = n$ (see e.g. [4, Fact 2.10.25]), we conclude that M is nonsingular. Therefore, when multiplying the above equation from the left with M^T, we get

$$\underbrace{B^T\nabla^2 F(x)B}_{=\nabla^2 f_x(0)}\Delta y_0 + \underbrace{B^T A^T}_{=0}\lambda = -\underbrace{B^T\nabla F(x)}_{=\nabla f_x(0)}$$

and

$$A\nabla^2 F(x)B\Delta y_0 + \underbrace{AA^T\lambda}_{=-A\cdot[\nabla F(x)+\nabla^2 F(x)B\Delta y_0]} = -A\nabla F(x).$$

This means the direction $B\Delta y_0$ is in fact the Newton direction for (2.18), because together with the λ defined above it satisfies (2.20), i.e. $\Delta x = B\Delta y_0$.

Consequently, the Newton decrement of f_x at the point $y = 0$ is

$$\delta_{y_0}^2 = -\nabla f_x(0)^T \Delta y_0 = -\nabla F(x)^T B\Delta y_0 = -\nabla F(x)^T\Delta x = \delta_x^2.$$

□

However, we want to stress here that the above mentioned result has only limited practical use because typically we do not have access to an elimination matrix B (that is expensive to compute). On the other hand, Lemma 2.3.3 will be useful to generalize some of the results from the previous section to the equality constrained case.

2.3.2 Newton's method for equality constrained minimization

Analogously to Section 2.2 we can consider Newton's method for equality constrained problems (2.18), where the Newton direction Δx is the solution of the linear system (2.20), and the Newton decrement is defined as the local norm of this direction, i.e. $\delta_x = ||\Delta x||_x$. In order to analyze such a Newton method we have the following two observations.

1. The Newton direction Δx lies in the null space of A. This means that if some point $x \in \mathcal{C}$ satisfies the linear constraints $Ax = b$, then so do all points along the Newton direction, i.e. $A(x + \gamma\Delta x) = b$ for any step size $\gamma \in \mathbb{R}$.

2. Using Lemma 2.3.3 we have that the Newton direction Δx for the equality constrained problem (2.18) can be expressed as $\Delta x = B\,\Delta y_0$, where Δy_0 is the Newton direction for the unconstrained problem (2.24) at $y = 0$. Moreover, both Newton decrements are the same: $\delta_x = \delta_{y_0}$. Using these two relations, it is straightforward to phrase all the expressions in terms of the variables x in Theorem 2.2.18, Theorem 2.2.19 and Theorem 2.2.23 from Section 2.2 in terms of the unconstrained variables y.

For the sake of completeness, let us present here the generalizations of Theorem 2.2.18, Theorem 2.2.19 and Theorem 2.2.23 to the equality constrained case. The Newton direction Δx denotes the solution of (2.20) and the Newton decrement denotes its local norm, i.e. $\delta_x = ||\Delta x||_x$.

Theorem 2.3.4. *Let F be self-concordant, $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $x \in \mathcal{C}$ such that $Ax = b$ and $\delta_x < 1$. Then*

$$\omega(\delta_x) \leq F(x) - F(x^*) \leq \omega_*(\delta_x), \tag{2.25}$$

$$\omega'(\delta_x) \leq ||x - x^*||_x \leq \omega_*'(\delta_x), \tag{2.26}$$

where x^* denotes an optimal solution for (2.18).

Theorem 2.3.5. *Let F be self-concordant, $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $x \in \mathcal{C}$ such that $Ax = b$ and $\delta_x < 1$. Then $x^+ = x + \Delta x \in \mathcal{C}$ with $Ax^+ = b$ and*

$$\delta_{x^+} \leq \left(\frac{\delta_x}{1 - \delta_x}\right)^2,$$

where δ_{x^+} denotes the (constrained) Newton decrement at the new iterate x^+.

Theorem 2.3.6. *Let F be self-concordant, $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $x \in \mathcal{C}$ such that $Ax = b$. We define the new iterate*

$$x^+ = x + \frac{1}{1 + \delta_x} \cdot \Delta x.$$

Then $x^+ \in \mathcal{C}$ and $Ax^+ = b$. Moreover, we have

$$F(x^+) \leq F(x) - \omega(\delta_x).$$

We are ready now to formulate the globalized Newton method for solving (2.18), the only difference being the Newton direction, that comes now from (2.20). We have again two phases:

1. **Damped phase:**
   As long as $\delta_x \geq \bar{\beta}$ with $\bar{\beta} < \frac{3-\sqrt{5}}{2}$ we can choose $\alpha = \frac{1}{1+\delta_x}$ which guarantees in each iteration a functional improvement of at least $\omega(\bar{\beta})$.

Algorithm 2 Damped Newton method for minimizing a self-concordant function subject to linear equality constraints

```
Input: $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$, $F$ self-concordant
Initialize: $\epsilon > 0$, initialize $x_0 \in \mathcal{C} \bigcap \{x : Ax = b\}$.
  loop
    1) compute Newton direction $\Delta x$ from (2.20)
    2) $x \leftarrow x + \alpha\,\Delta x$, where $\alpha$ is a suitable step length
    3) $\delta_x = ||\Delta x||_x$
  end loop
```

2. **Quadratically convergent phase:**
   If $\delta < \bar{\beta} < \frac{3-\sqrt{5}}{2}$, then we can choose $\alpha = 1$ and we have quadratic convergence with respect to the Newton decrement.

Theorem 2.3.7. *Let F be self-concordant, $x_0 \in \mathcal{C}$, such that $Ax_0 = b$, $\epsilon > 0$ and choose $\bar{\beta} \in \left(0, \frac{3-\sqrt{5}}{2}\right)$. Then we can find a point $\bar{x} \in \operatorname{dom} F = \mathcal{C}$ such that $A\bar{x} = b$ and*

$$F(\bar{x}) - F(x^*) \leq \epsilon$$

in no more than

$$N = N_1 + N_2$$

iterations, where

$$\begin{aligned} N_1 &\leq \frac{F(x_0) - F(x^*)}{\omega(\bar{\beta})}, \\ N_2 &= \mathcal{O}\left(\log_2\left(\log_2\left(1/\epsilon\right)\right)\right). \end{aligned}$$

2.3.3 Cost per iteration

The main cost per iteration in Algorithm 2 constitutes the process of solving (2.20). Now we are going to determine the complexity of this operation for the general case and several special situations. We estimate the complexity in terms of the floating-point operations (flops) that have to be carried out. A floating-point operation is an addition, a subtraction, a multiplication or a division of two floating-point numbers.

We will assume here that the matrix A has full row rank. We have argued at the beginning of this section that this assumption is – strictly speaking – not necessary. On the other hand it is not too restrictive either, because we can always reduce (2.18) to the minimization of a convex function subject to a linear system of equations with a matrix of full row rank.

The general case

One way of solving the system (2.20) is by using an LU factorization of the system matrix. The complexity of the factorization and the forward and backward substitutions to compute the final solution is

$$\frac{2}{3}(n+m)^3 + 2(n+m)^2 \tag{2.27}$$

flops (see, e.g. [6, Appendix C.3.1]).

However, we see that the system matrix has a special structure that can be exploited. The Hessian block $\nabla^2 F(x)$ is assumed to be positive definite and A has full row rank. We have seen above that the solution of (2.20) is given by (2.21) and (2.22), i.e.

$$\Delta x = -\nabla^2 F(x)^{-1}(\nabla F(x) + A^T \lambda),$$

where

$$\lambda = -[A\nabla^2 F(x)^{-1} A^T]^{-1} A\nabla^2 F(x)^{-1} \nabla F(x).$$

We see that computing λ and Δx involves the solving of linear systems with the system matrix $\nabla^2 F(x)$, which is positive definite. For this solving process we can make use of a Cholesky factorization of $\nabla^2 F(x)$, whose complexity is $\frac{1}{3}n^3$ (see [6, Appendix C.3.2]). The complexity of the forward and backward substitution for computing $\nabla^2 F(x)^{-1} A^T$ and $\nabla^2 F(x)^{-1} \nabla F(x)$ is $2(m+1)n^2$ flops (one forward substitution costs n^2 flops, the same for the backward substitution, in total there are $(m+1)$ right-hand side vectors). The next operations are the multiplications of A with the vector $\nabla^2 F(x)^{-1} \nabla F(x)$ ($2nm$ flops) and the matrix $\nabla^2 F(x)^{-1} A^T$ ($2m^2 n$ flops). The following step is the factorization of the positive definite matrix $A\nabla^2 F(x)^{-1} A^T$ ($\frac{1}{3}m^3$ flops) and the forward and backward substitution to compute λ, its complexity is $2m^2$. Finally, we have to execute the multiplication of $\nabla^2 F(x)^{-1} A^T$ with λ ($2nm$ flops) and add two vectors of size n (n flops).

If we count all the floating-point operations, we get the following complexity of solving (2.20) using a Cholesky factorization of the system matrices:

$$\frac{1}{3}n^3 + \frac{1}{3}m^3 + 2n^2 m + 2nm^2 + 2n^2 + 4nm + 2m^2 + n$$

which can be written in the more compact form

$$\frac{1}{3}(n+m)^3 + 2(n+m)^2 + n^2 m + nm^2 + n. \tag{2.28}$$

If we compare (2.28) and (2.27), we see that solving (2.20) using a Cholesky factorization of the system matrices is cheaper as compared to solving it using an LU factorization, since

$$n^2 m + nm^2 + n < \frac{1}{3}n^3 + n^2 m + nm^2 + \frac{1}{3}m^3 = \frac{1}{3}(n+m)^3,$$

unless $n = 1$ and $m = 0$.

Typically, the system (2.20) is not dense, which means that the actual complexity of solving it is much lower. Below, we analyze the complexity of solving (2.20) if the system matrix has a particular structure which we can exploit. These results will be used later in Section 5.3.5.

Diagonal Hessian

Let us consider now the special case where $\nabla^2 F(x)$ is not only positive definite, but additionally diagonal. This situation will occur e.g. in Section 5.3.1 when computing the Newton directions for the partial minimization subproblems. In that special case some of the operations above are significantly cheaper. In particular, it is not necessary to factorize $\nabla^2 F(x)$. Instead, $\nabla^2 F(x)^{-1} A^T$ and $\nabla^2 F(x)^{-1} \nabla F(x)$ can be computed by only $nm + n$ flops (each component of $\nabla F(x)$ and of A^T is simply scaled). The rest is unchanged as compared the the previous case. We get that the complexity of solving (2.20) with a diagonal Hessian is

$$\frac{1}{3}m^3 + 2nm^2 + 5nm + 2n + 2m^2. \tag{2.29}$$

Diagonal Hessian and sparse A

Let us consider now the very particular situation of a diagonal positive definite Hessian $\nabla^2 F(x)$ and a matrix A with full row rank and at most one nonzero in each column. As in the previous case the complexity of computing $\nabla^2 F(x)^{-1} \nabla F(x)$ is n flops. Since A has in total at most n nonzeros the complexity of computing $\nabla^2 F(x)^{-1} A^T$ is also at most n. For the same reason the complexity of multiplying A with the vector $\nabla^2 F(x)^{-1} \nabla F(x)$ is at most $2n$ flops. Moreover, $\nabla^2 F(x)^{-1} A^T$ has the same sparsity pattern as A. Since A has at most one nonzero in each column, we conclude that $A \nabla^2 F(x)^{-1} A^T$ must be diagonal. The cost of computing these diagonal elements is $2n$ flops. The complexity of solving a linear system with the diagonal matrix $A \nabla^2 F(x)^{-1} A^T$ is m flops. Finally, the cost of multiplying $\nabla^2 F(x)^{-1} A^T$ with λ is n and additionally we need to perform the addition of two vectors of size n to get the solution (cost: n flops). If we sum all the floating-point operations we get a total complexity of solving (2.20) with a diagonal Hessian and a sparse matrix A of

$$8n + m. \tag{2.30}$$

Multiple right-hand sides

Let us finally consider the situation of a diagonal positive definite Hessian $\nabla^2 F(x)$, a sparse matrix A with full row rank and at most one nonzero in each column and with several (say: k) right-hand sides, i.e.

$$\begin{bmatrix} D & A^T \\ A & 0 \end{bmatrix} \cdot \begin{bmatrix} J \\ L \end{bmatrix} = \begin{bmatrix} M \\ N \end{bmatrix}$$

where $D \in \mathbb{R}^{n,n}$ is a diagonal matrix, $A \in \mathbb{R}^{m,n}$, $M \in \mathbb{R}^{n,k}$ and $N \in \mathbb{R}^{m,k}$. With the same reasoning as above, we get that the solutions of the above system are

$$\begin{aligned} L &= [AD^{-1}A^T]^{-1} \, (AD^{-1}M - N), \\ J &= D^{-1} \, (M - A^T \, L). \end{aligned}$$

We can see that the complexity of computing $D^{-1}M$ is kn flops (since M has kn components). As in the previous case, the complexity of computing $D^{-1}A^T$

is n and of $AD^{-1}A^T$ is $2n$ flops. Since A has at most n nonzeros we have that the complexity of multiplying A with the matrix $D^{-1}M$ is at most $2kn$ flops. The addition of $AD^{-1}M$ with $-N$ costs mk flops. Since $AD^{-1}A^T$ is diagonal, we conclude that the complexity of solving a linear system with that diagonal matrix is m flops. Finally, the cost of computing the solution (one multiplication of $D^{-1}A^T$ with L and one addition) is $2kn$ flops. We conclude that the complexity of solving several systems of the form (2.20) with the same system matrix is

$$3n + 5nk + 2mk. \tag{2.31}$$

We see that if $N = 0$ and $k = 1$ we recover the same complexity as in (2.30), because the addition $AD^{-1}M + (-N)$ is not needed if $N = 0$ (i.e. $-m$ flops).

2.4 Constrained optimization

We have seen in Section 2.3 that equality constrained convex optimization problems are essentially equivalent to unconstrained convex problems, both with respect to the formulation (formally we can always eliminate some of the variables to generate an unconstrained convex problem) and with respect to Newton's method that nicely generalizes to the equality constrained situation. If convex inequalities are present instead of linear equalities, then the situation changes. On the one hand it is not possible anymore to remove these inequalities simply by eliminating some of the variables. On the other hand, Newton's method cannot be applied directly to the inequality constrained problem since its optimality conditions are not anymore a nonlinear system of equations. Instead, they additionally involve *inequalities.*

2.4.1 Problem statement

In this section we consider general convex optimization problems, i.e. the minimization of a linear objective function subject to constraints defined by a closed convex set $\mathcal{C}$ and linear equalities, i.e. problems of the form

$$\begin{aligned} &\min_x \, c^T x \\ &\text{s.t. } x \in \mathcal{C} \subseteq \mathbb{R}^n, \\ &\quad Ax = b. \end{aligned} \tag{2.32}$$

Note that if we have a convex (but not linear) objective function f, then we can introduce an epigraph variable τ (which will be the objective term) and add the convex constraint $f(x) \le \tau$, which results in the problem

$$\begin{aligned} &\min_{x,\tau} \, \tau \\ &\text{s.t. } x \in \mathcal{C}, \; f(x) \le \tau, \\ &\quad Ax = b. \end{aligned}$$

We see that this optimization problem is of the form (2.32) with the extended feasible set

$$\tilde{\mathcal{C}} = \{(x, \tau) : x \in \mathcal{C}, f(x) \le \tau\}.$$

We will impose the following three assumptions, which turn out not to be too restrictive.

Assumption 1

1. *$\mathcal{C} \subseteq \mathbb{R}^n$ is closed, convex and full-dimensional (the latter is equivalent to: $\mathcal{C}$ has nonempty interior).*

2. *$\mathcal{C} \bigcap \mathcal{L}$ (where $\mathcal{L} := \{x : Ax = b\}$) does not contain a straight line.*

3. *A has full row rank.*

4. *$\mathcal{C} \bigcap \mathcal{L} \neq \emptyset$.*

5. *(2.32) is bounded.*

For the first assumption, if $\mathcal{C}$ is not full-dimensional (but convex and closed), then we know that $\mathcal{C}$ can be written as the intersection of a full-dimensional closed convex set $\bar{\mathcal{C}}$ with an affine subspace $\bar{\mathcal{L}}$. These linear constraints can then be merged with the already present linear constraints $Ax = b$. The second assumption will be needed to ensure that the Newton directions are defined everywhere and that they are unique. The third assumption is not necessary per se, but as we have argued at the beginning of Section 2.3, this rank-deficiency can be circumvented by simply removing some of the constraints. The last two assumptions ensure that (2.32) admits feasible points and that its optimal value is bounded.

We want to emphasize here that we include the linear equality constraints $Ax = b$ explicitly in the formulation, even though they do not add generality. Indeed, as we have seen in Section 2.1.1, the set $\{x \in \mathbb{R}^n : Ax = b\}$ is a convex set and intersections with convex sets are convex too. However, we have decided too keep these constraints in the model for two reasons:

1. The elimination of some of the variables amounts to finding a particular point $\bar{x}$ such that $A\bar{x} = b$ and an elimination matrix B such that range$(B) =$ null(A). Then the constraint $\{x \in \mathcal{C}, Ax = b\}$ is the same as saying $(By+\bar{x}) \in \mathcal{C}$ in the variables y. We do not want a potential user to be concerned with the elimination process.

2. Later we want to exploit the structure of the convex set $\mathcal{C}$ (cf. Section 2.5). We could eliminate some of the variables x and thereby remove the linear equality constraints, but by doing so the structure in $\mathcal{C}$ gets lost, or it is more difficult to exploit that structure.

2.4.2 Path-following interior-point methods

Path-following schemes make use of so-called barrier functions whose domains are the interiors of *closed* convex sets $\mathcal{C}$. Note the difference to Section 2.2 and Section 2.3, where the *open* convex sets $\mathcal{C}$ were denoting the domains of the self-concordant functions F.

Definition 2.4.1. *Let $\mathcal{C}$ be a closed convex set with nonempty interior. A con-*

tinuous function F is a barrier *for $\mathcal{C}$ if*

$$\operatorname{dom} F = \operatorname{int} \mathcal{C}$$

and

$$F(x) \to \infty \text{ as } x \to \partial \mathcal{C}.$$

Remark 2.4.2. *We have seen before that self-concordant functions are barriers for the closure of their domain (cf. Theorem 2.2.6).*

The central path

Barrier functions can be used to define an important object, the so-called central path, which will lead us to an optimal solution of (2.32).

Definition 2.4.3. *Let F be a self-concordant function such that* $\operatorname{dom} F = \operatorname{int} \mathcal{C}$. *The* central path *of $\mathcal{C} \bigcap \mathcal{L}$ is defined as the set of (unique) solutions $\{x(t)\}$, with $t > 0$, for the parametrized family of linearly constrained convex problems*

$$\begin{aligned} \min\ t\, c^T x + F(x) &:= f_t(x) \\ Ax &= b. \end{aligned} \tag{P_t}$$

If $\mathcal{C}$ is bounded, then also $x(0)$ exists and it is called the analytic center *of $\mathcal{C}$.*

Intuitively we see that as t grows (P_t) approximates more and more the original problem (2.32). Indeed, feasibility with respect to $\mathcal{C}$ is always maintained since F is a barrier for $\mathcal{C}$ and therefore not defined outside $\operatorname{int} \mathcal{C}$. Moreover, for large t we increase the importance of the original objective term $c^T x$. Therefore we can expect $x(t)$ to be close to x^* for large t. Our goal is therefore to trace the central path as it leads us to an optimal solution.

The concept of path-following methods is the following: starting at some initial point $x_0 \in \operatorname{int} \mathcal{C} \bigcap \mathcal{L}$ and $t_0 > 0$ ($t_0 \geq 0$ if $\mathcal{C}$ is bounded) we solve the corresponding centering problem (P_t) with $t = t_0$. According to Theorem 2.3.7 this can be done with guaranteed complexity, provided that the objective function f_t is self-concordant. In view of Definition 2.6 this is the case, since self-concordance is not affected by adding terms of degree 1.

Given an approximation x_{t_0} for the point $x(t_0)$ on the central path we update the value of t from t_0 to t_1 and solve the new centering problem (P_t) with $t = t_1 > t_0$, starting at x_{t_0}.

For any $t > 0$ the optimality conditions for (P_t) are: $\exists\, \lambda(t)$ such that

$$\begin{aligned} \nabla f_t(x(t)) &= -A^T \lambda(t) \\ Ax(t) &= b, \end{aligned} \tag{2.33}$$

where the gradient of f_t at some $x \in \operatorname{int} \mathcal{C} \bigcap \mathcal{L}$ is given by

$$\nabla f_t(x) = t\, c + \nabla F(x).$$

We will go along Newton directions towards the minimizer of (P_t). To compute the Newton directions we need also the Hessian of the objective function f_t, which is given by

$$\nabla^2 f_t(x) = \nabla^2 F(x).$$

The Newton direction $\Delta x(t)$ is then defined as the solution of the linear system

$$\begin{bmatrix} \nabla^2 F(x) & A^T \\ A & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta x(t) \\ \lambda(t) \end{bmatrix} = \begin{bmatrix} -t\,c - \nabla F(x) \\ 0 \end{bmatrix}. \tag{2.34}$$

We denote the Newton decrement for (P_t) at the point x by

$$\delta_{x,t} = ||\Delta x(t)||_x.$$

If $\delta_{x,t}$ is small, it means x is close to $x(t)$ (see Theorem 2.3.4).

As the path-following framework is presented above, there are still two open questions.

1. How shall we update t (and therefore the target point $x(t)$) so that we can guarantee on the one hand polynomial complexity but on the other hand make sure not to lose sight of the central path?

2. What is a good stopping criterion for the process, i.e. at which value of t shall we stop tracing the path?

2.4.3 Self-concordant barriers

In order to answer the two questions that we have posed above, we need to impose an additional assumption on the self-concordant function F.

Definition 2.4.4. *A self-concordant function F is a ν-*self-concordant barrier[1] *for a closed convex set $\mathcal{C} \subseteq \mathbb{R}^n$ if*

$$\nabla F(x)^T \nabla^2 F(x)^{-1} \nabla F(x) \leq \nu, \ \forall x \in \operatorname{int} \mathcal{C}.$$

The value ν is called the parameter of the barrier F.

In view of Definition 2.2.17 we see that if F is a ν-self-concordant barrier, then the Newton decrement δ_x for the *unconstrained* problem

$$\min_x \ F(x)$$

is bounded by $\sqrt{\nu}$ for all $x \in \operatorname{int} \mathcal{C}$. Alternatively, we have

$$||\Delta x||_x = ||\nabla F(x)||_x^* \leq \sqrt{\nu},$$

where Δx denotes the Newton direction for the unconstrained problem (2.2). That means for ν-self-concordant barriers we impose a uniform bound on the size of the Newton step Δx for minimizing F.

We want to point out here again that in this section the set $\mathcal{C} \subseteq \mathbb{R}^n$ is assumed to be *closed*, as opposed to Section 2.2 and Section 2.3, where $\mathcal{C} \subseteq \mathbb{R}^n$ was denoting the *open* domain where the self-concordant function F was defined. This change of notation means that in this section we always assume that $\operatorname{dom} F = \operatorname{int} \mathcal{C}$.

A consequence of Definition 2.4.4 is the following lemma that can be found in [46, Theorem 4.2.4(1)].

[1] Some authors (e.g. Jarre [30]) have introduced the notion of self-limiting barriers, which corresponds to Definition 2.4.4, the only difference being that F is not assumed to be a self-concordant *function*.

Lemma 2.4.5. *Let F be a self-concordant barrier. Then we have for any $x \in \operatorname{dom} F$ and any $y \in \operatorname{dom} F$*

$$\nabla F(x)^T (y - x) \leq \nu.$$

We have seen in Section 2.2 that convex quadratic functions

$$F(x) = \frac{1}{2} x^T Ax + a^T x + \alpha$$

(and therefore also affine functions) are self-concordant functions. However, they are not self-concordant barriers, even if we assume A is positive definite. Indeed, we have $\nabla F(x) = Ax + a$ and $\nabla^2 F(x) = A$. Replacing this in Definition 2.4.4, and we get

$$\begin{aligned} \nabla F(x)^T (\nabla^2 F(x))^{-1} \nabla F(x) &= (Ax + a)^T A^{-1} (Ax + a) \\ &= x^T Ax + 2a^T x + aA^{-1} a \end{aligned}$$

which is clearly *not* bounded on $\mathbb{R}^n$.

Examples of self-concordant barriers

- $F : \mathbb{R}^n_{++} \to \mathbb{R}$, $F(x) = -\sum_{i=1}^n \log(x_i)$ is an n-self-concordant barrier for $\mathcal{C} = \mathbb{R}^n_+$.
- $F : \operatorname{int} \mathcal{C} \subseteq \mathbb{R}^{n+1} \to \mathbb{R}$, $F(x, \tau) = -\log(\tau^2 - \sum_{i=1}^n x_i^2)$ is a 2-self-concordant barrier for $\mathcal{C} = \mathbb{L}_n := \{(x, \tau) \in \mathbb{R}^n \times \mathbb{R} : ||x||_2 \leq \tau\}$.
- $F : \operatorname{int} \mathcal{C} \subseteq \mathcal{S}^n \to \mathbb{R}$, $F(X) = -\log(\det(X))$ is an n-self-concordant barrier for $\mathcal{C} = \{X \in \mathcal{S}^n : X \succeq 0\}$.
- $F : \operatorname{int} \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$, $F(x) = -\log(\varphi(x))$, where $\varphi(x) = -\frac{1}{2} x^T Ax + a^T x + \alpha$ with $A \succeq 0$ is a concave quadratic function. F is 1-self-concordant for $\mathcal{C} = \{x \in \mathbb{R}^n : \varphi(x) \geq 0\}$.

Operations that preserve self-concordance

Self-concordant barriers are naturally linked to convex sets where they are defined. We have seen in Section 2.1.1 that certain operations on convex sets preserve convexity. Therefore it is desirable that similar rules exist on the side of the barriers that preserve self-concordance.

1. **Barrier for intersections of convex sets.**
   Let $\mathcal{C}_1 \subseteq \mathbb{R}^n$ and $\mathcal{C}_2 \subseteq \mathbb{R}^n$ be closed convex sets, and let F_i be ν_i-self-concordant barriers for $\mathcal{C}_i$ respectively. Then

   $$F(x) = F_1(x) + F_2(x)$$

   is a $(\nu_1 + \nu_2)$-self-concordant barrier for $\mathcal{C} = \mathcal{C}_1 \bigcap \mathcal{C}_2$. This generalizes to intersections of more than 2 sets (Proof, see [52, Proposition 2.3.1(ii)]).

2. **Barrier for direct products of convex sets.**
   Let $\mathcal{C}_i \subseteq \mathcal{E}_i, i = 1, \ldots, m$ be convex sets with ν_i-self-concordant barriers F_i respectively. Then
   $$F(x) = \sum_{i=1}^{m} F_i(x_i)$$
   is $(\sum_{i=1}^{n} \nu_i)$-self-concordant on $\mathcal{C}_1 \times \ldots \times \mathcal{C}_m$ (Proof, see [52, Proposition 2.3.1(iii)]).

3. **Compositions with affine function.**
   Let $F : \operatorname{int} \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be ν-self-concordant, $\mathcal{A} : \mathbb{R}^p \to \mathbb{R}^n$ such that $\mathcal{A}(y) = By + c$ for $B \in \mathbb{R}^{n,p}$ and $c \in \mathbb{R}^n$. Assume $\mathcal{A}(\mathbb{R}^p) \bigcap \mathcal{C} \neq \emptyset$. Define
   $$\mathcal{C}^+ = \mathcal{A}^{-1}(\mathcal{C}) = \{y \in \mathbb{R}^p : \mathcal{A}(y) \in \mathcal{C}\} \subseteq \mathbb{R}^p.$$
   Then $\tilde{F} : \mathcal{C}^+ \to \mathbb{R}$ defined as
   $$\tilde{F}(y) = F(\mathcal{A}(y))$$
   is ν-self-concordant on $\operatorname{dom} \tilde{F} = \mathcal{C}^+$ (Proof, see [52, Proposition 2.3.1(i)]).

4. **Restriction to affine subspace.**
   Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be self-concordant and $\mathcal{L} = \{x \in \mathbb{R}^n : Ax = b\}$ an affine subspace, where $A \in \mathbb{R}^{m,n}$, $b \in \mathbb{R}^m$ and $m < n$. Then the restriction of F to $\mathcal{L}$, which we denote by $F|_{\mathcal{L}}$, is self-concordant on its domain. The restriction is understood analogously to the restriction of self-concordant functions in Section 2.2.2.

5. **Conic hull.**
   Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a closed convex set with ν-self-concordant barrier F. Then there exists $\theta > 0$ and $\tau > 0$ such that
   $$\tilde{F}(x, t) = \theta \left[F(x/t) - \tau\nu \log(t)\right]$$
   is 25ν-self-concordant[2] for the conic hull of $\mathcal{C}$,
   $$\operatorname{cone}(\mathcal{C}) := \left\{(x, t) \in \mathbb{R}^n \times \mathbb{R}_{++} : \frac{x}{t} \in \mathcal{C}\right\}.$$

6. **Partial minimization.**
   Let $F : \operatorname{int} \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$, such that $(x, y) \mapsto F(x, y)$, be ν-self-concordant and bounded from below. We assume that $\mathcal{C}$ does not contain a straight line. Then the partial minimization of F with respect to y
   $$G(x) = \inf_{y \in \mathcal{Q}(x)} F(x, y),$$
   where $\mathcal{Q}(x) = \{y : (x, y) \in \mathcal{C}\}$, is ν-self-concordant on $\operatorname{dom} G = \{x : \mathcal{Q}(x) \neq \emptyset\}$ (Proof [50, Theorem 3]).

[2] In fact, the authors show that, with a suitable choice of θ and τ, one obtains a $\gamma\,\nu$-self-concordant barrier for $\operatorname{cone}(\mathcal{C})$, where γ is a constant that is situated between 9 and 25.

We have seen above several examples of self-concordant barriers for convex sets, and also how certain (convexity-preserving) transformations of convex sets affect their barriers.

Below we will quote a theoretically highly important result which states that in principle any convex set admits a self-concordant barrier whose parameter is proportional to the dimension of the underlying space.

Theorem 2.4.6. *Every open convex set $\mathcal{C} \subset \mathbb{R}^n$ containing no straight line admits a ν-self-concordant barrier Φ (*universal barrier*), with $\nu = \mathcal{O}(n)$, defined by*

$$\Phi(x) = \mathcal{O}(1) \cdot \log u(x),$$

where

$$u(x) = \mathrm{vol}_n(\mathcal{C}^0(x)),$$

vol_n *is the n-dimensional Lebesgue measure and $\mathcal{C}^0(x)$ is the polar set of $\mathcal{C}$ centered at x, i.e.*

$$\mathcal{C}^0(x) = \{y \in \mathbb{R}^n : \langle z - x, y\rangle \leq 1, \forall z \in \mathcal{C}\}.$$

Proof. [52, Theorem 2.5.1]. □

Note that the value of $\mathcal{O}(1)$ in Theorem 2.4.6 is in general unknown (see Section 3.2).

Conditions for self-concordance

There are some necessary and sufficient conditions for self-concordance which we will present below.

Theorem 2.4.7. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a closed convex set and $F : \mathrm{int}\,\mathcal{C} \to \mathbb{R}$ self-concordant. Then F is a ν-self-concordant barrier for $\mathcal{C}$ if and only if for any $x, y \in \mathrm{int}\,\mathcal{C}$ we have*

$$F(y) \geq F(x) + \nabla F(x)^T(y - x) + \nu\omega_* \left(\frac{1}{\nu}\nabla F(x)^T(y - x)\right).$$

Proof. [46, Theorem 4.2.4(2)]. □

The following result links self-concordant barriers with certain "well-behaved" functions that are called *compatible* with the barrier F. Before we need to introduce the concept of convex cones.

Definition 2.4.8. *A* cone *is a set $\mathcal{K} \subseteq \mathcal{E}$, such that*

$$\lambda\, x \in \mathcal{K},\ \forall\, x \in \mathcal{K}, \forall\, \lambda \geq 0,$$

i.e. the whole ray going through a point $x \in \mathcal{K}$ also belongs to $\mathcal{K}$.

Any closed convex cone such that $\mathcal{K} \bigcap (-\mathcal{K}) = \{0\}$ and $\mathrm{int}\,\mathcal{K} \neq \emptyset$ induces an ordering relation in $\mathcal{E}$ in the following way: for $x, y \in \mathcal{E}$ we denote

$$x \succeq_{\mathcal{K}} y \qquad \Leftrightarrow \qquad (x - y) \in \mathcal{K}\ \text{ (or: } x - y \succeq_{\mathcal{K}} 0),$$

and analogously

$$x \succ_{\mathcal{K}} y \qquad \Leftrightarrow \qquad (x - y) \in \mathrm{int}\,\mathcal{K}\ \text{ (or: } x - y \succ_{\mathcal{K}} 0).$$

Definition 2.4.9. *Let $\mathcal{C}_1 \subseteq \mathcal{E}_1$ be a closed convex set with ν_1-self-concordant barrier $F : \mathcal{C}_1 \to \mathbb{R}$. Let $\xi : \mathcal{C}_1 \to \mathcal{E}_2$, $\mathcal{K} \subseteq \mathcal{E}_2$ a closed convex cone and $\beta \geq 1$. We call ξ β-*compatible *with F if ξ is concave with respect to $\mathcal{K}$ (i.e. $D^2\xi(x)[h,h] \preceq_{\mathcal{K}} 0$ for all $x \in \operatorname{int}\operatorname{dom}\xi$ and for all $h \in \mathcal{E}_1$) and*

$$D^3\xi(x)[h,h,h] \preceq_{\mathcal{K}} -3\beta\, D^2\xi(x)[h,h]\, ||h||_x.$$

Definition 2.4.10. *A direction $h \in \mathbb{R}^n$ is called* recession direction *for $\mathcal{C} \subseteq \mathcal{E}$ if for all $x \in \mathcal{C}$ we have*

$$x + \lambda h \in \mathcal{C},\ \forall\, \lambda \geq 0.$$

Theorem 2.4.11. *Let $\xi : \mathcal{C}_1 \subseteq \mathcal{E}_1 \to \mathcal{E}_2$ be β-compatible with the ν_1-self-concordant barrier $F : \mathcal{C}_1 \to \mathbb{R}$ (with respect to the closed convex cone $\mathcal{K} \subseteq \mathcal{E}_2$). Let the closed convex set $\mathcal{C}_2 \subseteq (\mathcal{E}_2 \times \mathcal{E}_3)$ admit the ν_2-self-concordant barrier ϕ and let us assume that $\mathcal{K} \times \{0\} \subset \mathcal{E}_2 \times \mathcal{E}_3$ only contains recession directions of $\mathcal{C}_2$. Then*

$$\Psi(x,z) := \phi(\xi(x), z) + \beta^3 F(x)$$

is a $(\nu_2+\beta^3\nu_1)$-self-concordant barrier for its domain $\{(x,y,z) : \xi(x) \succeq_{\mathcal{K}} y, (y,z) \in \mathcal{C}_2\}$.

Proof. [47, Theorem 3]. □

Nesterov [47] used the above result to prove self-concordance of barriers for the following convex sets:

1. $\mathcal{K}_\alpha = \{(x,z) \in \mathbb{R}^2_+ \times \mathbb{R} : x_1^\alpha x_2^{1-\alpha} \geq |z|\}$, where $0 \leq \alpha \leq 1$, with the 4-self-concordant barrier

$$F(x,z) = -\log(x_1^{2\alpha}x_2^{2-2\alpha} - z^2) - \log(x_1) - \log(x_2).$$

2. $\mathcal{K}_{\exp} = \{(x,y,z) \in \mathbb{R} \times \mathbb{R}^2_+ : \exp(x/z) \leq y/z\}$ with the 3-self-concordant barrier

$$F(x,y,z) = -\log(z\log(y/z) - x) - \log(y) - \log(z).$$

3. The hypograph of the geometric mean $\prod_{i=1}^n x_i^{\alpha_i}$, where $x \in \mathbb{R}^n_+$, $\alpha \geq 0$, $\sum_{i=1}^n \alpha_i = 1$ with the $(n+1)$-self-concordant barrier

$$F(x,z) = -\log\left(\prod_{i=1}^n x_i^{\alpha_i} - z\right) - \sum_{i=1}^n \log(x_i).$$

Tracing the central path

Let us come back now to the central path, that is defined as the trajectory of solutions $\{x(t)\}$ of the centering problems (P_t). For points on the central path we can bound the distance to an optimal solution for (2.32). We have the following result.

Theorem 2.4.12. *For any $t > 0$ we have*

$$c^T(x(t) - x^*) \le \frac{\nu}{t}.$$

Moreover, if $\delta_{x,t} \le \beta < 1$, then

$$c^T(x - x^*) \le \frac{1}{t}\left(\nu + \kappa(\beta,\nu)\right).$$

where $\kappa(\beta,\nu) = \frac{(\beta+\sqrt{\nu})\beta}{1-\beta}$.

Proof. This proof is essentially the same as [46, Theorem 4.2.7], the only difference being additionally we have to take into account the linear equality constraints.

In order to prove the first inequality, we use the fact $x(t)$ is characterized by the optimality conditions for (P_t), i.e. $\exists\,\lambda$ such that

$$\begin{aligned}\nabla f_t(x) &= -A^T\lambda,\\ Ax &= b.\end{aligned}$$

Since $\nabla f_t(x) = t\,c + \nabla F(x)$, we get $c = -\frac{1}{t}(\nabla F(x) + A^T\lambda)$ for some λ. It follows (using Lemma 2.4.5)

$$\begin{aligned}c^T(x(t) - x^*) &= -\frac{1}{t}(\nabla F(x(t)) + A^T\lambda)^T(x(t) - x^*)\\ &= -\frac{1}{t}\nabla F(x(t))^T(x(t) - x^*) - \frac{1}{t}\lambda^T\underbrace{A\,(x(t) - x^*)}_{=0}\\ &\le \frac{\nu}{t}.\end{aligned}$$

For the second inequality we get

$$\begin{aligned}tc^T(x - x(t)) &= (\nabla f_t(x) - \nabla F(x))^T(x - x(t))\\ &= (\nabla f_t(x) + A^T\lambda - \nabla F(x) - A^T\lambda)^T(x - x(t))\\ &= (\nabla f_t(x) + A^T\lambda - \nabla F(x))^T(x - x(t)) - \lambda^T\underbrace{A(x - x(t))}_{=0}\\ &\le (\underbrace{||\nabla f_t(x) + A^T\lambda||_x^*}_{=\delta_{x,t}} + \underbrace{||\nabla F(x)||_x^*}_{\le\sqrt{\nu}}) \cdot \underbrace{||x - x(t)||_x}_{\le\omega_*'(\delta_{x,t})}\\ &\le (\delta_{x,t} + \sqrt{\nu})\frac{\delta_{x,t}}{1 - \delta_{x,t}}\\ &\le \frac{(\beta + \sqrt{\nu})\beta}{1 - \beta} = \kappa(\beta,\nu).\end{aligned}$$

The desired result follows from a combination with the first inequality. □

That means if x is close enough to $x(t)$ (say, $\beta < 0.1$) and t is large enough, then we have a good bound on the difference between the current and the optimal objective value.

To answer the question how to update t, we need the following result that describes the so-called short-step algorithm.

Theorem 2.4.13. *Let F be a ν-self-concordant barrier for $\mathcal{C}$, $\beta < \epsilon_0 = \frac{3-\sqrt{5}}{2}$, $0 < \gamma \leq \frac{\sqrt{\beta}}{1+\sqrt{\beta}} - \beta$ and $x \in \operatorname{int} \mathcal{C}$ such that $Ax = b$ and $\delta_{x,t} \leq \beta$. We define*

$$\begin{aligned} t^+ &= t\left(1 + \frac{\gamma}{\beta + \sqrt{\nu}}\right), \\ x^+ &= x + \Delta x(t^+), \end{aligned}$$

where $\Delta x(t^+)$ is the solution of (2.34) with the new duality measure t^+. Then we have

$$\delta_{x^+,t^+} \leq \beta.$$

Proof. Again, this proof is very close to the one in [46, Theorem 4.2.8], but we have to take additionally into account the linear equalities. Let $x \in \operatorname{int} \mathcal{C}$ such that $Ax = b$. We denote by $\Delta x(t)$ the Newton direction at x towards $x(t)$ and by $\Delta x(t^+)$ the Newton direction at x towards $x(t^+)$. Similar to Section 2.3 we can consider the unconstrained problems where some of the variables x have been eliminated (see (2.24)), i.e. we get as the objective function

$$f_{x,t}(y) = t\, c^T(By + x) + F(By + x),$$

relative to $x \in \operatorname{int} \mathcal{C} \bigcap \mathcal{L}$ and $t > 0$, where $B \in \mathbb{R}^{n,n-m}$ is any elimination matrix such that $\operatorname{range}(B) = \operatorname{null}(A)$. The gradient of $f_{x,t}$ becomes then

$$\nabla f_{x,t}(y) = t\, B^T c + B^T \nabla F(By + x),$$

and at the point $y = 0$ this gives

$$\nabla f_{x,t}(0) = t\, B^T c + B^T \nabla F(x). \tag{2.35}$$

According to Lemma 2.3.3, we have $\delta_{y_0,t} = \delta_{x,t}$ and $\delta_{y_0,t^+} = \delta_{x,t^+}$, where $\delta_{x,t} = ||\Delta x(t)||_x$, $\delta_{x,t^+} = ||\Delta x(t^+)||_x$ and

$$\begin{aligned} \delta_{y_0,t} &= ||\nabla f_{x,t^+}(0)||^*_{y=0}, \\ \delta_{y_0,t^+} &= ||\nabla f_{x,t^+}(0)||^*_{y=0}, \end{aligned}$$

where $\Delta y_0(t)$ denotes the Newton direction at $y = 0$ towards the minimizer of $f_{x,t}(y)$ (respectively for t^+).

On the other hand, we have in view of the definition of t^+

$$\begin{aligned} \delta_{y_0,t^+} &= ||\nabla f_{x,t^+}(0)||^*_{y=0} \\ &\overset{(2.35)}{=} ||t^+ B^T c + B^T \nabla F(x)||^*_{y=0} \\ &= ||tB^T c + B^T \nabla F(x) + t \cdot \frac{\gamma}{\beta + \sqrt{\nu}} \cdot B^T c||^*_{y=0} \\ &\leq \underbrace{||tB^T c + B^T \nabla F(x)||^*_{y=0}}_{\overset{(2.35)}{=} ||\nabla f_{x,t}(0)||^*_{y=0} = \delta_{y_0,t}} + \gamma \frac{t \cdot ||B^T c||^*_{y=0}}{\beta + \sqrt{\nu}} \\ &\leq \delta_{y_0,t} + \gamma \frac{t \cdot ||B^T c||^*_{y=0}}{\beta + \sqrt{\nu}}. \end{aligned}$$

Using the definition of the gradient of $f_{x,t}$ at $y = 0$ (2.35), we see

$$\begin{aligned} t \cdot \|B^T c\|^*_{y=0} &= \|\nabla f_{x,t}(0) - B^T \nabla F(x)\|^*_{y=0} \\ &\leq \underbrace{\|\nabla f_{x,t}(0)\|^*_{y=0}}_{=\delta_{y_0,t}=\delta_{x,t}\leq\beta} + \|\underbrace{B^T\nabla F(x)}_{=\nabla f_x(0)}\|^*_{y=0} \\ &\leq \beta + \|\nabla f_x(0)\|^*_{y=0}. \end{aligned}$$

Since $f_x(y) = F(By + x)$ is a ν-self-concordant barrier, we have in view of Definition 2.4.4 that $\|\nabla f_x(0)\|^*_{y=0} \leq \sqrt{\nu}$. We conclude

$$t \cdot \|B^T c\|^*_{y=0} \leq \beta + \sqrt{\nu}.$$

It follows, using Lemma 2.3.3 and the above observation, that

$$\delta_{x,t^+} = \delta_{y_0,t^+} \leq \delta_{y_0,t} + \gamma = \delta_{x,t} + \gamma \leq \frac{\sqrt{\beta}}{1+\sqrt{\beta}} < 1.$$

This means the point x is not only close to $x(t)$, but also close to $x(t^+)$ in the sense that $\delta_{x,t^+} < 1$.

Using Theorem 2.3.5, we get

$$\delta_{x^+,t^+} \leq \left(\frac{\delta_{x,t^+}}{1-\delta_{x,t^+}}\right)^2 = \left(\omega_*'(\delta_{x,t^+})\right)^2.$$

We conclude, using monotonicity of ω_*' and Lemma 2.2.14,

$$\sqrt{\delta_{x^+,t^+}} \leq \omega_*'(\delta_{x,t^+}) \leq \omega_*'\left(\frac{\sqrt{\beta}}{1+\sqrt{\beta}}\right) = \omega_*'(\omega'(\sqrt{\beta})) = \sqrt{\beta},$$

which finishes the proof. □

The above theorem says that once we are close to the central path, we can update t in a moderate way so that one full Newton step brings us back to the initial proximity.

We see that the increase in t is linear and that it only depends on the barrier parameter ν (that is given) and algorithm parameters β and γ that we choose. Ideally we would like to choose γ as large as possible as this guarantees that the updating coefficient for t is large. In Figure 2.3 we see the upper bound on γ as a function of β. We can verify that a feasible (and reasonable) choice of the parameters is $\beta = \frac{1}{9}$ and $\gamma = \frac{5}{36}$ (suggested e.g. in [46, (4.2.22)]).

The following result summarizes the polynomial complexity of the so-called short-step path-following algorithm for solving (2.32).

Theorem 2.4.14. *Let F be a ν-self-concordant barrier for $\mathcal{C} \subseteq \mathbb{R}^n$. Let $t_0 > 0$, $\epsilon > 0$ and $\beta < \frac{3-\sqrt{5}}{2}$. Choose $0 < \gamma \leq \frac{\sqrt{\beta}}{1+\sqrt{\beta}} - \beta$. Let $x_0 \in \operatorname{int}\mathcal{C} \bigcap \{x : Ax = b\}$ such that*

$$\delta_{x_0,t_0} \leq \beta.$$

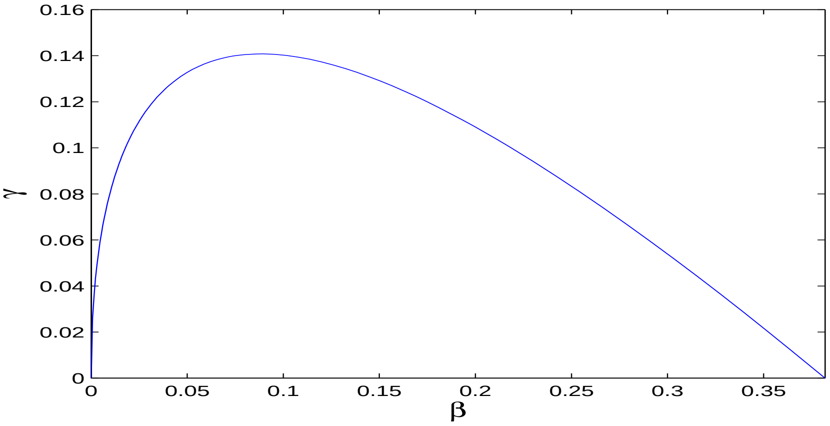


Figure 2.3: The upper bound on γ: $\gamma \leq \frac{\sqrt{\beta}}{1+\sqrt{\beta}} - \beta$.

Then it is possible to compute a point x_N *such that*

$$\langle c, x_N - x^* \rangle \leq \epsilon$$

in no more than

$$N = \mathcal{O}\left(\sqrt{\nu} \log\left(\frac{\nu}{t_0 \epsilon}\right)\right)$$

iterations.

Proof. In view of Theorem 2.4.13 we get recursively

$$t_k = \left(1 + \frac{\gamma}{\beta + \sqrt{\nu}}\right) t_{k-1} = \left(1 + \frac{\gamma}{\beta + \sqrt{\nu}}\right)^k t_0.$$

In combination with Theorem 2.4.12 we get that

$$c^T(x - x^*) \leq \epsilon$$

if

$$t \geq \frac{\kappa(\beta, \nu)}{\epsilon},$$

which in turn is satisfied if

$$\left(1 + \frac{\gamma}{\beta + \sqrt{\nu}}\right)^k t_0 \geq \frac{\kappa(\beta, \nu)}{\epsilon}.$$

We get

$$k \geq \log\left(\frac{\kappa(\beta, \nu)}{t_0 \epsilon}\right) / \log\left(1 + \frac{\gamma}{\beta + \sqrt{\nu}}\right).$$

Let us denote $\tau = \frac{\gamma}{\beta+\sqrt{\nu}}$. We have then $0 \leq \tau \leq \frac{\gamma}{\sqrt{\nu}} \leq \gamma < 0.15$ (see Figure 2.3). Since $\log(1 + \tau)$ is concave and because $\log(1 + 0.15) > \frac{0.15}{2}$, we conclude that

$\log(1+\tau) \geq \frac{\tau}{2}$ for all $0 \leq \tau \leq 0.15$. It follows that the stopping criterion is met after at most

$$k = \frac{2}{\tau}\log\left(\frac{\kappa(\beta,\nu)}{t_0\epsilon}\right) = \mathcal{O}\left(\sqrt{\nu}\log\left(\frac{\nu}{t_0\epsilon}\right)\right)$$

iterations.

□

Algorithm 3 Short-step path-following interior-point method

```
Input: $A \in \mathbb{R}^{m,n}$ with full row rank, $b \in \mathbb{R}^m$, $F$ $\nu$-self-concordant barrier for $\mathcal{C}$
Initialize: $k = 0$, $t_0 > 0$, $\epsilon > 0$, $\beta < \frac{3-\sqrt{5}}{2}$. Choose $0 < \gamma \leq \frac{\sqrt{\beta}}{1+\sqrt{\beta}} - \beta$. Define
  $\kappa(\beta,\nu) = \left(\nu + \frac{(\beta+\sqrt{\nu})\beta}{1-\beta}\right)$. Let $x_0 \in \operatorname{int}\mathcal{C}\bigcap\{x : Ax = b\}$ such that
      $\delta_{x_0,t_0} \leq \beta$,
  repeat
    1) update target $t_{k+1} = t_k\left(1 + \frac{\gamma}{\beta+\sqrt{\nu}}\right)$
    2) compute Newton direction $\Delta x(t_{k+1})$ towards $x(t_{k+1})$
    3) update iterate $x_{k+1} = x_k + \Delta x(t_{k+1})$
    4) $k := k + 1$
  until $\kappa(\beta,\nu) \leq \epsilon \cdot t_k$
```

Initialization

Note that we have assumed the availability of an initial point x_0 close to the central path for some value of $t_0 > 0$. In order to find such a point we have two options.

1. We can apply a damped Newton method to solve the initial centering problem for any value of t_0.. The complexity is

$$\mathcal{O}\left(F(x_0) - F(x^*)\right)$$

iterations, $x_0 \in \operatorname{int}\mathcal{C}\bigcap\mathcal{L}$ is any strictly feasible starting point of the auxiliary process (cf. Theorem 2.3.7).

2. If $\mathcal{C}\bigcap\mathcal{L}$ is bounded, we can use an auxiliary path-following scheme to find the analytic center of $\mathcal{C}\bigcap\{Ax = b\}$. The complexity is

$$\mathcal{O}\left(\sqrt{\nu}\log\left(\nu \cdot ||\nabla F(x_0)||^*_{x_F}\right)\right),$$

where $x_0 \in \operatorname{int}\mathcal{C}\bigcap\mathcal{L}$ is any strictly feasible starting point of the process (i.e. $\delta_{x_0,0} > \beta$) and x_F denotes the analytic center, i.e.

$$x_F = \operatorname{argmin}_{x:Ax=b} F(x),$$

(for a detailed discussion we refer to [46, Theorem 4.2.11]).

It is not possible to compare directly the complexities of both initialization procedures, but Nesterov argues at the end of Section 4.2.5 in [46] that the second approach typically is superior.

Large updates of t

We can also update t in a more aggressive manner by choosing a parameter $\theta > 1$ (that could possibly be depending on the current iterate) and update t as

$$t^+ = \theta\, t.$$

It is clear that if $\theta \gg 1 + \frac{\gamma}{\beta+\sqrt{\nu}}$, then we can expect fewer updates of t than for the short-step algorithm. Indeed, for $\beta < 1$ we have according to Theorem 2.4.12

$$c^T(x - x^*) \leq \frac{\kappa(\beta,\nu)}{t}.$$

If we desire an optimality gap of ϵ, i.e. $c^T(x - x^*) \leq \epsilon$, then this is guaranteed by finding a point x such that $\delta_{x,t} \leq \beta$ and $\frac{\kappa(\beta,\nu)}{t} \leq \epsilon$. The latter condition is satisfied after at most

$$N_{out} \leq \frac{\log(\kappa(\beta,\nu)) - \log(t_0\epsilon)}{\log(\theta)} = \mathcal{O}\left(\log\left(\frac{\nu}{t_0\,\epsilon}\right)\right) \tag{2.36}$$

outer iterations.

On the other hand, there are some drawbacks as compared to Algorithm 3:

1. If θ is large, we cannot apply Theorem 2.4.13 and therefore we cannot guarantee that one full Newton restores proximity to the central path, i.e. δ_{x^+,t^+} is not less than or equal to β.

2. We cannot be sure that a *full* Newton step is even feasible.

To overcome these two drawbacks one has to do several damped Newton steps towards the new target point $x(t^+)$. The number of damped Newton steps can be bounded in the following way. Assume $x \in \operatorname{int}\mathcal{C}$ such that $Ax = b$ and $\delta_{x,t} \leq \beta$ (that is x is close to the point $x(t)$ on the central path). If we update t to $\theta\, t$ and we impose additionally that $\beta \leq \frac{1}{4}$, we have the following bound on the functional difference to the new target point $x(t^+)$:

$$f_{t^+}(x) - f_{t^+}(x(t^+)) \leq \theta(\nu + \sqrt{\nu}).$$

The above inequality is proved e.g. by Renegar in [55, Section 2.4.3] for the case where no equality constraints are present. The same inequality applies to our setting because we can consider again the function $f_{x,t^+}(y) = f_{t^+}(By + x)$ that is parametrized by $x \in \operatorname{int}\mathcal{C}\bigcap\mathcal{L}$ and some matrix $B \in \mathbb{R}^{n,n-m}$ such that $\operatorname{range}(B) = \operatorname{null}(A)$.

That means if we do damped Newton steps with step size $\alpha = \frac{1}{1+\delta_{x,t^+}}$ (and δ_{x,t^+} denotes the Newton decrement in each inner iteration for Newton directions towards $x(t^+)$), then in accordance with Theorem 2.3.6 it takes no more than

$$N_{in} \leq \frac{\theta(\nu + \sqrt{\nu})}{\omega(\beta)} \tag{2.37}$$

inner iterations to compute a point $x^+ \in \operatorname{int} C$ such that $Ax^+ = b$ and $\delta_{x^+,t^+} \leq \beta$.

Algorithm 4 Long-step path-following interior-point method

```
Input: $A \in \mathbb{R}^{m,n}$ with full row rank, $b \in \mathbb{R}^m$, $F$ $\nu$-self-concordant barrier for $C$
Initialize: Choose parameters $\epsilon > 0$, $0 < \beta \leq \frac{1}{4}$, $\theta > 1$, define $\kappa(\beta, \nu) = \nu + \frac{(\beta+\sqrt{\nu})\beta}{1-\beta}$. Initialize $k = 0$, $i = 0$, $t_0 > 0$ and $x^{(0)} \in \operatorname{int} C \bigcap \{x : Ax = b\}$ such that $\delta_{t_0}(x^{(0)}) \leq \beta$.
  while $\epsilon \cdot t_k < \kappa(\beta, \nu)$ do
    1) compute Newton direction $\Delta x_{t_k}^{(i)}$ from (2.34)
    2) compute Newton decrement $\delta_{t_k}(x^{(i)}) = ||\Delta x_{t_k}^{(i)}||_{x^{(i)}}$
    while $\delta_{t_k}(x^{(i)}) > \beta$ do
      a) $x^{(i+1)} := x^{(i)} + \alpha \cdot \Delta x_{t_k}^{(i)}$, where $\alpha$ is a suitable step length
      b) $i := i + 1$
      c) compute Newton direction $\Delta x_{t_k}^{(i)}$ from (2.34)
      d) compute Newton decrement $\delta_{t_k}(x^{(i)}) = ||\Delta x_{t_k}^{(i)}||_{x^{(i)}}$
    end while
    4) update $t_{k+1} := \theta \cdot t_k$
    5) update $k := k + 1$
  end while
```

Note that $\alpha = \frac{1}{1+\delta}$ with $\delta = \delta_{t_k}(x^{(i)})$, is a feasible step length in step a) of the inner loop.

To describe the algorithm formally, let us introduce some notation. We denote by $\Delta x_{t_k}^{(i)}$ the Newton direction at the point $x^{(i)}$ towards the target point $x(t_k)$ on the central path. Further $\delta_{t_k}(x^{(i)}) = ||\Delta x_{t_k}^{(i)}||_{x^{(i)}}$ is the Newton decrement of $\Delta x_{t_k}^{(i)}$ with respect to the current iterate $x^{(i)}$.

In principle the centering accuracy β can be chosen more loosely. Since the long-step algorithm is only using Theorem 2.4.12 (and not Theorem 2.4.13), it is sufficient to take $\beta < 1$. However, note that for β close to 1, $\kappa(\beta, \nu)$ increases and therefore also the bound on the number of outer iterations (see (2.36)).

Theorem 2.4.15. *Algorithm 4 terminates after at most*

$$N \leq \mathcal{O}\left(\nu \log\left(\frac{\nu}{t_0 \epsilon}\right)\right)$$

iterations with a point x_N such that

$$\langle c, x_N - x^* \rangle \leq \epsilon.$$

Proof. As we have seen, the number outer iterations (2.36) is bounded by

$$N_{out} \leq \frac{\log(\kappa(\beta, \nu)) - \log(t_0 \epsilon)}{\log(\theta)} = \mathcal{O}\left(\log\left(\frac{\nu}{t_0 \epsilon}\right)\right).$$

On the other hand when updating t to θt, the number of inner iterations to generate a central point with accuracy β is according to (2.37) given by

$$N_{in} \leq \frac{\theta(\nu + \sqrt{\nu})}{\omega(\beta)} = \mathcal{O}(\nu).$$

□

2.5 Conic optimization

We have seen in Section 2.4 that general convex optimization problems can be solved using interior-point methods, provided that a ν-self-concordant barrier for the feasible set (and possibly the epigraph of a nonlinear objective function) is available. In this section we consider convex problems in conic format, which allows the design of primal-dual interior-point methods.

2.5.1 From convex to conic optimization

Convex optimization problems in conic form are problems of the form (2.32), where the closed convex set $\mathcal{C} \subseteq \mathbb{R}^n$ is a *cone* (see Definition 2.4.8). If we take for example $\mathcal{C} = \mathbb{R}^n_+$, then the corresponding conic problem is simply a primal linear program.

Definition 2.5.1. *We define the* dual cone $\mathcal{K}^* \subseteq \mathcal{E}^*$ *of* $\mathcal{K} \subseteq \mathcal{E}$ *by*

$$\mathcal{K}^* := \{s \in \mathcal{E}^* : \langle s, x \rangle \geq 0, \ \forall\, x \in \mathcal{K}\}.$$

It is easy to see that $\mathcal{K}^*$ is always closed and convex (even if $\mathcal{K}$ is not).

We can consider now the primal-dual conic pair defined as

$$\boxed{\begin{aligned} (\bar{P}) \ \min_{x}\ & c^T x \\ Ax &= b \\ x &\in \mathcal{K}, \end{aligned}} \qquad \boxed{\begin{aligned} (\bar{D}) \ \max_{y,s}\ & b^T y \\ s + A^T y &= c \\ s &\in \mathcal{K}^*. \end{aligned}}$$

We see that $(\bar{P})$ and $(\bar{D})$ are in fact equivalent to the standard convex format (2.32). Indeed, $(\bar{P})$ is already of the form (2.32), where $\mathcal{C}$ is chosen as the particular convex set $\mathcal{K}$. On the other hand, any convex problem (2.32) can be brought into conic form. Indeed, we see that (2.32) is equivalent to

$$\begin{aligned} \min_{x,t}\ & c^T x \\ (x,t) &\in \operatorname{cone}(C) \\ Ax &= b \\ t &= 1. \end{aligned}$$

That means we have introduced one new variable t and added the linear constraint $t = 1$, fixing this variable immediately to 1.

Conic duality

We have weak duality between the two problems, because for any primal feasible point x and any dual feasible point (y, s) we have

$$c^T x - b^T y = c^T x - x^T A^T y = c^T x - c^T x + s^T x \geq 0.$$

using the definition of the dual cone.

However, we have to be a little careful because it might happen that $\mathcal{K}$ (or $\mathcal{K}^*$) is empty or does not have interior points. To characterize this condition we need the following two notions.

Definition 2.5.2. *A cone is said to be* pointed *if* $\mathcal{K} \bigcap (-\mathcal{K}) = \{0\}$. $\mathcal{K}$ *is called* solid *if* $\operatorname{int} \mathcal{K} \neq \emptyset$.

Since $\mathcal{K}$ is assumed to be closed and convex, we have the following relations related to its dual cone $\mathcal{K}^*$ (see [3, Theorem 2.3.1])

$$\begin{array}{ccc} \mathcal{K} \text{ pointed} & \Leftrightarrow & \mathcal{K}^* \text{ solid} \\ \mathcal{K}^* \text{ pointed} & \Leftrightarrow & \mathcal{K} \text{ solid.} \end{array}$$

Definition 2.5.3. *A cone* $\mathcal{K}$ *is called* proper *if it is closed, convex, pointed and solid.*

We have already seen that between $(\bar{P})$ and $(\bar{D})$ weak duality holds. Under slightly stronger assumptions than in the LP case, we can also guarantee strong duality.

Theorem 2.5.4. *1. Let* $(\bar{P})$ *be bounded from below and strictly feasible, i.e.* $\exists\ x \in \operatorname{int} \mathcal{K}$ *such that* $Ax = b$. *Then* $(\bar{D})$ *is solvable and the optimal values of* $(\bar{P})$ *and* $(\bar{D})$ *are equal.*

2. Let (D) be bounded from above and strictly feasible, i.e. $\exists\ y : (c - A^T y) \in \operatorname{int} \mathcal{K}^*$. *Then* $(\bar{P})$ *is solvable and their optimal values are equal.*

Proof. [3, Theorem 2.4.1]. □

An immediate consequence of the above theorem is that if both $(\bar{P})$ and $(\bar{D})$ are strictly feasible then there exist $x^* \in \mathcal{K}$ such that $Ax^* = b$ and y^* such that $c - A^T y^* \in \mathcal{K}^*$ and strong duality holds, that is

$$c^T x^* = b^T y^*.$$

We see that the strong duality result is more restrictive as opposed to the LP case, where feasibility and boundedness of either the primal or the dual implied feasibility and boundedness of the other problem, even when no strictly feasible point is available. Additionally both optimal values are attained and strong duality holds. For a thorough discussion of this phenomenon, see [3, Section 2.4.1].

Theorem 2.5.4 requires strict feasibility and guarantees only that the optimal values are equal, not that they are attained for *both* problems, i.e. the optimum might be attained only for one of the two problems. In [3] one can find examples of situations where one problem is bounded and feasible (but not strictly feasible) and the dual is infeasible. Similarly it is possible that $(\bar{P})$ is strictly feasible and bounded, but not solvable (in the sense that the optimal value is not attained). These crucial observations have to be kept in mind when dealing with conic problems.

Provided that A has full row rank, we see easily that for $(\bar{P})$ the first three requirements of Assumption 1 (p. 44) are satisfied. Indeed pointedness implies that $\mathcal{K}$ does not contain straight lines (neither does then $\mathcal{K}\bigcap\{x : Ax = b\}$), and $\operatorname{int}\mathcal{K} \neq \emptyset$ is equivalent to saying $\mathcal{K}$ is full-dimensional. In other words, properness of $\mathcal{K}$ is a more compact way of ensuring the first and second part of Assumption 1. On the dual side we see that for any nondegenerate barrier $F(s)$ for $\mathcal{K}^*$ it holds that $\tilde{F}(y) = F(c - A^T y)$ is nondegenerate too if A has full row rank. The last statement can easily be seen when considering the Hessian of $\tilde{F}$, i.e.

$$\nabla^2 \tilde{F}(y) = (-A)\,\nabla^2 F(c - A^T y)\,(-A^T) = A\,\underbrace{\nabla^2 F(c - A^T y)}_{\succ 0}\,A^T.$$

This means the Newton directions in terms of y are defined for any strictly feasible point $y \in \{y : c - A^T y \in \operatorname{int}\mathcal{K}^*\}$.

As we have pointed out earlier, in this thesis we only consider feasible problems. Therefore, we assume that the last two properties in Assumption 1 are satisfied both for $(\bar{P})$ and $(\bar{D})$. In fact, to ensure strong duality, we have to impose additionally *strict* feasibility for the primal and the dual problem.

Logarithmically homogeneous barriers for proper cones

For proper cones we have a special class of barriers, so-called logarithmically homogeneous barriers.

Definition 2.5.5. *Let $\mathcal{K} \subseteq \mathcal{E}$ be a proper cone, $F : \operatorname{int}\mathcal{K} \to \mathbb{R}$ a twice continuously differentiable, convex barrier function. F is called ν-*logarithmically homogeneous *for $\mathcal{K}$ if*

$$F(\tau\, x) = F(x) - \nu\, \log(\tau) \tag{2.38}$$

for any $x \in \operatorname{int}\mathcal{K}$ and any $\tau > 0$.

In Definition 2.5.5 we have not assumed that F is a self-concordant function. The following theorem states that any ν-logarithmically homogeneous function which is also self-concordant is automatically a ν-self-concordant barrier.

Theorem 2.5.6. *Let $F : \operatorname{int}\mathcal{K} \to \mathbb{R}$ be both a self-concordant function with $\operatorname{dom} F = \operatorname{int}\mathcal{K}$ and ν-logarithmically homogeneous for $\mathcal{K}$. Then F is a ν-self-concordant barrier for $\mathcal{K}$.*

Proof. [52, Corollary 2.3.2]. □

As it turns out all the self-concordance-preserving operations presented in Section 2.4 also preserve the property of logarithmic homogeneity if the original barriers exhibited this property ([42, p. 208]).

Assumption 2
In the following, when we speak of a cone $\mathcal{K}$, we assume that it is proper. Similarly, when we speak of ν-self-concordant barriers F for $\mathcal{K}$, then we implicitly assume that F is ν-logarithmically homogeneous.

Direct consequences of the definition of logarithmic homogeneity are the following (they can be found e.g. in [53]).

$$\nabla F(\tau x) = \frac{1}{\tau}\nabla F(x), \tag{2.39}$$

$$\nabla^2 F(\tau x) = \frac{1}{\tau^2}\nabla^2 F(x), \tag{2.40}$$

$$\nabla^2 F(x)\, x = -\nabla F(x), \tag{2.41}$$

$$||x||_x = \sqrt{\nu}, \tag{2.42}$$

$$\langle \nabla F(x), x\rangle = -\nu. \tag{2.43}$$

Definition 2.5.7. [3] *Let F be a ν-logarithmically homogeneous barrier for $\mathcal{K}$. We define its* conjugate *to be*

$$F_*(s) = \sup_{x\in \operatorname{int}\mathcal{K}} \{-s^T x - F(x)\}.$$

The following theorem states that the conjugate of a barrier is in fact a barrier for the dual cone.

Theorem 2.5.8. *Let $F : \operatorname{int}\mathcal{K} \to \mathbb{R}$ be a ν-self-concordant barrier for the proper cone $\mathcal{K} \subseteq \mathcal{E}$. Then $F_*(s)$ is a ν-self-concordant barrier for $\mathcal{K}^* \subseteq \mathcal{E}^*$.*

Proof. [52, Theorem 2.4.4]. □

That means we have complete symmetry between the primal side with F a ν-self-concordant barrier for $\mathcal{K}$ and the dual side with F_* for $\mathcal{K}^*$. All the results listed below for F are therefore applicable for F_* too, where F_* is the conjugate of F.

Example 2.5.9. *Examples of convex cones with ν-self-concordant barriers.*

1. ***Nonnegative orthant:***
Let $\mathcal{E} = \mathbb{R}^n$ and $\mathcal{E} = (\mathbb{R}^n)^* = \mathbb{R}^n$. We consider the cone $\mathcal{K} = \mathbb{R}^n_+$. It can be easily seen that $\mathcal{K}^* = \mathbb{R}^n_+$. We know that $F(x) = -\sum_{i=1}^n \log(x_i)$ is an n-self-concordant barrier for $\mathcal{K}$ and it is clear that F is also n-logarithmically homogeneous. If we compute the conjugate of F, we get $F_*(s) = F(s) - n$.

2. ***Second-order cone:***
Let $\mathcal{E} = \mathbb{R}^{n+1}$ and $\mathcal{E} = (\mathbb{R}^{n+1})^* = \mathbb{R}^{n+1}$. We consider the second-order cone (Lorentz cone) $\mathcal{K} = \mathbb{L}^n$. It turns out that $\mathcal{K}^* = \mathcal{K}$. We have seen that $F(x) = -\log(t^2 - ||x||_2^2)$ is a 2-self-concordant barrier for $\mathcal{K}$. We can compute $F_*(s) = F(s) + 2\log(2) - 2$.

3. ***Cone of positive semidefinite matrices:***
Let $\mathcal{E} = \mathcal{S}^n$ the set of symmetric $n \times n$ matrices and $\mathcal{E} = (\mathcal{S}^n)^* = \mathcal{S}^n$. We consider the cone $\mathcal{K} = \mathcal{S}^n_+ = \{X \in \mathcal{S} : X \succeq 0\}$. It turns out $\mathcal{K}^* = \mathcal{K}$. As we have seen earlier, $F(X) = -\log(\det(X))$ is an n-self-concordant barrier for $\mathcal{K}$. One can also compute $F_*(S) = F(S) - n$.

[3] Note that this definition is slightly different from the standard definition of conjugate functions in convex analysis, where the conjugate is defined as $\max_{x\in\operatorname{int}\mathcal{K}}\{s^T x - F(x)\}$ (see e.g. [56]).

The details can be found e.g. in [53].

Let $x \in \operatorname{int} \mathcal{K}$. Then we have the following identities linking F and F_*, that can be found for example in [53, (2.7)-(2.12)]:

$$-\nabla F(x) \in \operatorname{int} K^* = \operatorname{dom} F_*, \qquad -\nabla F_*(x) \in \operatorname{int} K = \operatorname{dom} F, \tag{2.44}$$

$$F_*(-\nabla F(x)) = -\nu - F(x), \tag{2.45}$$

$$\nabla F_*(-\nabla F(x)) = -x, \qquad \nabla F(-\nabla F_*(s)) = -s, \tag{2.46}$$

$$\nabla^2 F_*(-\nabla F(x)) = \nabla^2 F(x)^{-1}, \qquad \nabla^2 F(-\nabla F_*(s)) = \nabla^2 F_*(s)^{-1}. \tag{2.47}$$

2.5.2 Conic optimization over symmetric cones

In Section 2.4 we have presented the basic path-following methods for convex optimization problems, assuming that a ν-self-concordant barrier for the feasible set $\mathcal{C}$ is available. We have seen that the iterates follow the central path defined by the centering problem

$$\begin{aligned} \min\ & t{\cdot}c^T x + F(x) \\ Ax &= b. \end{aligned} \tag{2.48}$$

In the conic setting $\mathcal{C}$ is replaced by a proper cone $\mathcal{K}$ and F is additionally assumed to be logarithmically homogeneous. As in the non-conic case (Section 2.4) the optimal solution for (2.48) is given by the optimality conditions (2.33). If we define for $t > 0$ the new point $y(t) = \frac{\lambda(t)}{t}$, then the optimality conditions (2.33) can be decomposed into: $x(t)$ is optimal for (2.48) if and only if $\exists\ (y(t), s(t))$ such that $s(t) \in \operatorname{int} \mathcal{K}^*$ and

$$\begin{aligned} Ax(t) &= b, \\ s(t) + A^T y(t) &= c, \\ \nabla F(x(t)) + ts(t) &= 0. \end{aligned} \tag{2.49}$$

Similarly, we can look at the dual centering problem

$$\begin{aligned} \max\ & t \cdot b^T y {-} F_*(s) \\ c - A^T y &= s, \end{aligned} \tag{2.50}$$

where F_* is the conjugate of F. As it turns out, $(y(t), s(t))$ with $s(t) \in \operatorname{int} \mathcal{K}^*$ is optimal for (2.50) if and only if $\exists\ x(t) \in \operatorname{int} \mathcal{K}^*$ such that

$$\begin{aligned} Ax(t) &= b, \\ s(t) + A^T y(t) &= c, \\ \nabla F_*(s(t)) + tx(t) &= 0. \end{aligned} \tag{2.51}$$

In view of (2.46) the two systems (2.49) and (2.51) are equivalent.

Unfortunately, the situation changes when we want to trace the primal-dual central path. Assume we have a primal-dual strictly feasible point that is close to a point on the primal-dual central path, that is defined by (2.49) or (2.51). If we update the duality measure from t to some $t^+ > t$, then we have two possibilities of defining the Newton direction towards the new target point on the central path, one using the primal barrier F, another one using F_*. It turns out that in general the directions obtained by the two approaches are not the same (for a more detailed discussion we refer to [42, Section 3.2]). If we want the primal-dual path-following directions to be completely symmetric in terms of F and F_*, we need to introduce the concept of *symmetric cones.*

Definition 2.5.10. *A proper cone $\mathcal{K}$ is called* symmetric *if it is*

1. *self-dual*[4]*: $\mathcal{K} = \mathcal{K}^*$,*

2. *homogeneous (for any $x_1, x_2 \in \mathcal{K}$ there exists a linear operator $\mathcal{A}$ with $\mathcal{A}\mathcal{K} = \mathcal{K}$ such that $\mathcal{A}x_1 = x_2$).*

In [28] Güler summarizes the classification of symmetric cones. Every symmetric cone can be uniquely decomposed into a direct product of irreducible symmetric cones, of which only 5 exist. As it turns out, every symmetric cone admits a so-called self-scaled barrier.[5]

Definition 2.5.11. *A ν-self-concordant barrier F for $\mathcal{K} \subseteq \mathcal{E}$ is called* self-scaled *if for any $x \in \operatorname{int} \mathcal{K}$ and $y \in \operatorname{int} \mathcal{K}$ we have*

$$\nabla^2 F(y)x \in \operatorname{int} \mathcal{K}^*$$

and

$$F_*(\nabla^2 F(y)x) = F(x) - 2F(y) - \nu.$$

An important consequence of the above definition is the existence of a so-called *scaling point.*

Theorem 2.5.12. *Let F be a self-scaled barrier for $\mathcal{K}$, $x \in \operatorname{int} \mathcal{K}$ and $s \in \operatorname{int} \mathcal{K}^*$. Then there exists a unique scaling point $w \in \operatorname{int} \mathcal{K}$ such that*

$$\nabla^2 F(w)x = s.$$

Proof. [53, Theorem 3.2]. □

Among the 5 classes of irreducible symmetric cones, the following three are by far the most important ones, since highly efficient implementations of interior-point methods over these cones exist.

1. $\mathcal{K} = \mathbb{R}^n_+$, the nonnegative orthant with $F(x) = -\sum_{i=1}^n \log(x_i)$,

2. $\mathcal{K} = \mathbb{L}^n \subset \mathbb{R}^{n+1}_+$, the second-order cone with $F(x) = -\log(t^2 - ||x||_2^2)$,

[4] Some authors use the similar notion where self-duality is defined using a more general inner product $\langle \cdot, \cdot \rangle_S$ with respect to some positive definite matrix S. If nothing else is specified we speak of self-duality using the Euclidean inner product.

[5] In fact the concept of symmetric cones was known since the 1960's. In [53] Nesterov and Todd developed primal-dual methods for conic problems that admit self-scaled barriers. Güler later established the equivalence between the two concepts.

3. $\mathcal{K} = \mathcal{S}^n_+ \subset \mathcal{S}^n$, the cone of positive semidefinite matrices with $F(X) = -\log(\det(X))$.

Nesterov and Todd ([53, 54]) have developed a framework of symmetric primal-dual interior-point methods for conic problems with symmetric cones (and thus self-scaled barriers). These methods actively make use of the existence of scaling points in order to define the symmetric primal-dual directions.

There are several advantages of symmetric primal-dual methods over primal-only (or dual-only) methods. For one, they nicely generalize the practically very efficient primal-dual methods for LP ([67, 41, 58]) to the convex conic case. Further, the variation of the value and the Hessian of F can be bounded in a larger neighborhood around a given point $x \in \operatorname{int}\mathcal{K}$ (compare [53, Theorem 4.1 and 4.2] to Theorem 2.2.10 and Theorem 2.2.16) which allows for potentially larger steps. For example Theorem 2.2.10 bounds the Hessian around some x in the Dikin ellipsoid with radius $r < 1$, i.e. for points y such that $||y - x||_x = r < 1$. On the other hand [53, Theorem 4.1] provides a similar bound around x, but for points in a larger neighborhood which is in fact the cone itself (for any $x \in \operatorname{int}\mathcal{K}$). This can be seen by the fact that the bound [53, Theorem 4.1] is given for points $y = x - \alpha\, p$ where $p \in \mathcal{E}$ is any direction and $\sigma_x(p)$ is a distance measure to the boundary of the cone (see [53, Section 4]). Moreover, $\sigma_x(p)$ is defined such that y from above can be *any* point in the interior of $\mathcal{K}$. In a similar fashion, primal-dual methods allow for adaptive updates of the duality measure t (see [40] and [37]). Primal-dual methods for conic problems over symmetric cones are successfully implemented in SeDuMi ([61]) and SDPT3 ([62]), both covering linear programming, second-order cone programming and semidefinite programming.

However, we also want to mention here that the algorithmic superiority of primal-dual methods over primal-only (or dual-only) methods has not been proven so far. Above we have mentioned some arguments that speak in favor of the conic primal-dual setting, but we also have to admit that they are mainly of conceptual or cosmetic nature. There is no proof that the worst-case complexity of a primal-dual algorithm is better than the one for primal-only algorithms (the best-known complexity so far is in both cases $\mathcal{O}(\sqrt{\nu}\log(1/\epsilon))$).

2.5.3 Conic optimization over nonsymmetric cones

In this section we consider general conic problems $(\bar{P})$ and $(\bar{D})$ over proper cones. We do not assume the cones to be symmetric. Although, there are not too many results on general conic problems, we can mention here some recent progression by Nesterov ([49, 48, 51]). For general conic problems there are mainly two drawbacks as compared to the symmetric case:

1. The conjugate barrier is typically not known. It is given by Definition 2.5.7, which involves the solution of another optimization problems. In general it is not possible to find analytically a solution to it. However, it might be possible to evaluate the conjugate barrier at a given point approximately. In the symmetric case, on the other hand, the conjugates are explicitly known (in fact they are up to a constant equal to the primal barrier).

2. We do not have access to a scaling point (Theorem 2.5.12). This point plays a crucial role in defining primal-dual symmetric search directions. In the nonsymmetric case we do not know how to compute such a scaling point for a primal dual pair (x, s), in fact we do not even know if it exists.

A special conic format

We are coming back now to the primal-dual conic pair $((\bar{P}), (\bar{D}))$ defined in the previous subsection. It is characterized by a proper cone $\mathcal{K}$ and its dual cone $\mathcal{K}^*$, a matrix A with full row rank a primal objective vector c and a dual objective vector b.

The important assumption here is that $\mathcal{K}$ should be proper, that means in particular its interior should not be empty and it should be pointed. Otherwise it is impossible to find a barrier for $\mathcal{K}$.

However, in some situations we might arrive at a primal conic format where some of the variables x are not restricted to a pointed cone, or some of the dual variables s are fixed to 0 (which prevents s be an interior point of the dual cone). Let us therefore adapt our definition of the conic pair, that allows linear equality constraints on the dual side (or equivalently primal free variables). Of course, in the algorithms we will have to treat these constraints differently because they correspond to cones that are not proper. We denote by the primal and dual cone

$$\begin{aligned} \bar{\mathcal{K}} &= \mathcal{K} \times \mathbb{R}^{n_f}, \\ \bar{\mathcal{K}}^* &= \mathcal{K}^* \times \{0\} \end{aligned}$$

where $\mathcal{K}$ denotes the "proper" part of the cone $\bar{\mathcal{K}}$ and n_f is the number of primal free variables. Moreover, we denote by x those components of the primal variables that lie in $\mathcal{K}$ and by x_f those components that are free. On the dual side we denote by s those components of the dual variables that lie in $\mathcal{K}^*$ and by s_f those that are restricted to 0. Let the constraint matrix and the primal objective vector be partitioned accordingly, i.e. $\bar{A} = [A, A_f]$ and $\bar{c} = [c, c_f]$.

The primal-dual conic pair becomes then:

$$\boxed{\begin{aligned} (P) \quad \min_{x, x_f} \; & c^T x + c_f^T x_f \\ Ax + A_f x_f &= b, \\ x &\in \mathcal{K}, \\ x_f & \text{ free} \end{aligned}}$$

$$\boxed{\begin{aligned} (D) \quad \max_{y, s} \; & b^T y \\ s + A^T y &= c \\ A_f^T y &= c_f \\ s &\in \mathcal{K}^* \end{aligned}}$$

It is easy to see that weak duality is preserved. Indeed

$$\begin{aligned} c^T x + c_f^T x_f &= (s + A^T y)^T x + (A_f^T y)^T x_f \\ &= y^T (Ax + A_f^T x_f) + \underbrace{s^T x}_{\geq 0} \\ &\geq b^T y. \end{aligned}$$

The assumptions from the standard conic formulation have to be modified a little bit. Let us summarize them here.

Assumption 3 *[Conic optimization]*

1. *$\mathcal{K}$ and $\mathcal{K}^*$ are proper cones and dual to each other.*
2. *$[A, A_f]$ has full row rank.*
3. *A_f has full column rank.*
4. *The dual feasible set $\{y : c - A^T y \in \mathcal{K}^*, A_f^T y = c_f\}$ does not contain a straight line.*

The first assumption is necessary to ensure weak and strong duality. The second assumption is needed to ensure that the primal and/or dual Newton directions are defined. The third does not restrict generality as we have argued in Section 2.3. If the dual problem is feasible although A_f has columns that are linearly dependent, then some of the constraints on the dual side (or variables on the primal side) are redundant and can be eliminated. The last assumption is needed to guarantee that the dual barrier in terms of y is nondegenerate (and therefore that the dual Newton directions are defined).

Nonsymmetric primal-dual predictor-corrector method

The method we are going to present now is essentially the dual variant of the one proposed by Nesterov in [49]. Additionally, it is adapted to handle linear equality constraints on the dual side, which correspond to free variables on the primal side. Its main feature is that it only needs access to the dual barrier (and its derivatives of course), but not its conjugate. However, it still is a primal-dual method in the sense that we compute primal-dual strictly feasible points close to the primal-dual central path defined by the barrier for the dual cone and its conjugate (that we do not need to know). The results and proofs in this section are largely inspired by [49]. However, for sake of transparency we have decided to include them explicitly.

Each iteration of the method consists of 4 steps:

1. dual centering,
2. primal-dual lifting,
3. primal-dual affine-scaling,
4. updating of duality measure t.

We only assume knowledge of the barrier F_* for the dual cone $\mathcal{K}^* \subset \mathcal{E}^*$. As F_* will be our barrier of reference, we have to adapt the definition of the local norms a bit. Let $s \in \operatorname{int} \mathcal{K}^*$. For $h \in \mathcal{E}$ we denote its local norm with respect to the dual barrier F_* at s by

$$||h||_s := \langle \nabla^2 F_*(s)^{-1} h, h \rangle^{1/2}.$$

Note that the local norm of the point h in the primal space is defined using the *inverse* of the Hessian of F_*, since F_* is defined in the dual space $\mathcal{E}^*$ (compare Definition 2.2.7). Analogously, we define the local norm of $g \in \mathcal{E}^*$ by

$$||g||_s^* := \langle g, \nabla^2 F_*(s) g \rangle^{1/2}.$$

Dual centering

Consider for $t > 0$ the dual centering problem

$$\begin{aligned} \max t\, b^T y - F_*(s) &= f_t(y,s) \\ s.t.\ s + A^T y &= c, \\ A_f^T y &= c_f, \end{aligned} \tag{D_t}$$

with the variables $y \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$. We denote the unique optimal solution for (D_t) by $(y(t), s(t))$. Based on the results of Section 2.3 we want to solve (D_t) using the damped Newton method (Algorithm 2).

The corresponding Newton system is

$$\begin{bmatrix} \nabla^2 f_t(y,s) & \tilde{A}^T \\ \tilde{A} & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y \\ \Delta s \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f_t(y,s) \\ 0 \end{bmatrix}$$

where

$$\tilde{A} = \begin{bmatrix} A^T & I \\ A_f^T & 0 \end{bmatrix} \qquad \lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix},$$

$$\nabla f_t(y,s) = \begin{bmatrix} t\, b \\ -\nabla F_*(s) \end{bmatrix} \qquad \nabla^2 f_t(y,s) = \begin{bmatrix} 0 & 0 \\ 0 & -\nabla^2 F_*(s) \end{bmatrix}$$

so that the Newton system becomes

$$\begin{bmatrix} 0 & 0 & A & A_f \\ 0 & -\nabla^2 F_*(s) & I & 0 \\ A^T & I & 0 & 0 \\ A_f^T & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y \\ \Delta s \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -t \cdot b \\ \nabla F_*(s) \\ 0 \\ 0 \end{bmatrix}. \tag{2.52}$$

The dual Newton steps will be

$$\begin{aligned} \hat{y} &= y + \frac{1}{1+\delta_s} \Delta y, \\ \hat{s} &= s + \frac{1}{1+\delta_s} \Delta s, \end{aligned}$$

where $\delta_s = ||\Delta s||_s^* = \left(-\Delta s^T \nabla F_*(s)\right)^{1/2}$ denotes the Newton decrement. In view of Theorem 2.2.24 we can solve (D_t) in no more than $\mathcal{O}(f_t(y(t), s(t)) - f_t(y^{(0)}, s^{(0)}))$ iterations, where $(y^{(0)}, s^{(0)})$ is the starting point of the process. Let us show that we can bound this quantity for a particular choice of t. Let $z = (x, x_f, y, s)$ be any primal-dual strictly feasible point. We introduce the so-called *functional proximity measure* as a way to define proximity to the primal-dual central path:

$$\Omega(z) = F(x) + F_*(s) + \nu \log\left(\frac{\langle s, x\rangle}{\nu}\right) + \nu,$$

where F is the conjugate to F_*. It is clear why Ω is a proximity measure: for a primal-dual point $(x(t), x_f(t), y(t), s(t))$ on the central path, we have the following two relations (see [54, Theorem 4.1])

$$\begin{aligned} \langle s(t), x(t)\rangle &= \frac{\nu}{t}, \\ F(x(t)) + F_*(s(t)) &= -\nu + \nu \log(t). \end{aligned}$$

Using that, it follows immediately that $\Omega(z) = 0$ if z is on the primal-dual central path. Conversely, if we define for z the duality measure

$$t = t(z) = \frac{\nu}{\langle s, x\rangle},$$

then we have

$$0 \leq f_t(y(t), s(t)) - f_t(y, s) \leq \Omega(z). \tag{2.53}$$

In ([49, Lemma 2]) Nesterov has shown the analogous inequality for the primal centering problem. That means if $\Omega(z) = 0$, then for the particular choice of t shown above the optimality gap to the point on the dual central path is equal to 0, i.e. (y, s) is on the dual central path.

We conclude that, given a primal-dual strictly feasible point $z^{(0)}$, we can solve (D_t) in no more than $\mathcal{O}(\Omega(z^{(0)}))$ iterations, provided that we take as duality measure $t_0 = \frac{\nu}{\langle s^{(0)}, x^{(0)}\rangle}$.

The output of Algorithm 2 is a point (y, s) close to the dual central path with target value t, and it satisfies the centering condition

$$\delta_s \leq \beta < 1,$$

where $\beta > 0$ is the desired accuracy. Note that we can solve (D_t) by removing the dependence on s. This elimination would result in a Newton system of smaller size only in terms of Δy and λ_2. However, as we will see in the next step, we need to compute at least once the full Newton direction $(\Delta y, \Delta s)$ with both multipliers (λ_1, λ_2) at the end of the centering process.

Primal-dual lifting

Given a point (y, s) close to the dual central path, the dual Newton step $(\Delta y, \Delta s)$ and primal multipliers (λ_1, λ_2), we construct the following primal-dual point

$$\begin{aligned} x^+ &= -\frac{1}{t}\lambda_1 \\ x_f^+ &= -\frac{1}{t}\lambda_2 \\ y^+ &= y - \Delta y \\ s^+ &= s - \Delta s. \end{aligned} \tag{2.54}$$

We want to emphasize here that the new dual points (y^+, s^+) are obtained by going a step in the *opposite* Newton direction. This is certainly counter-intuitive. However, as we will see in the next theorem, despite doing this negative Newton step we remain feasible with respect to the dual variables.

Theorem 2.5.13. *The point $z^+ = (x^+, x_f^+, y^+, s^+)$ is strictly feasible and well-centered, i.e. we can bound the functional proximity measure in the following way*

$$\Omega(z^+) \leq 2\omega_*(\beta) + \beta^2. \tag{2.55}$$

Furthermore, $w = \sqrt{t}\, s$ is a scaling point for the primal-dual pair (x^+, s^+), i.e.

$$x^+ = \nabla^2 F_*(w)\, s^+. \tag{2.56}$$

Finally, we have the following scaling relation

$$||\nabla F_*(s^+) - \frac{1}{t}\nabla^2 F_*(s)\nabla F(x^+)||_s \leq \frac{2\beta^2}{1-\beta}. \tag{2.57}$$

The duality measure has the following bounds

$$t(z^+) \geq \frac{t_0}{(1+\beta/\sqrt{\nu})^2} \geq t_0 \cdot \exp\left(-\frac{2\beta}{\sqrt{\nu}}\right), \tag{2.58}$$

$$t(z^+) \leq \frac{t_0}{(1-\beta/\sqrt{\nu})^2} \leq t_0 \cdot \exp\left(\frac{2\beta}{\sqrt{\nu}-\beta}\right). \tag{2.59}$$

Proof. We have

$$||s^+ - s||_s^* = ||-\Delta s||_s^* \leq \beta < 1.$$

Since $s \in \operatorname{int}\mathcal{K}^*$ we have in accordance with Lemma 2.2.9 that $s^+ \in \operatorname{int}\mathcal{K}^*$, i.e. s^+ is strictly feasible. Moreover, it is clear from (2.52) and (2.54) that the lifted dual point (y^+, s^+) is feasible with respect to the linear equality constraints, i.e. it holds $s^+ + A^T y^+ = c$ and $A_f^T y^+ = c_f$.

From the second equation of (2.52) it follows

$$\begin{aligned}
||-\lambda_1 + \nabla F_*(s)||_s &= ||-\nabla^2 F_*(s)\,\Delta s||_s \\
&= \langle \nabla^2 F_*(s)^{-1}\, \nabla^2 F_*(s)\,\Delta s, \nabla^2 F_*(s)\,\Delta s\rangle^{1/2} \\
&= \langle \Delta s, \nabla^2 F_*(s)\,\Delta s\rangle^{1/2} \\
&= ||\Delta s||_s^* \leq \beta < 1.
\end{aligned}$$

On the other hand, if we denote $h = -\nabla F_*(s)$ we have $h \in \operatorname{int}\mathcal{K}$ (see (2.44)) and in view of 2.47) the identity $(\nabla^2 F_*(s))^{-1} = \nabla^2 F(h)$. That means we can write

$$\begin{aligned}
||-\lambda_1 + \nabla F_*(s)||_s &= ||-\lambda_1 - h||_s \\
&= \langle (\nabla^2 F_*(s))^{-1}(-\lambda_1 - h), (-\lambda_1 - h)\rangle^{1/2} \\
&= \langle (\nabla^2 F(h)(-\lambda_1 - h), (-\lambda_1 - h)\rangle^{1/2} \\
&= ||-\lambda_1 - h||_{\nabla^2 F(h)}.
\end{aligned}$$

Since F is self-concordant on $\operatorname{int}\mathcal{K}$, we conclude in combination with the previous result and in view of Lemma 2.2.9 that $-\lambda_1 \in \operatorname{int}\mathcal{K}$. Therefore, according to the primal-dual lifting (2.54) we get $x^+ = -\frac{1}{t}\lambda_1 \in \operatorname{int}\mathcal{K}$. Moreover, according to (2.54) we have

$$Ax^+ + A_f x_f^+ = -\frac{1}{t}\underbrace{[A\lambda_1 + A_f\lambda_2]}_{\overset{(2.52)}{=}-t\,b} = b,$$

which means that (x^+, x_f^+) is also feasible with respect to the primal equality constraints.

To prove (2.55) let us consider now the primal-dual proximity measure at (x^+, s^+):

$$\Omega(z^+) = F(x^+) + F_*(s^+) + \nu + \nu\,\log\left(\frac{\langle s^+, x^+\rangle}{\nu}\right). \tag{2.60}$$

We have that

$$F(x^+) = F\left(-\frac{1}{t}\lambda_1\right) \overset{(2.38),(2.52)}{=} F\left(-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s\right) + \nu\,\log(t). \tag{2.61}$$

Let us try to bound the barrier term of the right-hand side in (2.61). Its argument $-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s$ can be related to $-\nabla F_*(s)$. We find

$$\begin{aligned} || - \nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s - (-\nabla F_*(s))||_s &= || - \nabla^2 F_*(s)\,\Delta s||_s \\ &= ||\Delta s||_s^* \leq \beta < 1, \end{aligned}$$

so we can conclude (again in view of Lemma 2.2.9) that

$$(-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s) \in D(-\nabla F_*(s), \beta),\ \beta < 1$$

where $D(z, \beta)$ is the Dikin ellipsoid with radius β centered at z. That means, in accordance with (2.14), we can bound the value of the primal barrier at the point $(-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s)$:

$$\begin{aligned} F(-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s) \leq& F(-\nabla F_*(s)) \\ &+ \langle \nabla F(-\nabla F_*(s)), -\nabla^2 F_*(s)\,\Delta s\rangle + \omega_*(\beta) \\ \overset{(2.45)}{\leq}& -\nu - F_*(s) + \langle s, \nabla^2 F_*(s)\Delta s\rangle + \omega_*(\beta) \\ \overset{(2.41)}{=}& -\nu - F_*(s) - \langle \Delta s, \nabla F_*(s)\rangle + \omega_*(\beta). \end{aligned} \tag{2.62}$$

If we combine the bound (2.62) with (2.61) and replace it in (2.60), we get

$$\begin{aligned} \Omega(z^+) &= F(x^+) + F_*(s^+) + \nu + \nu\cdot\log\left(\frac{\langle s^+, x^+\rangle}{\nu}\right) \\ &\overset{(2.61)}{=} F(-\nabla F_*(s) - \nabla^2 F_*(s)\cdot\Delta s) + F_*(s^+) + \nu + \nu\cdot\log\left(\frac{t\langle s^+, x^+\rangle}{\nu}\right) \\ &\overset{(2.62)}{\leq} F_*(s^+) - F_*(s) - \langle \Delta s, \nabla F_*(s)\rangle + \omega_*(\beta) + \nu\,\log\left(\frac{t\langle s^+, x^+\rangle}{\nu}\right). \end{aligned}$$

According to (2.14) we can bound the first three terms by

$$\begin{aligned} F_*(s^+) - F_*(s) - \langle \Delta s, \nabla F_*(s)\rangle &\le \omega_*(||\Delta s||_s) - 2\langle \Delta s, \nabla F_*(s)\rangle \\ &\le \omega_*(\beta) - 2\langle \Delta s, \nabla F_*(s)\rangle, \end{aligned}$$

which gives the following bound on $\Omega(z^+)$:

$$\Omega(z^+) \le 2\omega_*(\beta) - 2\langle \Delta s, \nabla F_*(s)\rangle + \nu \log\left(\frac{t\langle s^+, x^+\rangle}{\nu}\right). \tag{2.63}$$

On the other hand, in view of (2.41) and the primal-dual lifting (2.54) we have

$$-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s = \nabla^2 F_*(s)\,(s - \Delta s) = \nabla^2 F_*(s)\, s^+. \tag{2.64}$$

It follows that we can write, using (2.54), (2.52) and (2.42)

$$\begin{aligned} t\,\langle s^+, x^+\rangle &= -\langle s^+, \lambda_1\rangle \\ &\overset{(2.52)}{=} \langle s^+, -\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s\rangle \\ &= \langle s^+, \nabla F_*^2(s) s^+\rangle \\ &= ||s^+||_s^{*2} \\ &= ||s||_s^{*2} - 2\langle s, \nabla F_*^2(s)\,\Delta s\rangle + ||\Delta s||_s^{*2} \\ &\overset{(2.42)}{\le} \nu + 2\langle \Delta s, \nabla F_*(s)\rangle + \beta^2 \\ &= \nu - 2\delta_s^2 + \beta^2. \end{aligned} \tag{2.65}$$

The last equation follows directly from the definition of the Newton decrement $\delta_s = -\langle \Delta s, \nabla F_*(s)\rangle^{1/2}$. If we replace the bound (2.65) in (2.63), we get

$$\Omega(z^+) \le 2\omega_*(\beta) + 2\delta_s^2 + \nu \log\left(1 + \frac{-2\delta_s^2 + \beta^2}{\nu}\right).$$

Looking at the argument of the log-term, we see that $\frac{-2\delta_s^2+\beta^2}{\nu} > -1$, since $\delta_s \le \beta$ (by assumption) and $\delta_s \le \sqrt{\nu}$ (since F_* is ν-self-concordant). Using the fact that $\log(1+\tau) \le \tau$ as long as $\tau > -1$, we conclude

$$\log\left(1 + \frac{-2\delta_s^2 + \beta^2}{\nu}\right) \le \frac{-2\delta_s^2 + \beta^2}{\nu}$$

and combining this with the last bound for $\Omega(z^+)$ we obtain

$$\Omega(z^+) \le 2\omega_*(\beta) + \beta^2,$$

which is exactly the desired inequality (2.55).

Let us prove the scaling relation (2.56). For $w = \sqrt{t}\, s$ we have the following chain of identities

$$\nabla^2 F_*(w)\, s^+ \overset{(2.40)}{=} \frac{1}{t}\nabla^2 F_*(s)\, s^+ \overset{(2.64)}{=} \frac{1}{t}\left(-\nabla F_*(s) - \nabla^2 F_*(s)\,\Delta s\right) \overset{(2.52)}{=} -\frac{1}{t}\lambda_1 \overset{(2.54)}{=} x^+.$$

To prove (2.57) let us denote $r_p := \nabla F_*(s^+) - \nabla F_*(s) + \nabla^2 F_*(s)\Delta s$ and $r_d := \nabla F(-\lambda_1) + s + \Delta s$. Using Lemma 2.2.11, we get

$$||r_p||_s = ||\nabla F_*(s^+) - \nabla F_*(s) + \nabla^2 F_*(s)\Delta s||_s \leq \frac{\beta^2}{1-\beta}$$

because $||s^+ - s||_s^* = ||\Delta s||_s^* = \delta_s \leq \beta$ by assumption. On the other hand

$$\begin{aligned} r_d = {} & \nabla F(-\lambda_1) + s + \Delta s \\ \overset{(2.46)}{=} {} & \nabla F(-\lambda_1) - \nabla F(-\nabla F_*(s)) + \Delta s \\ \overset{(2.47)}{=} {} & \nabla F(-\lambda_1) - \nabla F(-\nabla F_*(s)) + \nabla^2 F(-\nabla F_*(s)) \cdot \nabla^2 F_*(s) \cdot \Delta s \\ \overset{(2.52)}{=} {} & \nabla F\left(-\nabla^2 F_*(s)\Delta s - \nabla F_*(s)\right) - \nabla F(-\nabla F_*(s)) \\ & - \nabla^2 F(-\nabla F_*(s)) \cdot \left(-\nabla^2 F_*(s)\right) \cdot \Delta s. \end{aligned}$$

But since $|| - \nabla^2 F_*(s)\Delta s||_s = ||\Delta s||_s^* = \delta_s \leq \beta$, we can apply again Lemma 2.2.11 and obtain

$$||r_d||_s^* \leq \frac{\beta^2}{1-\beta}.$$

Using the definition of r_p and r_d, we conclude

$$\begin{aligned} \nabla F_*(s^+) - \frac{1}{t_0}\nabla^2 F_*(s)\nabla F(x^+) &= \nabla F_*(s^+) - \nabla^2 F_*(s)\nabla F(-\lambda_1) \\ &= [r_p + \nabla F_*(s) - \nabla^2 F_*(s)\Delta s] - \nabla^2 F_*(s)\,[r_d - s - \Delta s] \\ &= [r_p + \nabla F_*(s)] - \nabla^2 F_*(s)\,[r_d - s] \\ &= [r_p + \nabla F_*(s)] - \nabla^2 F_*(s)\, r_d + \underbrace{\nabla^2 F_*(s)\, s}_{\overset{(2.41)}{=} -\nabla F_*(s)} \\ &= r_p - \nabla^2 F_*(s)\, r_d. \end{aligned}$$

It remains to note that

$$||r_p - \nabla^2 F_*(s)\, r_d||_s \leq ||r_p||_s + ||\nabla^2 F_*(s)\, r_d||_s = ||r_p||_s + ||r_d||_s^* \leq 2\frac{\beta^2}{1-\beta}.$$

Let us prove now the two bounds on the duality measure t. We have

$$\begin{aligned} \langle s^+, x^+ \rangle &\overset{(2.56)}{=} \left\langle \frac{1}{t}\nabla^2 F_*(s)s^+, s^+ \right\rangle \\ &= \frac{1}{t}\left(||s^+||_s^*\right)^2 \leq \frac{1}{t}\left(\underbrace{||s||_s^*}_{=\sqrt{\nu}} + \underbrace{||\Delta s||_s^*}_{\leq \beta}\right)^2 \\ &\leq \frac{1}{t}\left(\sqrt{\nu} + \beta\right)^2, \end{aligned} \tag{2.66}$$

which implies

$$t(z^+) = \frac{\nu}{\langle s^+, x^+\rangle} \geq t\,\frac{\nu}{\left(\sqrt{\nu}+\beta\right)^2} = t\,\frac{1}{\left(1+\frac{\beta}{\sqrt{\nu}}\right)^2} \geq t\,\exp\left(-\frac{2\beta}{\sqrt{\nu}}\right),$$

using the fact that $\exp(\tau) \geq \tau + 1$ for all τ. This proves (2.58). The analogous reasoning gives

$$\begin{aligned}
\langle s^+, x^+\rangle &= \frac{1}{t}\left(||s^+||_s^*\right)^2 = \frac{1}{t}\left(||s-\Delta s||_s^*\right)^2 \\
&= \frac{1}{t}\left((||s||_s^*)^2 - 2\,\underbrace{s^T\nabla^2 F(s)\Delta s}_{\leq ||s||_s^* ||\Delta s||_s^*} + \left(||s^+||_s^*\right)^2\right) \\
&\geq \frac{1}{t}\left((||s||_s^*)^2 - 2||s||_s^*\cdot||\Delta s||_s^* + (||\Delta s||_s^*)^2\right) \\
&= \frac{1}{t}\left(\underbrace{||s||_s^*}_{=\sqrt{\nu}} - \underbrace{||\Delta s||_s^*}_{\leq\beta}\right)^2 \\
&\geq \frac{1}{t}\left(\sqrt{\nu}-\beta\right)^2,
\end{aligned}$$

thus

$$t(z^+) \leq t\cdot\frac{1}{\left(1-\frac{\beta}{\sqrt{\nu}}\right)^2} \leq t\cdot\exp\left(\frac{2\beta}{\sqrt{\nu}-\beta}\right),$$

because

$$\frac{1}{\left(1-\frac{\beta}{\sqrt{\nu}}\right)^2} = \left(\frac{\sqrt{\nu}}{\sqrt{\nu}-\beta}\right)^2 = \left(1+\frac{\beta}{\sqrt{\nu}-\beta}\right)^2.$$

This proves (2.59). □

Primal-dual affine-scaling

Given a primal-dual well-centered point (x^+, x_f^+, y^+, s^+) with the corresponding scaling point w we can define the so-called affine-scaling direction. For points exactly on the primal-dual central path, this direction is tangent to the central path (for a discussion, see e.g. [54, Section 5.1]). The affine-scaling direction is given by the solution of the following linear system.

$$\begin{aligned}
\Delta x + \nabla^2 F_*(w)\,\Delta s &= x^+ \\
A\,\Delta x + A_f\,\mu &= 0 \\
A^T\,\Delta y + \Delta s &= 0 \\
A_f^T\,\Delta y &= 0.
\end{aligned} \tag{2.67}$$

Note that under Assumption 2.5.3 the affine-scaling direction (2.67) is unique.

Theorem 2.5.14. *The affine-scaling direction (2.67) has the following properties:*

$$
\begin{aligned}
\langle \Delta s, \Delta x\rangle &= 0, && (2.68)\\
\langle s^+, \Delta x\rangle + \langle \Delta s, x^+\rangle &= \langle s^+, x^+\rangle, && (2.69)\\
\langle c, x^+ - \Delta x\rangle + \langle c_f, x_f - \mu\rangle - \langle b, y - \Delta y\rangle &= 0, && (2.70)\\
||\Delta x||_w^2 + (||\Delta s||_w^*)^2 &= \langle s^+, x^+\rangle, && (2.71)\\
|\nu + \langle \nabla F(x^+), \Delta x\rangle + \langle \Delta s, \nabla F_*(s^+)\rangle| & \\
\le \frac{1}{2}\langle s^+, x^+\rangle^{1/2}\, & ||\nabla F_*(s^+) - \nabla^2 F_*(w)\nabla F(x^+)||_w && (2.72)\\
& \le \langle t_0 s^+, x^+\rangle^{1/2} \cdot \frac{\beta^2}{1-\beta} && (2.73)\\
& \le \frac{\beta^2}{1-\beta} \cdot (\beta + \sqrt{\nu}). && (2.74)
\end{aligned}
$$

Proof. In view of the last three equations of (2.67) we get (2.68)

$$\langle \Delta s, \Delta x\rangle = -\Delta x^T A^T \Delta y = -(A\Delta x)^T \Delta y = (A_f \mu)^T \Delta y = \mu^T A_f^T \Delta y = 0.$$

Further, we get from the first equation of (2.67)

$$
\begin{aligned}
\langle s^+, \Delta x\rangle + \langle \Delta s, x^+\rangle &= \langle s^+, x^+ - \nabla^2 F_*(w)\Delta s\rangle + \langle \Delta s, x^+\rangle\\
&= \langle s^+, x^+\rangle - \langle s^+, \nabla^2 F_*(w)\Delta s\rangle + \langle \Delta s, x^+\rangle\\
&= \langle s^+, x^+\rangle - \langle \Delta s, \nabla^2 F_*(w) s^+\rangle + \langle \Delta s, x^+\rangle.
\end{aligned}
$$

According to the scaling relation (2.56) we have $\nabla^2 F_*(w)s^+ = x^+$, which proves (2.69).

In order to prove (2.70), we use the fact that view of Theorem 2.5.13 the primal-dual point (x^+, x_f^+, y^+, s^+) satisfied the primal and dual linear equality constraints, i.e.

$$
\begin{aligned}
Ax^+ + A_f x_f^+ &= b,\\
s^+ + A^T y^+ &= c,\\
A_f^T y^+ &= c_f.
\end{aligned}
$$

Replacing these expressions for b, c and c_f in the left-hand side of (2.70) yields

$$\begin{aligned}
\langle c, x^+ - \Delta x\rangle + \langle c_f, x_f - \mu\rangle - \langle b, y^+ - \Delta y\rangle \\
=& \langle s^+ + A^T y^+, x^+ - \Delta x\rangle + \langle A_f^T y^+, x_f - \mu\rangle \\
& - \langle Ax^+ + A_f x_f^+, y^+ - \Delta y\rangle \\
=& \langle s^+, x^+ - \Delta x\rangle - \langle A^T y^+, \Delta x\rangle - \langle A_f^T y^+, \mu\rangle \\
& + \langle Ax^+ + A_f x_f^+, \Delta y\rangle \\
=& \langle s^+, x^+ - \Delta x\rangle - \langle y^+, \underbrace{A\Delta x + A_f \mu}_{\overset{(2.67)}{=} 0}\rangle \\
& + \langle Ax^+ + A_f x_f^+, \Delta y\rangle \\
=& \langle s^+, x^+ - \Delta x\rangle + \langle Ax^+ + A_f x_f^+, \Delta y\rangle \\
=& \langle s^+, x^+\rangle - \langle s^+, \Delta x\rangle - \langle \Delta s, x^+\rangle + \langle \Delta s, x^+\rangle \\
& + \langle Ax^+, \Delta y\rangle + \langle A_f x_f^+, \Delta y\rangle \\
=& \langle s^+, x^+\rangle - \langle s^+, \Delta x\rangle - \langle \Delta s, x^+\rangle \\
& + \langle \underbrace{\Delta s + A^T \Delta y}_{\overset{(2.67)}{=} 0}, x^+\rangle + \langle \underbrace{A_f^T \Delta y}_{\overset{(2.67)}{=} 0}, x_f^+\rangle \\
=& \langle s^+, x^+\rangle - \langle s^+, \Delta x\rangle - \langle \Delta s, x^+\rangle.
\end{aligned}$$

The last term is equal to 0 in according to (2.69). This proves (2.70).

Moreover, when we multiply the first line of (2.67) from the left by Δs^T, we get

$$\langle \Delta s, \nabla^2 F_*(w)\Delta s\rangle + \underbrace{\langle \Delta s, \Delta x\rangle}_{\overset{(2.68)}{=} 0} = \langle \Delta s, x^+\rangle,$$

which means $(||\Delta s||_w^*)^2 = \langle \Delta s, x^+\rangle$. If we multiply the first line of (2.67) from the left by $(\nabla^2 F(w)^{-1}\Delta x)^T$ and use again the scaling relation $\nabla^2 F_*(w)s^+ = x^+$, we get

$$\underbrace{\langle \Delta s, \Delta x\rangle}_{=0} + \langle \Delta x^T \nabla^2 F_*(w)^{-1}, \Delta x\rangle = \langle s^+, \Delta x\rangle,$$

which means $||\Delta x||_w^2 = \langle s^+, \Delta x\rangle$. Combining these two reformulations, we obtain

$$||\Delta x||_w^2 + (||\Delta s||_w^*)^2 = \langle s^+, \Delta x\rangle + \langle \Delta s, x^+\rangle \overset{(2.69)}{=} \langle s^+, x^+\rangle,$$

which is exactly (2.71).

Finally, if we multiply the first line of (2.67) by $\nabla F(x^+)^T$, we get

$$\langle \nabla F(x^+), \Delta x + \nabla^2 F_*(w)\Delta s\rangle = \langle \nabla F(x^+), x^+\rangle \overset{(2.43)}{=} -\nu.$$

On the other hand, if we multiply the first line of (2.67) by $\left([\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+)\right)^T$, we get

$$\begin{aligned}\langle[\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+), \Delta x + \nabla^2 F_*(w)\Delta s\rangle &= \langle[\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+), x^+\rangle \\ &= \langle\underbrace{\nabla^2 F_*(w)]^{-1}x^+}_{\overset{(2.56)}{=}s^+}, \nabla F_*(s^+)\rangle \\ &= \langle s^+, \nabla F_*(s^+)\rangle \\ &\overset{(2.43)}{=} -\nu.\end{aligned}$$

If we add these two equalities, we get

$$\begin{aligned}-2\nu =& \langle\nabla F(x^+) + [\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+), \nabla^2 F_*(w)\Delta s + \Delta x\rangle \\ =& \langle\nabla F(x^+) - [\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+), \nabla^2 F_*(w)\Delta s - \Delta x\rangle + 2\langle\nabla F(x^+), \Delta x\rangle \\ &+ 2\langle\Delta s, \nabla F_*(s^+)\rangle,\end{aligned}$$

which is the same as

$$\begin{aligned}&2\langle\nabla F(x^+), \Delta x\rangle + 2\langle\Delta s, \nabla F_*(s^+)\rangle + 2\nu \\ &\qquad = \langle\nabla F(x^+) - [\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+), \nabla^2 F_*(w)\Delta s - \Delta x\rangle.\end{aligned}$$

Using the Hölder inequality we have

$$\begin{aligned}|\nu + \langle\nabla F(x^+), \Delta x\rangle + \langle\Delta s, \nabla F_*(s^+)\rangle| \le& \frac{1}{2}||\nabla F(x^+) - [\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+)||_w^* \\ &\cdot ||\nabla^2 F_*(w)\Delta s - \Delta x||_w.\end{aligned}$$

Because of orthogonality of Δs and Δx, we get

$$\left(||\nabla^2 F_*(w)\Delta s - \Delta x||_w\right)^2 = (||\Delta s||_w^*)^2 + ||\Delta x||_w^2.$$

The last term is equal to $\langle s^+, x^+\rangle$ in view of (2.71). It remains to note that

$$||\nabla F(x^+) - [\nabla^2 F_*(w)]^{-1}\nabla F_*(s^+)||_w^* = ||\nabla F_*(s^+) - \nabla^2 F_*(w)\nabla F(x^+)||_w,$$

which proves (2.72).

According to (2.57) we have

$$||\nabla F_*(s^+) - \nabla^2 F_*(w)\nabla F(x^+)||_s \le \frac{2\beta^2}{1-\beta}.$$

Using the scaling relation $w = \sqrt{t}\,s$ and the identity (2.40) we conclude (note that the primal norm is defined using the *inverse* of $\nabla^2 F$)

$$||\nabla F_*(s^+) - \nabla^2 F_*(w)\nabla F(x^+)||_w \le \sqrt{t}\,\frac{2\beta^2}{1-\beta}.$$

In combination with (2.72) we get the desired inequality (2.73).

The last inequality (2.74) follows directly from (2.66). □

The above theorem justifies why the primal-dual affine-scaling direction is potentially a good direction. In fact, according to (2.70) a full affine-scaling step reduces the duality gap to 0 (this is always the case for affine-scaling directions; however, a full step is never feasible).

Let us denote by $z^+ = (x^+, y^+, s^+)$ the primal-dual strictly feasible point which is the result of the primal-dual lifting (2.54), let $\Delta z^{(AS)} = (\Delta x, \mu, \Delta y, \Delta s)$ be the primal-dual affine-scaling direction from (2.67). We define the new primal-dual point $\tilde{z}$ as

$$\tilde{z} = z^+ - \alpha\, \Delta z^{(AS)}. \tag{2.75}$$

Note that we are taking a negative step since (2.67) defines in fact the direction towards the analytic center. We are interested in the opposite direction towards the optimal solution.

The step size α has to be chosen to ensure that the new iterate is strictly feasible. The following theorem justifies a good choice for α which guarantees the inclusions $\tilde{x} \in \operatorname{int} \mathcal{K}$ and $\tilde{s} \in \operatorname{int} \mathcal{K}^*$. On the other hand, using the proposed step size we can bound the proximity measure Ω at the new point $\tilde{z}$, i.e. we make sure not to drift too far from the primal-dual central path.

Theorem 2.5.15. *Denote* $\gamma = \frac{\beta+\sqrt{\nu}}{1-\beta} > 1$. *For* $\alpha \in \left[0, \frac{1}{\gamma}\right)$ *we have*

$$\Omega(\tilde{z}) \leq 2\omega_*(\beta) + \omega_*(\alpha\gamma) + \beta^2(1+\alpha\gamma).$$

Proof. Using the definition of the proximity measure Ω we get

$$\begin{aligned}\Omega(\tilde{z}) - \Omega(z^+) =& F(\tilde{x}) + F_*(\tilde{s}) + \nu \log\left(\frac{\langle \tilde{s}, \tilde{x}\rangle}{\nu}\right) + \nu \\ &- F(x^+) - F_*(s^+) - \nu \log\left(\frac{\langle s^+, x^+\rangle}{\nu}\right) - \nu \\ =& F(\tilde{x}) + F_*(\tilde{s}) - F(x^+) - F_*(s^+) + \nu \log\left(\frac{\langle \tilde{s}, \tilde{x}\rangle}{\langle s^+, x^+\rangle}\right).\end{aligned}$$

According to the definition of $\tilde{s}$ and $\tilde{x}$ we have

$$\begin{aligned}\frac{\langle \tilde{s}, \tilde{x}\rangle}{\langle s^+, x^+\rangle} &= \frac{\langle s^+ - \alpha\Delta s, x^+ - \alpha\Delta x\rangle}{\langle s^+, x^+\rangle} = 1 - \alpha \underbrace{\frac{\langle s^+, \Delta x\rangle + \langle \Delta s, x^+\rangle}{\langle s^+, x^+\rangle}}_{\overset{(2.69)}{=} 1} + \alpha^2 \underbrace{\frac{\langle \Delta s, \Delta x\rangle}{\langle s^+, x^+\rangle}}_{\overset{(2.68)}{=} 0} \\ &= 1 - \alpha.\end{aligned}$$

That means we can write

$$\Omega(\tilde{z}) - \Omega(z^+) = F(\tilde{x}) + F_*(\tilde{s}) - F(x^+) - F_*(s^+) + \nu \log(1-\alpha).$$

In a neighborhood around x^+ and around s^+ we can bound the function value of

F and F_* respectively (see (2.14)). We get

$$\begin{aligned}\Omega(\tilde{z}) - \Omega(z^+) =& F(\tilde{x}) - F(x^+) + F_*(\tilde{s}) - F_*(s^+) + \nu \log(1-\alpha)\\ \leq& -\alpha\langle \nabla F(x^+), \Delta x\rangle + \omega_*(\alpha||\Delta x||_{x^+})\\ &-\alpha\langle \nabla F_*(s^+), \Delta s\rangle + \omega_*(\alpha||\Delta s||^*_{s^+}) + \nu\log(1-\alpha)\\ =& -\alpha[\langle \nabla F(x^+), \Delta x\rangle + \langle \nabla F_*(s^+), \Delta s\rangle] + \nu \log(1-\alpha)\\ &+ \omega_*(\alpha||\Delta x||_{x^+}) + \omega_*(\alpha||\Delta s||^*_{s^+}),\end{aligned}$$

provided that $\alpha||\Delta x||_{x^+} < 1$ and $\alpha||\Delta s||^*_{s^+} < 1$.

From (2.74) it follows (using the definition of γ)

$$\langle \nabla F(x^+), \Delta x\rangle + \langle \nabla F_*(s^+), \Delta s\rangle \geq -\frac{\beta^2}{1-\beta}(\beta + \sqrt{\nu}) - \nu = -\beta^2\gamma - \nu.$$

Replacing this inequality in the previous bound on $\Omega(\tilde{z}) - \Omega(z^+)$ yields

$$\begin{aligned}\Omega(\tilde{z}) - \Omega(z^+) \leq& \alpha\beta^2\gamma + \nu\alpha + \nu\log(1-\alpha)\\ &+ \omega_*(\alpha||\Delta x||_{x^+}) + \omega_*(\alpha||\Delta s||^*_{s^+}).\end{aligned}$$

Using the fact that $\alpha + \log(1-\alpha) \leq 0$, for all $\alpha < 1$, we can remove the second and third term on the right-hand side and get the following simplified bound

$$\Omega(\tilde{z}) - \Omega(z^+) \leq \alpha\beta^2\gamma + \omega_*(\alpha||\Delta x||_{x^+}) + \omega_*(\alpha||\Delta s||^*_{s^+}).$$

Let us look at the last two terms. We denote $r = [||\Delta x||^2_{x^+} + (||\Delta s||^*_{s^+})^2]^{1/2}$. We define the function $\psi(t) := \omega_*(\sqrt{t})$, which is convex because we have that $\psi'(t) = \frac{\omega'(\sqrt{t})}{2\sqrt{t}}$ and

$$\begin{aligned}\psi''(t) &= \omega''(\sqrt{t})\left(\frac{1}{2\sqrt{t}}\right)^2 - \omega'(\sqrt{t})\frac{1}{4t^{3/2}}\\ &= \frac{1}{4t^{3/2}}\left[\sqrt{t}\,\omega''(\sqrt{t}) - \omega'(\sqrt{t})\right]\\ &= \frac{1}{4t^{3/2}}\left[\frac{\sqrt{t}}{(1-\sqrt{t})^2} - \frac{\sqrt{t}}{1-\sqrt{t}}\right]\\ &= \frac{1}{4t^{3/2}}\frac{t}{(1-\sqrt{t})^2} > 0.\end{aligned}$$

Moreover, we have for any $0 \leq t_1, t_2$ with $t_1 + t_2 < 1$ that

$$\psi(t_1) + \psi(t_2) \leq \psi(t_1 + t_2).$$

Indeed, we can assume without loss of generality that $0 \leq t_1 \leq t_2 \leq t_1 + t_2 < 1$. If $t_2 = 0$, then this implies that $t_1 = t_1 + t_2 = 0$ and the above inequality is trivially satisfied. Let therefore $t_2 > 0$. Then we can write t_1 as a convex combination of 0 and $t_1 + t_2 > 0$, i.e. $t_1 = \lambda\, 0 + (1-\lambda)\,(t_1 + t_2)$. Thus, $\lambda = \frac{t_2}{t_1+t_2}$. Because of convexity we get then

$$\begin{aligned}\psi(t_1) &\leq \frac{t_2}{t_1+t_2}\underbrace{\psi(0)}_{=0} + \left(1-\frac{t_2}{t_1+t_2}\right)\psi(t_1+t_2)\\ &= \frac{t_1}{t_1+t_2}\,\psi(t_1+t_2),\end{aligned}$$

or equivalently, after multiplying both sides of the inequality with $\frac{t_1+t_2}{t_2} = \frac{t_1}{t_2}+1 > 0$, we have

$$\left(1+\frac{t_1}{t_2}\right)\psi(t_1) \leq \frac{t_1}{t_2}\,\psi(t_1+t_2). \tag{2.76}$$

On the other hand we can write t_2 as a convex combination of t_1 and t_1+t_2, i.e. $t_2 = \frac{t_1}{t_2}\,t_1 + (1-\frac{t_1}{t_2})\,(t_1+t_2)$, where $\frac{t_1}{t_2} \in [0,1]$. Because of convexity of ψ we get

$$\begin{aligned}\psi(t_2) &\leq \frac{t_1}{t_2}\,\psi(t_1) + \left(1-\frac{t_1}{t_2}\right)\psi(t_1+t_2)\\ &= \underbrace{\frac{t_1}{t_2}\,[\psi(t_1)-\psi(t_1+t_2)]}_{\overset{(2.76)}{\leq}\ -\psi(t_1)} + \psi(t_1+t_2)\\ &\leq -\psi(t_1)+\psi(t_1+t_2).\end{aligned}$$

That means we have for any $t_1 \geq 0$ and any $t_2 \geq 0$ such that $t_1+t_2 < 1$ that $\psi(t_1)+\psi(t_2) \leq \psi(t_1+t_2)$.

It follows

$$\begin{aligned}\omega_*(\alpha||\Delta x||_{x^+}) + \omega_*(\alpha||\Delta s||^*_{s^+}) &= \psi(\alpha^2||\Delta x||^2_{x^+}) + \psi(\alpha^2(||\Delta s||^*_{s^+})^2)\\ &\leq \psi(\alpha^2[||\Delta x||^2_{x^+} + (||\Delta s||^*_{s^+})^2])\\ &= \psi(\alpha^2 r^2)\\ &= \omega_*(\alpha r).\end{aligned}$$

On the other hand, because $||s^+ - s||_s \leq \beta < 1$, we have according to Theorem 2.2.10

$$\nabla^2 F_*(s^+) \preceq \frac{1}{(1-\beta)^2}\nabla^2 F_*(s),$$

and using the scaling relation $w = \sqrt{t}\,s$ in combination with (2.40) yields

$$\nabla^2 F_*(s^+) \preceq \frac{t}{(1-\beta)^2}\nabla^2 F_*(w). \tag{2.77}$$

On the other hand we have

$$\frac{1}{t^2}\nabla^2 F(x^+) \overset{(2.40)}{=} \nabla^2 F(tx^+) \overset{(2.54)}{=} \nabla^2 F(-\lambda_1).$$

In the proof of Theorem 2.5.13 we have seen that $||-\lambda_1-(-\nabla F_*(s))||_s \leq \beta < 1$. Using again Theorem 2.2.10 we get

$$\begin{aligned} \frac{1}{t^2}\nabla^2 F(x^+) &\preceq \frac{1}{(1-\beta)^2}\nabla^2 F(-\nabla F_*(s)) \\ &\overset{(2.47)}{=} \frac{1}{(1-\beta)^2}\nabla^2 F_*(s)^{-1} \\ &\overset{(2.40)}{=} \frac{1}{t(1-\beta)^2}\nabla^2 F_*(w)^{-1}. \end{aligned}$$

In other words, we have

$$\nabla^2 F(x^+) \preceq \frac{t}{(1-\beta)^2}\nabla^2 F_*(w)^{-1}. \tag{2.78}$$

Let us find a bound on the quantity r. Using inequalities (2.77) and (2.78), we get

$$\begin{aligned} r^2 &= ||\Delta x||_{x^+}^2 + (||\Delta s||_{s^+}^*)^2 \\ &= \langle \nabla^2 F(x^+)\Delta x, \Delta x\rangle + \langle \Delta s, \nabla^2 F_*(s^+)\Delta s\rangle \\ &\leq \frac{t}{(1-\beta)^2}[\langle \nabla^2 F_*(w)^{-1}\Delta x, \Delta x\rangle + \langle \Delta s, \nabla^2 F_*(w)\Delta s\rangle] \\ &= \frac{t}{(1-\beta)^2}\cdot (||\Delta x||_w^2 + (||\Delta s||_w^*)^2) \end{aligned}$$

Finally, the right-hand side term can be bounded in the following way.

$$(||\Delta x||_w^2 + (||\Delta s||_w^*)^2) \overset{(2.71)}{=} \langle s^+, x^+\rangle \overset{(2.66)}{\leq} \frac{1}{t}(\sqrt{\nu}+\beta)^2$$

That means we get the following bound on r:

$$r^2 \leq \frac{t}{(1-\beta)^2}\cdot (||\Delta x||_w^2 + (||\Delta s||_w^*)^2) \leq \frac{(\sqrt{\nu}+\beta)^2}{(1-\beta)^2} = \gamma^2.$$

Since both r and γ are positive, we conclude $r \leq \gamma$. Using monotonicity of ω_* we get then

$$\omega_*(\alpha\, r) \leq \omega_*(\alpha\, \gamma).$$

Coming back to the original quantity that we wanted to estimate, we get

$$\begin{aligned} \Omega(\tilde{z}) - \Omega(z^+) &\leq \alpha\beta^2\gamma + \omega_*(\alpha||\Delta x||_{x^+}) + \omega_*(\alpha||\Delta s||_{s^+}^*) \\ &\leq \alpha\beta^2\gamma + \omega_*(\alpha\, r) \\ &\leq \alpha\beta^2\gamma + \omega_*(\alpha\, \gamma). \end{aligned}$$

To finish the proof it remains to note that according to Theorem 2.5.13 we have

$$\Omega(z^+) = 2\omega_*(\beta) + \beta^2.$$

□

Update of duality measure t

Given a primal-dual strictly feasible point z^+ and the affine-scaling direction $\Delta z^{(AS)}$ that is given by 2.67, we go a damped step along this direction (see (2.75)). In Theorem 2.5.15 we have proposed a suitable choice of the step size parameter α that ensures proximity to the primal-dual central path. We will show now that this step also provides a sufficient increase of the duality measure. Let us define the new duality measure as

$$t(\tilde{z}) = \frac{\nu}{\langle \tilde{s}, \tilde{x} \rangle}.$$

Theorem 2.5.16. *We have*

$$t(\tilde{z}) \geq t(z^+) \cdot \exp\left(\alpha - \frac{2\beta}{\sqrt{\nu}}\right).$$

Proof. We see that

$$\begin{aligned} t(\tilde{z}) = t(z^+ - \alpha\Delta z) &= \frac{\nu}{\langle s^+ - \alpha\Delta s, x^+ - \alpha\Delta x\rangle} \\ &= \frac{\nu}{\langle s^+, x^+\rangle \underbrace{-\alpha(\langle s^+, \Delta x\rangle + \langle \Delta s, x^+\rangle)}_{\overset{(2.69)}{=} -\alpha\langle s^+, x^+\rangle} + \alpha^2 \underbrace{\langle \Delta s, \Delta x\rangle}_{\overset{(2.68)}{=} 0}} \\ &= \frac{1}{1-\alpha} \cdot \frac{\nu}{\langle s^+, x^+\rangle}. \end{aligned}$$

According to (2.66) we have $t\,\langle s^+, x^+\rangle \leq (\sqrt{\nu} + \beta)^2 = \nu(1 + \frac{\beta}{\sqrt{\nu}})^2$. Using this inequality, we get

$$\begin{aligned} t(\tilde{z}) &\geq \frac{1}{1-\alpha} \frac{t}{(1 + \frac{\beta}{\sqrt{\nu}})^2} \\ &\geq \underbrace{\frac{1}{1-\alpha}}_{\geq \exp(\alpha)} t \exp\left(-\frac{2\beta}{\sqrt{\nu}}\right) \\ &\geq t \exp\left(\alpha - \frac{2\beta}{\sqrt{\nu}}\right). \end{aligned}$$

□

It is clear that if we want to make sure that there is an actual increase in the duality measure t we need to take a step size α such that

$$\alpha > \frac{2\beta}{\sqrt{\nu}} =: \tau.$$

On the other hand, Theorem 2.5.15 is only valid for step sizes such that

$$\alpha < \frac{1-\beta}{\beta + \sqrt{\nu}} = \frac{1}{\gamma}.$$

Let us check for which choice of β the interval $\left[\tau, \frac{1}{\gamma}\right]$ has interior points. Or, equivalently, we want to verify that

$$\tau < \frac{1}{\gamma}.$$

This is true if and only if $\sqrt{\nu}(1-\beta) > 2\beta(\beta+\sqrt{\nu})$, which is equivalent to $\Leftrightarrow$ $\sqrt{\nu}(1-3\beta) > 2\beta^2$. Isolating ν, we get the condition

$$\sqrt{\nu} > \frac{2\beta^2}{(1-3\beta)}. \tag{2.79}$$

The right-hand side term of (2.79) is less than 1 if $3\beta < 1$ and $2\beta^2 < 1 - 3\beta$. If we solve the latter inequality for β, we get

$$\left(\beta + \frac{3}{4}\right)^2 < \frac{17}{16},$$

which means that as soon as

$$0 \leq \beta < \frac{-3+\sqrt{17}}{4} \approx 0.2808$$

it is possible to find an α that satisfies both above criteria. Then we can choose $\alpha = \lambda\tau + (1-\lambda)\frac{1}{\gamma}$, for any $\lambda \in (0,1)$. A choice of λ close to 1 stands for a conservative prediction with slow progress in terms of the duality measure t, but also smaller increase in the proximity measure.

Note that we have used the fact that $\nu \geq 1$, and we have bounded only the right-hand side term of (2.79) away from 1. This was sufficient, yet not necessary, to show validity of (2.79). Therefore it is possible to find slightly larger values of β than $\frac{-3+\sqrt{17}}{4}$ (keep in mind that $\beta < 1/3$) that satisfy (2.79). However, they do involve ν. For sake of simplicity, we continue with the safe choice of β shown above.

Complexity analysis of the primal-dual predictor-corrector method

We are ready now to define the nonsymmetric primal-dual predictor-corrector method with a bias on the dual space (Algorithm 5).

Let the parameters β, λ, γ and τ be as described in Algorithm 5. We have seen above that when choosing $\beta < \frac{-3+\sqrt{17}}{4}$, then $\tau < \frac{1}{\gamma}$.

Theorem 2.5.17. *The rate of convergence of Algorithm 5 is*

$$\langle s_k, x_k\rangle \leq \langle s_0, x_0\rangle \exp(-k\rho_1),$$

where $\rho_1 = (1-\lambda)\cdot\left[\frac{1}{\gamma} - \tau\right] > 1.$

Algorithm 5 Nonsymmetric predictor-corrector method in dual space

```
Input: $A$ with full row rank, $b, c$ and $F_*$ $\nu$-self-concordant barrier for $\mathcal{K}^*$.
Parameter: Choose $\epsilon > 0$, $0 < \beta < \frac{-3+\sqrt{17}}{4}$ and $\lambda \in (0,1)$. Define $\gamma = \frac{\beta+\sqrt{\nu}}{1-\beta}$,
$\tau = \frac{2\beta}{\sqrt{\nu}}$, $\alpha = \lambda\tau + (1-\lambda)\frac{1}{\gamma}$.
Initialize: $z_0$ primal-dual strictly feasible starting point, $t_0 = \frac{\nu}{\langle s_0, x_0 \rangle}$, $k = 0$.
  loop
    1) correction phase, compute dual Newton steps from (2.52) until $\delta_{t_k}(s) \le \beta$
    2) primal-dual lifting (2.54), output: $z^+$
    if $\langle s^+, x^+ \rangle < \epsilon$ then
      RETURN
    end if
    3) primal-dual affine scaling step (2.67) $z_{k+1} = z^+ - \alpha \Delta z^{(AS)}$
    4) update of duality measure $t_{k+1} = \frac{\nu}{s_{k+1}^T x_{k+1}}$
    5) $k = k + 1$
  end loop
```

Proof. Indeed, using Theorem 2.5.16 we have

$$\begin{aligned} t_{k+1} \ge t_k \cdot \exp(\alpha - \tau) &= t_k \, \exp\left(-(1-\lambda)\tau + (1-\lambda)\frac{1}{\gamma}\right) \\ &= t_k \, \exp\left((1-\lambda)\left[\frac{1}{\gamma} - \tau\right]\right) \\ &= t_k \, \exp\left(\rho_1\right), \end{aligned}$$

and using the relation $t_k \langle s_k, x_k \rangle = \nu$, we get

$$\langle s_k, x_k \rangle \le \langle s_{k-1}, x_{k-1} \rangle \, \exp(-\rho_1) \le \langle s_0, x_0 \rangle \, \exp(-\rho_1)^k = \langle s_0, x_0 \rangle \, \exp(-k\rho_1).$$

□

For the final complexity theorem, let us denote $\rho_2 = \alpha\gamma$. Since $\alpha = \lambda\tau + (1-\lambda)\frac{1}{\gamma}$, we see that

$$\rho_2 = \alpha\gamma = \left(\lambda\tau + (1-\lambda)\frac{1}{\gamma}\right)\gamma = \lambda\tau\gamma + (1-\lambda) = \lambda(\gamma\tau - 1) + 1 < 1.$$

The latter inequality follows from the fact that we have chosen β such that $\tau < \frac{1}{\gamma}$ (see discussion above).

Theorem 2.5.18. *Let the parameters be chosen as in Algorithm 5, let z_0 be a primal-dual strictly feasible point. Then the number of affine-scaling steps in Algorithm 5 to generate a primal-dual feasible point (x, s) such that the duality gap is not more than ϵ, is bounded by*

$$N_{out} = \mathcal{O}\left(\sqrt{\nu} \cdot \log\left(\langle s_0, x_0 \rangle / \epsilon\right)\right).$$

The complexity for solving the initial centering problem is $N_0 = \mathcal{O}(z_0)$ iterations. After that, the number of iterations for each centering problem to generate the dual central points is bounded by a constant, i.e.

$$N_{in} \leq \frac{2\omega_*(\beta) + \omega_*(\rho_2) + \beta^2(1+\rho_2)}{\omega(\beta)}.$$

Proof. Using Theorem 2.5.17, we see that the duality gap $s_k^T x_k$ in iteration k is bounded by ϵ if $s_0^T x_0 \cdot \exp(-k\rho_1) \leq \epsilon$. When isolating k, we get the condition

$$k \geq -\frac{\log(\epsilon / s_0^T x_0)}{\rho_1} = \frac{\log(s_0^T x_0/\epsilon)}{\rho_1}.$$

Moreover, we see that

$$\begin{aligned}
\frac{1}{\rho_1} &= \frac{1}{1-\lambda} \cdot \frac{\gamma}{1-\tau\gamma} \\
&= \frac{1}{1-\lambda} \cdot \frac{\frac{\beta+\sqrt{\nu}}{1-\beta}}{1 - \frac{2\beta}{\sqrt{\nu}}\frac{\beta+\sqrt{\nu}}{1-\beta}} \\
&= \frac{1}{1-\lambda} \cdot \frac{(\beta+\sqrt{\nu})\sqrt{\nu}}{(1-\beta)\sqrt{\nu} - 2\beta(\beta+\sqrt{\nu})} \\
&= \frac{1}{1-\lambda} \cdot \frac{\nu + \sqrt{\nu}\beta}{(1-3\beta)\sqrt{\nu} - 2\beta^2} \\
&= \mathcal{O}(\sqrt{\nu}).
\end{aligned}$$

The complexity bound for the correction phases follows immediately from (2.53) in combination with Theorem 2.5.15 and the fact that we apply a damped Newton method for minimizing a self-concordant function up to accuracy β. □

Remark 2.5.19. *Algorithm 5 requires as input a primal-dual strictly feasible point z_0. However, we can still use Algorithm 5 if only a dual strictly feasible point (y_0, s_0) is available, while the complexity result (Theorem 2.5.18) remains essentially the same.*

The modifications for that situation are the following: we have to choose *the initial duality value t_0 (instead of computing it as in Algorithm 5). The complexity for solving the initial centering problem becomes then in view of Theorem 2.2.24*

$$N_0 = \mathcal{O}(f_{t_0}(y(t_0), s(t_0)) - f_{t_0}(y_0, s_0)),$$

as opposed to $\mathcal{O}(z_0)$. If no primal strictly feasible point is at hand, then there is no immediate way to bound the initial functional gap $f_{t_0}(y(t_0), s(t_0)) - f_{t_0}(y_0, s_0)$ further. The rest of Theorem 2.5.18 is unchanged.

CHAPTER 3

New self-concordant barriers for nonsymmetric cones

We demonstrated in the previous chapter that in order to design polynomial-time algorithms for convex optimization problems it is crucial to have self-concordant barriers for the feasible set available. Moreover, if the feasible set is defined using a proper cone, then we can use primal-dual methods, such as the primal-dual predictor-corrector method proposed in Section 2.5.

In this chapter we propose new self-concordant barriers for two important nonsymmetric cones, the so-called power cone and the p-cone.

3.1 A new barrier for the power cone

Let us consider the following convex cone. For $\alpha \in [0, 1]$ we define the *power cone* as

$$\mathcal{K}_\alpha := \left\{(x, z) \in \mathbb{R}^2_+ \times \mathbb{R} : x_1^\alpha x_2^{1-\alpha} \geq |z|\right\}.$$

This cone has been proposed already in the 50's by Koecher [35].

We will see in Chapter 4 that $\mathcal{K}_\alpha$ is very versatile in that it can be used to model many convex constraints. We see that if $\alpha = 0$ or $\alpha = 1$, then $\mathcal{K}_\alpha$ reduces to a polyhedral cone, e.g. for $\alpha = 0$, we get

$$\mathcal{K}_0 = \{(x_1, x_2, z) : x_1 \geq 0, x_2 \geq |z|\}.$$

On the other hand, for $\alpha = \frac{1}{2}$ the cone $\mathcal{K}_\alpha$ is exactly the rotated second-order cone in dimension 3:

$$\mathcal{Q}^3 = \{(x, z) \in \mathbb{R}^2_+ \times \mathbb{R} : x_1 x_2 \geq z^2\}.$$

For these special values of α the power cone is in fact symmetric (see Definition 2.5.10). In all other cases $\mathcal{K}_\alpha$ is nonsymmetric. Indeed, Truong and Tunçel [63] have presented a proof that $\mathcal{K}_\alpha$ is indeed not homogeneous (except for $\alpha \in \{0, 0.5, 1\}$).

In [49] Nesterov has proposed the following 4-self-concordant barrier for $\mathcal{K}_\alpha$:

$$F(x, z) = -\log(x_1^{2\alpha} x_2^{2-2\alpha} - z^2) - \log(x_1) - \log(x_2).$$

Given the observation that for the particular values of α cited above there are self-concordant barriers known with lower self-concordance parameter (for $\alpha \in \{0, 1\}$ we have $\nu = 3$ and for $\alpha = \frac{1}{2}$ we have $\nu = 2$), it becomes clear that F is not optimal with respect to its self-concordance parameter. In this chapter we propose a new self-concordant barrier with better self-concordance parameter.

3.1.1 Proof of self-concordance

We will see now that when scaling the last two terms of the above 4-self-concordant barrier for $\mathcal{K}_\alpha$, we preserve the self-concordance property while reducing the parameter value from 4 to 3.

Theorem 3.1.1. *Let $\alpha \in [0, 1]$. The function*

$$F_\alpha(x, z) = -\log(x_1^{2\alpha} x_2^{2-2\alpha} - z^2) - (1 - \alpha)\log(x_1) - \alpha \log(x_2)$$

is a 3-self-concordant barrier for the power cone $\mathcal{K}_\alpha$.

Proof. First, let us note that F_α is logarithmically homogeneous of degree 3. In view of Theorem 2.5.6 we need to show that F_α is on top of that a self-concordant function, i.e. it is necessary to verify the characteristic inequality of self-concordant functions (2.6): for any $(x, z) \in \operatorname{int} \mathcal{K}_\alpha$ and any $h = [\Delta x; \Delta z] \in \mathbb{R}^3$ we have to show that

$$|D^3 F_\alpha(x, z)[h, h, h]| \leq 2 \left(D^2 F_\alpha(x, z)[h, h]\right)^{3/2}.$$

It is easy to see that the absolute values on the left-hand side term can be dropped, because if

$$D^3 F_\alpha(x, z)[h, h, h] \leq 2 \left(D^2 F_\alpha(x, z)[h, h]\right)^{3/2}. \tag{3.1}$$

holds for *any* direction $h \in \mathbb{R}^3$, then it must hold in particular also for $-h$. In that case (3.1) becomes

$$D^3 F_\alpha(x, z)[-h, -h, -h] \leq 2 \left(D^2 F_\alpha(x, z)[-h, -h]\right)^{3/2}.$$

In combination with (3.1) this is exactly the same as the original inequality in Definition 2.2.3. Therefore, it suffices to show validity of (3.1) in order to prove self-concordance of F_α.

Let us define for $(x, z) \in \operatorname{int} \mathcal{K}_\alpha$

$$\xi(x) := x_1^\alpha x_2^{1-\alpha} > 0$$

and

$$\omega(\xi(x), z) = \xi(x) - \frac{z^2}{\xi(x)} > 0.$$

Then the barrier function can be written as

$$\begin{aligned}F_\alpha(x,z) &= -\log\left(\xi(x)^2 - z^2\right) - (1-\alpha)\log(x_1) - \alpha\log(x_2)\\ &= -\log\left(\xi(x)^2 - z^2\right) - \log(x_1) - \log(x_2) + \log(\xi(x))\\ &= -\log\left(\frac{\xi(x)^2 - z^2}{\xi(x)}\right) - \log(x_1) - \log(x_2)\\ &= \underbrace{-\log\left(\omega(\xi(x),z)\right)}_{=:\Phi(\omega(\xi(x),z))}\underbrace{-\log(x_1) - \log(x_2)}_{=:F(x)}.\end{aligned}$$

Denoting in the following $\omega = \omega(\xi(x),z)$, the directional derivatives of $F_\alpha(x,z)$ in direction h become

$$\begin{aligned}D_2 := D^2F_\alpha(x,z)[h,h] =& D^2\Phi(\omega)[h,h] + D^2F(x)[h,h],\\ D_3 := D^3F_\alpha(x,z)[h,h,h] =& D^3\Phi(\omega)[h,h,h] + D^3F(x)[h,h,h]\end{aligned}$$

Introducing the notation $\delta_i = \frac{\Delta x_i}{x_i}$ for $i = 1,2$, we get

$$D^2F(x)[h,h] = \left(\frac{\Delta x_1}{x_1}\right)^2 + \left(\frac{\Delta x_2}{x_2}\right)^2 = \delta_1^2 + \delta_2^2 =: t_2$$

and

$$D^3F(x)[h,h,h] \le 2(D^2F(x)[h,h])^{3/2} = 2t_2^{3/2}$$

because F is self-concordant.

Further, let us denote $\sigma_1 = \left(\frac{\omega'}{\omega}\right)^2$ and $\sigma_2 = -\frac{\omega''}{\omega}$. Then we compute the directional derivatives of Φ as

$$\begin{aligned}D\Phi(\omega)[h] &= -\frac{\omega'}{\omega}\\ D^2\Phi(\omega)[h,h] &= \underbrace{\left(\frac{\omega'}{\omega}\right)^2}_{=\sigma_1}\underbrace{-\frac{\omega''}{\omega}}_{=\sigma_2} \ge 0\\ D^3\Phi(\omega)[h,h,h] &= \underbrace{-2\left(\frac{\omega'}{\omega}\right)^3}_{\le 2\sigma_1^{3/2}} + 3\underbrace{\frac{\omega'\omega''}{\omega^2}}_{\le\sigma_1^{1/2}\sigma_2} - \frac{\omega'''}{\omega},\end{aligned}$$

where ω',ω'',ω''' denote the directional derivatives of ω at $(x,z) \in \operatorname{int}\mathcal{K}_\alpha^{(n)}$ in direction h. That means we have the following bound on $D^3\Phi$:

$$D^3\Phi(\omega)[h,h,h] \le 2\sigma_1^{3/2} + 3\sigma_1^{1/2}\sigma_2 - \frac{\omega'''}{\omega}.$$

In the following our aim is to bound the last term. If we manage to show

$$-\frac{\omega'''}{\omega} \le 3\,\sigma_2\,t_2^{1/2},$$

or equivalently

$$\omega''' - 3\,\omega''\,t_2^{1/2} \geq 0, \tag{3.2}$$

(using the definition of σ_2), then we get the following bound on $D^3\Phi$:

$$D^3\Phi(\omega)[h,h,h] \leq 2\sigma_1^{3/2} + 3\sigma_1^{1/2}\sigma_2 + 3\,\sigma_2\,t_2^{1/2}.$$

This implies

$$\begin{aligned} D_3 &= D^3\Phi(\omega(\xi(x),z))[h,h,h] + D^3F(x)[h,h,h] \\ &\leq 2\sigma_1^{3/2} + 3\sigma_1^{1/2}\sigma_2 + 3\sigma_2 t_2^{1/2} + 2t_2^{3/2} \\ &= (\sigma_1^{1/2} + t_2^{1/2})(2\sigma_1 - 2\sigma_1^{1/2}t_2^{1/2} + 2t_2 + 3\sigma_2). \end{aligned}$$

Note that it holds $D_2 = D^2F + D^2\Phi = \sigma_1 + \sigma_2 + t_2$. Using this identity, the last expression can be simplified and we get

$$\begin{aligned} D_3 &\leq (\sigma_1^{1/2} + t_2^{1/2})\left(3D_2 - (\sigma_1^{1/2} + t_2^{1/2})^2\right) \\ &\leq 2D_2^{3/2}. \end{aligned}$$

The last inequality is true because the maximum of $\tau(3D_2 - \tau^2)$ over nonnegative arguments is attained at $\tau = \sqrt{D_2}$ and equals $2D_2^{3/2}$.

That means validity of (3.2) for all $(x,z) \in \mathcal{K}_\alpha$ and all directions $h \in \mathbb{R}^3$ implies self-concordance of F_α. In order to show (3.2), we need to evaluate the derivatives of ω, which in turn depend on the derivatives of ξ.

Computing the directional derivatives

For any direction $h = [\Delta x; \Delta z] \in \mathbb{R}^3$ we denote by

$$\xi_k = D^k\xi(x)[\Delta x, \ldots, \Delta x]$$

the k-th directional derivative of ξ in direction Δx. We get the following recursion for ξ_k:

$$\begin{aligned} \xi_1 &= \xi_0\,\underbrace{(\alpha\delta_1 + (1-\alpha)\delta_2)}_{=:e_1(\alpha,\delta)} = \xi_0\,e_1(\alpha,\delta) \\ \xi_2 &= -\xi_0\,\underbrace{\alpha(1-\alpha)(\delta_1-\delta_2)^2}_{=:e_2(\alpha,\delta)} = -\xi_0\,e_2(\alpha,\delta) \leq 0 \\ \xi_3 &= -\xi_2\,\underbrace{[(2-\alpha)\delta_1 + (1+\alpha)\delta_2]}_{=-e_1(\alpha,\delta)+2(\delta_1+\delta_2)} = \xi_0\,\underbrace{e_2\left(-e_1(\alpha,\delta) + 2(\delta_1+\delta_2)\right)}_{=:e_3(\alpha,\delta)}. \end{aligned}$$

The partial derivatives of $\omega(\xi,z)=\xi-\frac{z^2}{\xi}$ are

$$\begin{aligned}
\nabla_z\omega &= -\frac{2z}{\xi_0}, & \nabla^3_{\xi\xi\xi}\omega &= \frac{6z^2}{\xi_0^4},\\
\nabla_\xi\omega &= 1+\frac{z^2}{\xi_0^2}, & \nabla^3_{\xi\xi z}\omega &= -\frac{4z}{\xi_0^3},\\
\nabla^2_{\xi\xi}\omega &= -\frac{2z^2}{\xi_0^3}, & \nabla^3_{\xi zz}\omega &= \frac{2}{\xi_0^2},\\
\nabla^2_{\xi z}\omega &= \frac{2z}{\xi_0^2}, & \nabla^3_{zzz}\omega &= 0,\\
\nabla^2_{zz}\omega &= -\frac{2}{\xi_0}.
\end{aligned}$$

Using the above results, we can compute now the derivatives of $\omega(\xi(x),z)=\xi(x)-\frac{z^2}{\xi(x)}$ in direction h. Denoting in the following $e_i=e_i(\alpha,\delta)$ for $i=1,2,3$, we get

$$\begin{aligned}
\omega' := D\omega[h] =& \nabla_\xi\omega\cdot\xi_1+\nabla_z\omega\cdot\Delta z\\
=&\left(1+\frac{z^2}{\xi_0^2}\right)\cdot\xi_0\cdot e_1-2\frac{z}{\xi_0}\Delta z\\
=&\left(\xi_0+\frac{z^2}{\xi_0}\right)e_1-2\frac{z\Delta z}{\xi_0}=\omega e_1+2\frac{z}{\xi_0}(ze_1-\Delta z),
\end{aligned}$$

$$\begin{aligned}
\omega'' := D^2\omega[h,h] =& \nabla^2_{\xi\xi}\omega\cdot\xi_1^2+\nabla_\xi\omega\cdot\xi_2+2\nabla^2_{\xi z}\omega\cdot\xi_1\cdot\Delta z+\nabla^2_{zz}\omega\cdot\Delta z^2\\
=&-2\frac{z^2}{\xi_0^3}\cdot\xi_1^2+4\frac{z}{\xi_0^2}\cdot\xi_1\cdot\Delta z-2\frac{1}{\xi_0}\cdot\Delta z^2+\left(1+\frac{z^2}{\xi_0^2}\right)\cdot\xi_2\\
=&-\frac{2}{\xi_0}\left(ze_1-\Delta z\right)^2-\left(\xi_0+\frac{z^2}{\xi_0}\right)\cdot e_2\le 0,
\end{aligned}$$

$$\begin{aligned}
\omega''' := D^3\omega[h,h,h] =& \nabla^3_{\xi\xi\xi}\omega\cdot\xi_1^3+3\nabla^3_{\xi\xi z}\omega\cdot\xi_1^2\cdot\Delta z+3\nabla^3_{\xi zz}\omega\cdot\xi_1\cdot\Delta z^2+3\nabla^3_{zzz}\omega\cdot\Delta z^3\\
&+3\nabla^2_{\xi\xi}\omega\cdot\xi_1\cdot\xi_2+3\nabla^2_{\xi z}\omega\cdot\xi_2\cdot\Delta z+\nabla_\xi\omega\cdot\xi_3\\
=&6\frac{z^2}{\xi_0^4}\cdot\xi_1^3-12\frac{z}{\xi_0^3}\cdot\xi_1^2\cdot\Delta z+6\frac{1}{\xi_0^2}\cdot\xi_1\cdot\Delta z^2\\
&-6\frac{z^2}{\xi_0^3}\cdot\xi_1\cdot\xi_2+6\frac{z}{\xi_0^2}\cdot\xi_2\cdot\Delta z+\left(1+\frac{z^2}{\xi_0^2}\right)\cdot\xi_3\\
=&6\frac{z^2}{\xi_0}\cdot e_1^3-12\frac{z}{\xi_0}\cdot e_1^2\cdot\Delta z+6\frac{1}{\xi_0}\cdot e_1\cdot\Delta z^2\\
&+6\frac{z^2}{\xi_0}\cdot e_1\cdot e_2-6\frac{z}{\xi_0}\cdot e_2\cdot\Delta z+\left(1+\frac{z^2}{\xi_0^2}\right)\cdot\xi_3\\
=&6\frac{e_1}{\xi_0}\left(ze_1-\Delta z\right)^2+6\frac{ze_2}{\xi_0}\left(ze_1-\Delta z\right)+\left(1+\frac{z^2}{\xi_0^2}\right)\cdot\xi_3.
\end{aligned}$$

Bounding the derivatives

Using the above expressions for ω'' and ω''', the left-hand side term of (3.2) becomes

$$\begin{aligned}\omega''' - 3\,\omega''\,t_2^{1/2} =& 6\frac{e_1}{\xi_0}\,(ze_1-\Delta z)^2 + 6\frac{ze_2}{\xi_0}\,(ze_1-\Delta z) + \left(1+\frac{z^2}{\xi_0^2}\right)\underbrace{\xi_3}_{=\xi_0\,e_3}\\ &+\frac{6}{\xi_0}\,(ze_1-\Delta z)^2\,t_2^{1/2} - 3\left(1+\frac{z^2}{\xi_0^2}\right)\underbrace{\xi_2}_{=-\xi_0\,e_2}\,t_2^{1/2}\\ =&\frac{1}{\xi_0}\left[6\,(e_1+t_2^{1/2})\,(ze_1-\Delta z)^2+6ze_2\,(ze_1-\Delta z)\right.\\ &\left.+(\xi_0^2+z^2)\,(e_3+3e_2t_2^{1/2})\right].\end{aligned}$$

Note that we have $e_3+3e_2t_2^{1/2} = e_2\,[-e_1+2(\delta_1+\delta_2)+3t_2^{1/2}] \geq 0$, since

$$e_1-2(\delta_1+\delta_2) = -\left\langle\begin{pmatrix}2-\alpha\\1+\alpha\end{pmatrix},\begin{pmatrix}\delta_1\\\delta_2\end{pmatrix}\right\rangle \leq \underbrace{\left\|\begin{matrix}2-\alpha\\1+\alpha\end{matrix}\right\|}_{\leq\sqrt{5}}\cdot\left\|\begin{matrix}\delta_1\\\delta_2\end{matrix}\right\| \leq \sqrt{5}\sqrt{t_2}\leq 3\sqrt{t_2}.$$

That means we get the bound

$$\begin{aligned}\omega''' - 3\,\omega''\,t_2^{1/2} =&\frac{1}{\xi_0}\left[6\,(e_1+t_2^{1/2})\,(ze_1-\Delta z)^2+6ze_2\,(ze_1-\Delta z)+z^2\,(e_3+3e_2t_2^{1/2})\right]\\ &+\underbrace{\xi_0\,(e_3+3e_2t_2^{1/2})}_{\geq 0}\\ \geq&\frac{1}{\xi_0}\underbrace{\left[6\,(e_1+t_2^{1/2})\,(ze_1-\Delta z)^2+6ze_2\,(ze_1-\Delta z)+z^2\,(e_3+3e_2t_2^{1/2})\right]}_{=:h(\alpha,\delta,z,\Delta z)}.\end{aligned}$$

Our goal is to show that h is nonnegative for all feasible combinations of α, δ, z and Δz. With a similar argument as above we get $e_1+t_2^{1/2}\geq 0$, since

$$-e_1 = \left\langle\begin{pmatrix}-\alpha\\-1+\alpha\end{pmatrix},\begin{pmatrix}\delta_1\\\delta_2\end{pmatrix}\right\rangle \leq \underbrace{\left\|\begin{matrix}\alpha\\1-\alpha\end{matrix}\right\|}_{\leq 1}\cdot\left\|\begin{matrix}\delta_1\\\delta_2\end{matrix}\right\| \leq t_2^{1/2}.$$

One sees that h is a quadratic form in the variables (v_1,v_2) with $v_1=ze_1-\Delta z$ and $v_2=z$ and the symmetric matrix

$$M=\begin{bmatrix}6\,(e_1+t_2^{1/2}) & 3e_2\\ 3e_2 & e_3+3e_2t_2^{1/2}\end{bmatrix}$$

If we manage to show that M is positive semidefinite, it follows that $h(\alpha,\delta,z,\Delta z)\geq 0$.

First, let us consider the situation where $e_1+t_2^{1/2}=0$. This is only possible in three cases:

1. $\delta = 0$,
2. $\delta_1 = 0$, $\delta_2 < 0$ and $\alpha = 0$,
3. $\delta_2 = 0$, $\delta_1 < 0$ and $\alpha = 1$.

In all three cases we have trivially that $h(\alpha, \delta, z, \Delta z) = 0$ because then $e_2 = 0$. This implies that $M = 0$.

Let us consider now the situation where $e_1 + t_2^{1/2} > 0$. Then the Schur complement of M with respect to the upper left component becomes

$$S = e_3 + 3e_2 t_2^{1/2} - 3e_2[6\,(e_1 + t_2^{1/2})]^{-1} 3e_2$$

We have that S is positive semidefinite if and only if

$$(e_3 + 3e_2 t_2^{1/2}) 6\,(e_1 + t_2^{1/2}) \geq (3e_2)^2. \tag{3.3}$$

We have seen above that e_3 can be factored using e_2, therefore the left-hand side expression of (3.3) can be written as $\left[e_2\,(-e_1 + 2(\delta_1 + \delta_2)) + 3e_2 t_2^{1/2}\right] 6\,(e_1 + t_2^{1/2})$. Thus, (3.3) is true if and only if

$$2(-e_1 + 2(\delta_1 + \delta_2) + 3t_2^{1/2})\,(e_1 + t_2^{1/2}) \geq 3e_2$$

Let us bring all terms in the above inequality on the left-hand side and denote

$$g(\alpha, \delta) = 2(e_1 + t_2^{1/2})(-e_1 + 2(\delta_1 + \delta_2) + 3t_2^{1/2}) - 3e_2.$$

In order to show that g is nonnegative, let us make the following change of variables:

$$\delta_1 = r\,\cos(\varphi), \qquad\qquad \delta_2 = r\,\sin(\varphi),$$

where $r \geq 0$ and $\varphi \in [0, 2\pi]$. We get then

$$\begin{aligned}
e_1 &= \alpha\delta_1 + (1-\alpha)\delta_2 = r\,[\alpha\cos(\varphi) + (1-\alpha)\sin(\varphi)]\,, \\
e_2 &= \alpha(1-\alpha)(\delta_1 - \delta_2)^2 = r^2\,\alpha(1-\alpha)\,(\cos(\varphi) - \sin(\varphi))^2\,, \\
t_2 &= r^2\cos(\varphi)^2 + r^2\sin(\varphi)^2 = r^2.
\end{aligned}$$

Substituting the above expressions in g, we get the following function in terms of α, r and φ:

$$\begin{aligned}
\tilde{g}(\alpha, r, \varphi) =& 2\,[r\,\,(\alpha\cos(\varphi) + (1-\alpha)\sin(\varphi)) + r] \\
& \cdot r\,[-\alpha\cos(\varphi) - (1-\alpha)\sin(\varphi) + 2(\cos(\varphi) + \sin(\varphi)) + 3] \\
& - 3r^2\alpha(1-\alpha)(\cos(\varphi) - \sin(\varphi))^2 \\
=& 2\,r^2\,[(\alpha\cos(\varphi) + (1-\alpha)\sin(\varphi)) + 1] \\
& \cdot\,\,[(2-\alpha)\cos(\varphi) + (1+\alpha)\sin(\varphi) + 3] \\
& - 3\,r^2\alpha(1-\alpha)(\cos(\varphi) - \sin(\varphi))^2.
\end{aligned}$$

We can omit the nonnegative coefficient r^2 and get a function only in terms of α and φ. Moreover, let us denote $u_1 = \cos(\varphi) + 1$ and $u_2 = \sin(\varphi) + 1$. We get

$$\begin{aligned}\frac{\tilde{g}(\alpha, r, \varphi)}{r^2} =& 2\ [\alpha\cos(\varphi) + (1-\alpha)\sin(\varphi) + 1] \cdot\ [(2-\alpha)\cos(\varphi) + (1+\alpha)\sin(\varphi) + 3] \\ & - 3\,\alpha(1-\alpha)(\cos(\varphi) - \sin(\varphi))^2 \\ =& 2\ [\alpha(u_1 - 1) + (1-\alpha)(u_2 - 1) + 1] \\ & \cdot [(2-\alpha)(u_1 - 1) + (1+\alpha)(u_2 - 1) + 3] - 3\,\alpha(1-\alpha)(u_1 - u_2)^2 \\ =& [2\alpha u_1 + 2(1-\alpha)u_2] \cdot\ [(2-\alpha)u_1 + (1+\alpha)u_2] \\ & - 3\,\alpha(1-\alpha)(u_1 - u_2)^2,\end{aligned}$$

which is a quadratic polynomial in u_1 and u_2 for each $\alpha \in [0, 1]$. By definition we have $u_i \geq 0$ for $i = 1, 2$. Furthermore, the coefficients for each monomial term are nonnegative too. Indeed, the latter polynomial can be written as

$$g_\alpha(u_1, u_2) = \beta_1 u_1^2 + \beta_2 u_2^2 + \beta_3 u_1 u_2,$$

where

- $\beta_1 = 2\alpha(2-\alpha) - 3\alpha(1-\alpha) = \alpha(4 - 2\alpha - 3 + 3\alpha) = \alpha(1+\alpha) \geq 0$,
- $\beta_2 = 2(1-\alpha)(1+\alpha) - 3\alpha(1-\alpha) = (1-\alpha)(2+2\alpha-3\alpha) = (1-\alpha)(2-\alpha) \geq 0$,
- $$\begin{aligned}\beta_3 &= 2\alpha(1+\alpha) + 2(1-\alpha)(2-\alpha) + 6\alpha(1-\alpha) \\ &= 2\alpha + 2\alpha^2 + 4 - 2\alpha - 4\alpha + 2\alpha^2 + 6\alpha - 6\alpha^2 \\ &= -2\alpha^2 + 2\alpha + 4 \geq 0,\end{aligned}$$

for all $\alpha \in [0, 1]$. That means in fact that g_α is a polynomial in nonnegative variables u_1 and u_2 with nonnegative coefficients. According to the above reasoning we conclude that $g(\alpha, \delta) \geq 0$ for all $\alpha \in [0, 1]$ and all $\delta \in \mathbb{R}^2$, which implies that (3.3) is valid. Thus, (3.2) is true and we have

$$-\frac{\omega'''}{\omega} \leq 3\,\sigma_2\, t_2^{1/2}.$$

In view of the calculations above, we have that F_α is self-concordant. This finishes the proof.

□

Remark 3.1.2. *During the proof we saw that $e_3 \geq -3e_2 t_2^{1/2}$ is a valid inequality for all feasible α and δ. When multiplying this inequality with $\xi(x) = x_1^\alpha x_2^{1-\alpha} > 0$, we get directly that $\xi_3 \geq 3\xi_2 t_2^{1/2}$. This implies that $\xi(x)$ is 1-compatible with*

$$F(x) = -\log(x_1) - \log(x_2)$$

with respect to the standard ordering relation induced by $\mathcal{K} = \mathbb{R}_+$ (see Definition 2.4.9).

In accordance with Theorem 2.4.11 we see that when taking $\mathcal{C}_2 = \{(y,z) : y \geq |z|\}$ *with the* 2*-self-concordant barrier* $\phi(y,z) = -\log(y^2 - z^2)$*, the function*

$$\bar{F}_\alpha(x,z) = -\log\left(x_1^{2\alpha} x_2^{2-2\alpha} - z^2\right) - \log(x_1) - \log(x_2)$$

is a 4*-self-concordant barrier for* $\mathcal{K}_\alpha$*, which is weaker than the result of Theorem 3.1.1. This barrier has been proposed e.g. by Nesterov in [47, Theorem 6].*

On the other hand, using the same theorem and taking instead $\mathcal{C}_2 = \{(y,z) : y \geq z\}$ *with the* 1*-self-concordant barrier* $\phi(y,z) = -\log(y-z)$*, we get that*

$$\hat{F}_\alpha(x,z) = -\log\left(x_1^{\alpha} x_2^{1-\alpha} - z\right) - \log(x_1) - \log(x_2)$$

is a 3*-self-concordant barrier for the hypograph of the geometric mean*

$$\mathcal{C} = \left\{(x,z) \in \mathbb{R}^n_+ \times \mathbb{R} : x_1^{\alpha} x_2^{1-\alpha} \geq z\right\}.$$

Note that this barrier was established e.g. by Nesterov in [47, Section 4].

3.1.2 Three conjectures for self-concordant barriers

High-dimensional power cone

Numerical tests suggest that Theorem 3.1.1 can be generalized to any dimension. For $\alpha \in \mathbb{R}^n$ such that $\alpha \geq 0$ and $\sum_{i=1}^n \alpha_i = 1$ we define the $(n+1)$-dimensional power cone

$$\mathcal{K}_\alpha^{(n)} := \left\{(x,z) \in \mathbb{R}^n_+ \times \mathbb{R} : \prod_{i=1}^n x_i^{\alpha_i} \geq |z|\right\}.$$

Then we conjecture that the function

$$F_\alpha^{(n)}(x,z) = -\log\left(\prod_{i=1}^n x_i^{2\alpha_i} - z^2\right) - \sum_{i=1}^n (1-\alpha_i)\log(x_i)$$

is an $(n+1)$-self-concordant barrier for the high-dimensional power cone $\mathcal{K}_\alpha^{(n)}$. It is easy to see that $F_\alpha^{(n)}$ is $(n+1)$-logarithmically homogeneous.

A possible proof could be similar to the one of Theorem 3.1.1. Analogously we denote for $(x,z) \in \operatorname{int} \mathcal{K}_\alpha^{(n)}$

$$\xi(x) := \prod_{i=1}^n x_i^{\alpha_i} > 0$$

and

$$\omega(\xi(x), z) = \xi(x) - \frac{z^2}{\xi(x)} > 0.$$

and the main task is to show that (3.2) holds, where ω'' and ω''' denote the second and third directional derivative in some direction $h = [\Delta x; \Delta z] \in \mathbb{R}^{n+1}$ for the ω

defined above, and $t_2 = \sum_{i=1}^n \delta_i^2$ with $\delta_i = \frac{\Delta x_i}{x_i}, i = 1, \ldots, n$. It turns out that the derivatives of ξ have a similar recursion as in the proof of Theorem 3.1.1, namely

$$\begin{aligned}
\xi_1 &= \xi_0 \underbrace{s_1}_{=:e_1} = \xi_0\, e_1 \\
\xi_2 &= -\xi_0 \underbrace{(s_2 - s_1^2)}_{=:e_2} = -\xi_0\, e_2 \\
\xi_3 &= \xi_0 \underbrace{(s_1^3 - 3s_1 s_2 + 2s_3)}_{=:e_3} = \xi_0\, e_3,
\end{aligned}$$

where

$$s_1 = \sum_{i=1}^n \alpha_i \delta_i, \qquad s_2 = \sum_{i=1}^n \alpha_i \delta_i^2, \qquad s_3 = \sum_{i=1}^n \alpha_i \delta_i^3.$$

We have checked numerically the validity of (3.2) in this case which would imply that indeed $F_\alpha^{(n)}$ is an $(n+1)$-self-concordant barrier for $\mathcal{K}_\alpha^{(n)}$.

A generalization of $\mathcal{K}_\alpha^{(n)}$

Numerical tests also suggest that Theorem 3.1.1 can be generalized to the case where z is a vector instead of a scalar. The generalization of $\mathcal{K}_\alpha^{(n)}$ would then become

$$\mathcal{K}_\alpha^{(n,m)} = \left\{ (x,z) \in \mathbb{R}_+^n \times \mathbb{R}^m : \prod_{i=1}^n x_i^{\alpha_i} \geq ||z||_2 \right\}$$

with the conjectured $(n+1)$-self-concordant barrier

$$\tilde{F}_\alpha(x,z) = -\log\left(\prod_{i=1}^n x_i^{2\alpha_i} - z^T z \right) - \sum_{i=1}^n (1-\alpha_i) \log(x_i).$$

Numerical tests of inequality (3.2) have been done with a random sampling of 10,000 points $(x,z) \in \operatorname{int} \mathcal{K}_\alpha^{(n,m)}$ and directions $(\Delta x, \Delta z)$ in dimension $n = 10$ and $m = 5$. The rest of the proof directly generalizes (z^2 has to be replaced by $z^T z$ and Δz^2 by $\Delta z^T \Delta z$). Note that the self-concordance parameter of $\tilde{F}_\alpha$ is independent from the value of m. However, we are not able to confirm that observation with an analytic proof.

On the other hand, the cone $\mathcal{K}_\alpha^{(n,m)}$ can be modelled in the following way: $(x,z) \in \mathcal{K}_\alpha^{(n,m)}$ if and only if

$$\begin{aligned}
(x, \tilde{z}) &\in \mathcal{K}_\alpha^{(n)} \\
(z, \tilde{z}) &\in \mathbb{L}^m,
\end{aligned} \tag{3.4}$$

where $\tilde{z}$ is an artificial modelling variable and $\mathbb{L}^m$ denotes the $(m+1)$-dimensional second order cone. Using the new self-concordant barrier for $\mathcal{K}_\alpha^{(n)}$ from Theorem 3.1.1 and the 2-self-concordant barrier for $\mathbb{L}^m$, we conclude that we can find a self-concordant barrier for (3.4) with parameter $\nu = n + 3$. Note again that the value of ν is independent from m.

An even better barrier for $\mathcal{K}_\alpha$

We mentioned earlier that for $n = 2$ and $\alpha_1 = \alpha_2 = \frac{1}{2}$ the power cone $\mathcal{K}_\alpha$ is a rotated second-order cone with the optimal barrier parameter of $\nu = 2$.

Numerical tests suggest that

$$\begin{aligned}\bar{F}_\alpha(x, z) = &- \log\left(\prod_{i=1}^{2} x_i^{2\alpha_i} - z^2\right) \\ &- \max\{1 - 2\alpha_1, 0\}\log(x_1) - \max\{1 - 2\alpha_2, 0\}\log(x_2)\end{aligned}$$

is a self-concordant barrier for $\mathcal{K}_\alpha$ with parameter $\nu = 3 - 2\min\{\alpha_1, \alpha_2\}$.

Note that in the symmetric case $\alpha_1 = \alpha_2 = \frac{1}{2}$ the last two terms vanish and the barrier becomes exactly the (optimal) barrier for the rotated second-order cone with parameter 2. For $\alpha_1 = 1$ (or $\alpha_2 = 1$) we get the barrier for the polyhedral limit cone with optimal parameter 3. Both cases are coherent with the theory. For $\alpha_1 \in (0, \frac{1}{2})$ (or $\alpha_2 \in (0, \frac{1}{2})$) we get a barrier whose parameter depends linearly on α and is sandwiched between 2 and 3.

3.1.3 Optimality of the new barrier

In the Theorem 3.1.1 we presented a new 3-self-concordant barrier for the three-dimensional power cone $\mathcal{K}_\alpha$. We have also conjectured that this result can be generalized to the $(n+1)$-dimensional power cone $\mathcal{K}_\alpha^{(n)}$ with an $(n+1)$-self-concordant barrier. We will show now that this barrier value is in fact "almost" optimal. First we need the following technical results

Theorem 3.1.3. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a convex set and $y \in \mathcal{C}$. Let $\{h_1, \dots, h_k\}$ be recession directions for $\mathcal{C}$ with coefficients $\{\lambda_i\}_{i=1}^k$ such that*

$$y - \lambda_i h_i \notin \operatorname{int} \mathcal{C}.$$

If for some $\{\delta_i\}_{i=1}^k$ it holds

$$y - \sum_{i=1}^{k} \delta_i h_i \in \mathcal{C},$$

then

$$\nu \geq \sum_{i=1}^{k} \frac{\delta_i}{\lambda_i}.$$

Proof. [46, Theorem 4.3.1]. □

Note that if $\mathcal{C} = \mathcal{K}$ is a proper cone, then any $h \in \mathcal{K}$ is also a recession direction.

Lemma 3.1.4. *Any self-concordant barrier for $\mathcal{K}_\alpha^{(n)}$ has a parameter ν such that*

$$\nu \geq n.$$

Proof. Let $n = 1$. Then

$$\mathcal{K}_\alpha^{(1)} = \{x \geq |z|\},$$

for which we know that the optimal barrier has a parameter of $\nu = 2 = n + 1 \geq n$.

For $n \geq 2$ we will use Theorem 3.1.3. Let us choose $y = (x, z) = [1, \dots, 1, 0] \in \mathbb{R}^{n+1}$ and $h_i = [e_i, 0] \in \mathcal{K}_\alpha^{(n)}$ where $e_i = [0, \dots, 0, 1, 0, \dots, 0]$, $i = 1, \dots, n$ is the i-th unit vector, and $\delta_i = \lambda_i = 1$, $i = 1, \dots, n$. We get

$$\nu \geq \sum_{i=1}^{n} \frac{1}{1} = n.$$

□

Note that the upper bound $n + 1$ for the self-concordance parameter can be tight. Indeed, if $\alpha = e_i$ for some $i \in \{1, \dots, n\}$, then the power cone becomes polyhedral

$$\mathcal{K}_\alpha^{(n)} = \{x_i \geq |z|\} \times \{x_j \geq 0\}, j \neq i.$$

We know that in this extreme case the optimal barrier has a parameter of $\nu = n+1$, which coincides with the parameter value that we have obtained. On the other hand, for $n = 2$ and $\alpha = [1/2, 1/2]$ we have the rotated second-order cone (with optimal barrier with parameter $\nu = n = 2$). In that case the lower bound is tight.

3.1.4 New barriers for certain convex sets

Epigraph of increasing power function

Note that by intersecting $\mathcal{K}_\alpha$ with $\{(x, z) : x_2 = 1\} \subset \mathbb{R}^3$ we get the epigraph of the increasing power function

$$\{x_1^{\alpha_1} \geq |z|\} \Leftrightarrow \{x_1 \geq |z|^p\}, \; \frac{1}{\alpha_1} = p \geq 1$$

with the self-concordant barrier

$$F_p(x_1, z) = -\log(x^{2/p} - z^2) - (1 - 1/p)\log(x_1)$$

with self-concordance parameter value of at most 3. This improves the previously known barrier with parameter 4 (cf. [46, Section 4.3.5.4]). It remains an open question whether F_p is already optimal or not.

Rotated positive power cone

Let $a_i \in \mathbb{R}^m$ and $\alpha_i \in \mathbb{R}^m$ such that $a_i \geq 0, \alpha_i \geq 0, \sum_{i=1}^n \alpha_i = 1$. In [64] Tunçel and Nemirovski consider the rotated (positive) power cone

$$\mathcal{C} = \left\{ (x, t) \in \mathbb{R}_+^n \times \mathbb{R}_+ : \prod_{i=1}^{m} \langle a_i, x \rangle^{\alpha_i} \geq t \right\}.$$

It is clear that $\mathcal{C}$ can be modelled using $\mathcal{K}_\alpha^{(n)}$. Indeed, $(x,t) \in \mathcal{C}$ if and only if

$$\begin{aligned} \prod_{i=1}^m y_i^{\alpha_i} &\geq t \\ y_i &= a_i^T x \\ t &\geq 0 \end{aligned}$$

or equivalently, $(Ax, t) \in \mathcal{K}_\alpha^{(m)}$, $t \in \mathbb{R}_+$. That means the self-concordance parameter would become $\nu = m+2$, and it would depend only on m. On the other hand, Tunçel and Nemirovski propose a barrier for $\mathcal{C}$ that only depends on n and whose self-concordance parameter is $\nu = 1 + \left(\frac{7}{3}\right)^2 n$ (see [64, Corollary 2.2]). One can see that it is beneficial to use our new barrier for $\mathcal{K}_\alpha^{(m)}$ to model $\mathcal{C}$ whenever the number of product terms m in $\mathcal{C}$ is not too large with respect to the size of x, i.e. when $m \leq \left(\frac{7}{3}\right)^2 \cdot n - 1 \approx 5n$.

Moreover, when modelling $\mathcal{C}$ with the power cone $\mathcal{K}_\alpha^{(n)}$ the coefficients a_i are not restricted to nonnegative vectors as it

Conic hull of epigraph of exponential function

Let us consider the convex cone

$$\mathcal{K}_{\exp} = \mathrm{cl}\left(\left\{ z_1 \in \mathbb{R}, z_2 \in \mathbb{R}_+, z_3 \in \mathbb{R}_{++} : \exp\left(\frac{z_1}{z_3}\right) \leq \frac{z_2}{z_3} \right\}\right).$$

We will see in Section 4.2 that $\mathcal{K}_{\exp}$ can be viewed as a limit of a linear transformation of the 3D power cone $\mathcal{K}_\alpha$, namely

$$\tilde{\mathcal{K}_\alpha} = \{(z_1, z_2, z_3) : z_2^\alpha z_3^{1-\alpha} \geq |z_3 + \alpha z_1|\},$$

for $\alpha \to 0$. In view of Theorem 3.1.1 and the self-concordance-preserving operations in Section 2.4.3 we conclude that

$$\begin{aligned} \tilde{F}_\alpha(z_1, z_2, z_3) &= F_\alpha(z_2, z_3, z_3 + \alpha z_1) \\ &= -\log\left(z_2^{2\alpha}\, z_3^{2(1-\alpha)} - (z_3 + \alpha z_1)^2\right) - (1-\alpha)\log z_2 - \alpha \log z_3 \end{aligned}$$

is a 3-self-concordant barrier for $\tilde{\mathcal{K}_\alpha}$. Letting $\alpha \to 0$, we see that the argument of the first log-term tends to 0. However, if we add to $\tilde{F}_\alpha$ the term $\log(2\alpha)$, the function is still 3-self-concordant for its domain $\mathrm{int}\,\tilde{\mathcal{K}_\alpha}$, since the derivatives of $\tilde{F}_\alpha$ are not affected by such a change. We get

$$\tilde{F}_\alpha(z_1, z_2, z_3) + \log(2\alpha) = -\log\underbrace{\left(\frac{z_2^{2\alpha}\, z_3^{2(1-\alpha)} - (z_3 + \alpha z_1)^2}{2\alpha}\right)}_{=:h(\alpha)=\frac{h_1(\alpha)}{h_2(\alpha)}} - (1-\alpha)\log z_2 - \alpha \log z_3.$$

We see that $h(\alpha)$ tends to an expression of the form $\frac{0}{0}$, as $\alpha \to 0$. Using the l'Hospital's rule, we get

$$\frac{h_1'(\alpha)}{h_2'(\alpha)} = \frac{z_3^2\, 2\, (\frac{z_2}{z_3})^{2\alpha} \log(\frac{z_2}{z_3}) - 2z_3 z_1 - 2\alpha z_1^2}{2} \to z_3^2 \log\left(\frac{z_2}{z_3}\right) - z_3 z_1.$$

Therefore, we conclude that

$$\begin{aligned}\lim_{\alpha\to 0}\left\{\tilde{F_\alpha}(z)+\log(2\alpha)\right\} &= -\log\left(z_3^2\log\left(\frac{z_2}{z_3}\right)-z_3z_1\right)-\log(z_2)\\ &= -\log\left(z_3\log\left(\frac{z_2}{z_3}\right)-z_1\right)-\log(z_2)-\log(z_3)\\ &= F_{\exp}(z),\end{aligned}$$

i.e. the function *values* of the 3-self-concordant barrier $\tilde{F_\alpha}(z)+\log(2\alpha)$ for $\tilde{\mathcal{K}_\alpha}$ converge to the function value of the known 3-self-concordant barrier $F_{\exp}$ for $\mathcal{K}_{\exp}$ (see e.g. [47, Section 4]). Note, however, that in general this observation alone is not sufficient to conclude that the limit function of a family of self-concordant barriers is also self-concordant. This is true if additionally the first three directional derivatives converge to the derivatives of the limit function.

3.2 Universal barrier for p-cone

For $p \geq 1$ and dimension $n \geq 1$ let us consider the p-cone

$$\mathcal{P}_p^{(n)} := \{(x,t) : t \geq ||x||_p\} \subset \mathbb{R}^{n+1}.$$

No simple barrier with low parameter is known for $\mathcal{P}_p^{(n)}$. For example Yue and Ye [66] use a general result by Nesterov and Nemirovski ([52, Proposition 5.1.4]) according to which, given a ν-self-concordant barrier F for some closed convex set $\mathcal{C}$ it is possible to compute a $2\theta^2\nu$-self-concordant barrier for the conic hull of $\mathcal{C}$ (see Section 2.4), where θ is some well-chosen positive number. In the case of $\mathcal{C}_p = \{x : ||x||_p \leq 1\}$ its conic hull is exactly the p-cone $\mathcal{P}_p^{(n)}$. In [66, Theorem 3.1] the authors show that for $\mathcal{C}_p$ we can take $\theta = 5$ and obtain a $\tilde{\nu}$-self-concordant barrier for $\mathcal{P}_p^{(n)}$ with $\tilde{\nu} = 50\nu$. In [20, Theorem 4] the authors improve the result by [52] and show that, given a ν-self-concordant barrier for $\mathcal{C}$ it is possible to compute a $\bar{\nu}$-self-concordant barrier for the conic hull of $\mathcal{C}$ with $\bar{\nu} < 25\,\nu$.

Both approaches rely on the conic hull of the p-unit ball $\mathcal{C}_p$. The p-unit ball, on the other hand, can be modelled using n epigraphs of p-powers (see Section 4.1). The epigraphs of p-powers admit a 3-self-concordant barrier, as described in Section 3.1.4[1]. Using the approach of Freund et al. [20] or Xue and Ye [66], we can ultimately derive a ν-self-concordant barrier for $\mathcal{P}_p^{(n)}$ with $\nu < 75n$ (respectively $\nu = 150n$).

Another way of modelling the p-cone has been proposed e.g. by Nesterov in [49]. That modelling approach uses the power cone $\mathcal{K}_\alpha$, and it is described in detail in Section 4.1. The proposed decomposition yields a ν-self-concordant barrier for $\mathcal{P}_p^{(n)}$ with $\nu = 3n$. This significant reduction in terms of the barrier parameter is due to the fact that the modelling of $\mathcal{P}_p^{(n)}$ using $\mathcal{K}_\alpha$ is not based on the modelling of the (nonhomogeneous) set $\mathcal{C}_p$. Therefore we do not need to apply the expensive operation of taking the conic hull, that results in an increase of the parameter by a factor of 25.

[1] At the time the best-known parameter for epi($|x|^p$) was $\nu = 4$.

3.2.1 Universal barrier and characteristic function

We recall that every convex set $\mathcal{C} \subseteq \mathbb{R}^n$ admits a so-called *universal barrier* Φ with self-concordance parameter of $\nu = \mathcal{O}(n)$ (see Theorem 2.4.6). Unfortunately, the definition of the universal barrier at a point x involves the computation of the volume of the polar set at the point x. Therefore, it is in general not possible to write down a closed form description of the universal barrier. For the conic setting, Güler established a link between the universal barrier of a cone $\mathcal{K}$ and its characteristic function.

Definition 3.2.1. *The characteristic function $\varphi : \operatorname{int} \mathcal{K} \to \mathbb{R}$ of a cone $\mathcal{K} \subseteq \mathcal{E}$ is defined as*

$$\zeta(x) = \int_{\mathcal{K}^*} e^{-\langle s,x \rangle}\, ds. \tag{3.5}$$

Theorem 3.2.2. *Let $\mathcal{K} \subseteq \mathbb{R}^n$ be a proper cone and ζ its characteristic function. Then the following equality holds:*

$$\log \zeta(x) = \log(\mathrm{vol}_n(\mathcal{C}^0(x)))) + \log n!,$$

where $\mathcal{C}^0(x)$ denotes the polar set centered at $x \in \operatorname{int} \mathcal{K}$

Proof. [28, Theorem 4.1]. □

In view of Theorem 2.4.6 the term on the right-hand side is essentially the universal barrier for $\mathcal{K}$. Note, however, that the "true" universal barrier is given by

$$\Phi(x) = \mathcal{O}(1) \, \log(\mathrm{vol}_n(\mathcal{C}^0(x)))),$$

that is, we might have to find a constant $\kappa = \mathcal{O}(1)$ to scale the barrier in order to make it self-concordant.

It follows that, if we omit the constant term $\log n!$, we get according to Theorem 3.2.2 that the universal barrier for $\mathcal{K}$ is given by

$$\Phi(x) = \kappa \log(\zeta(x)),$$

for some constant κ. In other words, if we are able to evaluate the characteristic function and its derivatives, we have a means to compute the universal barrier. Moreover, if ζ is homogeneous of degree $-\theta$, then we conclude in view of Theorem 2.5.6 that the universal barrier Φ is a logarithmically homogeneous self-concordant barrier for $\mathcal{K}$ with parameter $\nu = \kappa\, \theta$.

In some cases it is possible to compute the characteristic function analytically. Some examples are given in [28], including the characteristic function for the second-order cone (which is in fact $\mathcal{P}_2^{(n)}$). In that case Güler showed that by the construction above we obtain a scaled version of the optimal self-concordant barrier for the second-order cone (see [28, Lemma 7.1]). This means in fact that it might be possible that the constant κ is less than 1. Note that in view of the self-concordance preserving operations in Section 2.4.3 scaling a ν-self-concordant barrier F with a constant $\kappa \geq 1$ yields a $\kappa\nu$-self-concordant barrier. However, for $\kappa < 1$ this is in general not true. In that case we have to check whether the self-concordance property in Definition 2.2.3 is still true for κF. In the case of the second order cone scaling the universal barrier with $\kappa < 1$ was indeed possible without violating Definition 2.2.3.

In the case of the p-cone $\mathcal{P}_p^{(n)}$ this above result could be useful because it is well-known[2] that

$$\left(\mathcal{P}_p^{(n)}\right)^* = \mathcal{P}_q^{(n)} = \{s = (z, \tau) : ||s||_q \leq \tau\},$$

where p and q are conjugate exponents, that is $\frac{1}{p} + \frac{1}{q} = 1$. That means we have an explicit description of the dual cone and we might try to compute the characteristic function ζ.

3.2.2 Computation of the characteristic function

For the rest of this section we consider the p-cone with $n = 2$. For sake of simplifying notation we denote in the following $\mathcal{K} = \mathcal{P}_p^{(2)}$. Then

$$\begin{aligned}(z, \tau) \in (\mathcal{K})^* &\Leftrightarrow |z_1|^q + |z_2|^q \leq \tau^q, \tau \geq 0 \\ &\Leftrightarrow \left|\frac{z_1}{\tau}\right|^q + \left|\frac{z_2}{\tau}\right|^q \leq 1, \tau > 0.\end{aligned}$$

Let us consider therefore the change of variables

$$\begin{aligned} z_1 &= u\,\tau \\ z_2 &= v\,\tau. \end{aligned}$$

The functional determinant is

$$\frac{D(z_1, z_2, \tau)}{D(u, v, \tau)} = \tau^2,$$

and defining $B_q = \{(u, v) : |u|^q + |v|^q \leq 1\}$, the integral (3.5) becomes

$$\begin{aligned}\zeta(x) = \int_{(\mathcal{K})^*} e^{-\langle s,x\rangle} ds &= \int_{(u\tau, v\tau, \tau)\in(\mathcal{K})^*} e^{-x_1 u\tau - x_2 v\tau - x_3\tau}\ \tau^2\, d\tau\, du\, dv \\ &= \int_{(u,v)\in B_q} \int_{\tau=0}^{\infty} e^{-\tau(x_1 u + x_2 v + x_3)}\ \tau^2\, d\tau\, du\, dv.\end{aligned}$$

The inner integral can be computed as

$$\int_{\tau=0}^{\infty} e^{-\tau(x_1 u + x_2 v + x_3)}\ \tau^2\, d\tau = \frac{2}{(ux_1 + vx_2 + x_3)^3}.$$

So it remains to compute

$$2\int_{(u,v)\in B_q} (ux_1 + vx_2 + x_3)^{-3}\ du\, dv.$$

[2] see e.g. [66, Section 2]

Let us apply the following change of variables:

$$\begin{aligned} u &= r_q(\theta, r)\, \cos\theta \\ v &= r_q(\theta, r)\, \sin\theta, \end{aligned}$$

where

$$r_q(\theta, r) = \frac{r}{\sqrt[q]{|\cos\theta|^q + |\sin\theta|^q}} = \frac{r}{g(\theta)},$$

and $g(\theta) = ||\cos\theta \quad \sin\theta||_q$. The functional determinant becomes

$$\frac{D(u,v)}{D(r,\theta)} = r \cdot (|\cos\theta|^q + |\sin\theta|^q)^{-\frac{2}{q}} = r \cdot g(\theta)^{-2}.$$

That means $(u, v) \in B_q$ if and only if $\theta \in [0, 2\pi]$ and $r \in [0, 1]$. So the characteristic function becomes

$$\begin{aligned} \zeta(x) &= 2 \int_{(u,v)\in B_q} (ux_1 + vx_2 + x_3)^{-3} \; du\, dv \\ &= 2 \int_{\theta=0}^{2\pi} \int_{r=0}^{1} \frac{r \cdot g(\theta)^{-2}}{(x_1 \cdot r_q(\theta, r) \cdot \cos\theta + x_2 \cdot r_q(\theta, r) \cdot \sin\theta + x_3)^3} \, dr\, d\theta \\ &= 2 \int_{\theta=0}^{2\pi} g(\theta)^{-2} \int_{r=0}^{1} \frac{r}{[\frac{r}{g(\theta)} \cdot (x_1 \cdot \cos\theta + x_2 \cdot \sin\theta) + x_3]^3} \, dr\, d\theta \\ &= 2 \int_{\theta=0}^{2\pi} g(\theta) \int_{r=0}^{1} \frac{r}{[r \cdot (x_1 \cdot \cos\theta + x_2 \cdot \sin\theta) + g(\theta) \cdot x_3]^3} \, dr\, d\theta \\ &= \frac{1}{x_3} \int_{\theta=0}^{2\pi} \frac{1}{(x_1 \cdot \cos\theta + x_2 \cdot \sin\theta + g(\theta) \cdot x_3)^2} \, d\theta \\ &= \frac{1}{x_3^3} \int_{\theta=0}^{2\pi} \frac{1}{(f(\theta, x) + g(\theta))^2} \, d\theta, \end{aligned}$$

where $f(\theta, x) = \frac{x_1}{x_3} \cdot \cos\theta + \frac{x_2}{x_3} \cdot \sin\theta$.

Note that we have

$$\zeta(\lambda\, x) = \frac{1}{\lambda^3 x_3^3} \int_0^{2\pi} \frac{d\theta}{(f(\theta, x) + g(\theta))^2} = \frac{1}{\lambda^3}\, \zeta(x),$$

which means that ζ is homogeneous of degree -3. Accordingly, the unscaled universal barrier

$$\Phi_u(x) = \log\left(\zeta(x)\right)$$

is a logarithmically homogeneous function of degree $\theta = 3$.

Even though Φ_u is not in closed form at hand, we can compute its derivatives with respect to x. We get the gradient and Hessian

$$\begin{aligned} \nabla\Phi_u(x) &= \frac{1}{\zeta(x)}\; \nabla\zeta(x) \\ \nabla^2\Phi_u(x) &= \frac{1}{\zeta(x)} \cdot \nabla^2\zeta(x) - \frac{\nabla\zeta(x) \cdot \nabla\zeta(x)^T}{\zeta(x)^2}, \end{aligned}$$

Let us denote $I(\theta,x) = \frac{1}{(f(\theta,x)+g(\theta))^2}$. Then

$$\begin{aligned}
\zeta(x) &= \frac{1}{x_3^3}\int_{\theta=0}^{2\pi} I(\theta,x)\,d\theta \\
\nabla\zeta(x) &= \begin{bmatrix} 0 \\ 0 \\ -\frac{3}{x_3^4}\cdot\int_0^{2\pi} I(\theta,x)\,d\theta \end{bmatrix} + \frac{1}{x_3^3}\cdot\int_0^{2\pi}\nabla I(\theta,x)\,d\theta \\
\nabla^2\zeta(x) &= \frac{12}{x_3^5}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \int_0^{2\pi} I(\theta,x)\,d\,\theta \end{bmatrix} + \frac{1}{x_3^3}\cdot\int_0^{2\pi}\nabla^2 I(\theta,x)\,d\,\theta \\
&\quad -\frac{3}{x_3^4}\begin{bmatrix} 0 & 0 & \int_0^{2\pi}\frac{\partial I}{\partial x_1}\,d\,\theta \\ 0 & 0 & \int_0^{2\pi}\frac{\partial I}{\partial x_2}\,d\,\theta \\ \int_0^{2\pi}\frac{\partial I}{\partial x_1}\,d\,\theta & \int_0^{2\pi}\frac{\partial I}{\partial x_2}\,d\,\theta & 2\cdot\int_0^{2\pi}\frac{\partial I}{\partial x_3}\,d\,\theta \end{bmatrix}.
\end{aligned} \tag{3.6}$$

3.2.3 Finding the scale

We showed that the unscaled universal barrier Φ_u is a logarithmically homogeneous function of degree $\theta = 3$. In view of the observations above we have that the actual universal barrier is given by

$$\Phi(x) = \kappa\,\Phi_u(x),$$

with self-concordance parameter $\nu = 3\kappa$, for all $p \geq 1$.

As we have pointed out, we still have to find a constant κ, as small as possible, so that $\Phi = \kappa\,\Phi_u$ is a self-concordant function. Then in view of Theorem 2.5.6 Φ is a 3κ-self-concordant barrier for $\mathcal{K}$.

According to Renegar (see [55, Section 2.2]) a function $F : \operatorname{int}\mathcal{K} \subset \mathbb{R}^n \to \mathbb{R}$ is self-concordant if for all $x \in \operatorname{int}\mathcal{K}$, $y \in D_0(x,1) = \{y : ||y-x||_x < 1\}$ and for all $v \neq 0$ the following two inequalities are satisfied

$$1 - ||y-x||_x \leq \frac{||v||_y}{||v||_x} \leq \frac{1}{1-||y-x||_x}. \tag{3.7}$$

By scaling F with a parameter κ it follows directly that the Hessian of $(\kappa\,F)$ at any $x \in \operatorname{int}\mathcal{K}$ becomes $\nabla^2(\kappa\,F)(x) = \kappa\nabla^2 F(x)$. This implies that the local norm of any $z \in \mathbb{R}^n$ in terms of the scaled Hessian at x becomes

$$||z||_{\nabla^2(\kappa\,F)(x)} = \langle\nabla^2(\kappa\,F)(x)z, z\rangle^{1/2} = \sqrt{\kappa}\,||z||_{\nabla^2 F(x)}.$$

If we apply this observation to (3.7), we see that the term in the middle is unaffected by a scaling of F, i.e.

$$\frac{||v||_{\nabla^2(\kappa\,F)(y)}}{||v||_{\nabla^2(\kappa\,F)(x)}} = \frac{\sqrt{\kappa}||v||_{\nabla^2 F(y)}}{\sqrt{\kappa}||v||_{\nabla^2 F(x)}} = \frac{||v||_y}{||v||_x}.$$

On the other hand, the norm terms $||y-x||_x$ do depend on the scaling of F. We have

$$||y-x||_{\nabla^2(\kappa\,F)(x)} = \sqrt{\kappa}||y-x||_{\nabla^2 F(x)} = \sqrt{\kappa}||y-x||_x.$$

That means by scaling F with a constant κ the inequalities (3.7) are equivalent to

$$1-\sqrt{\kappa}||y-x||_x \leq \frac{||v||_y}{||v||_x} \leq \frac{1}{1-\sqrt{\kappa}||y-x||_x}. \tag{3.8}$$

We see that if $\kappa \geq 1$, then (3.8) is less restrictive than (3.7), since $1-\sqrt{\kappa}||y-x||_x \leq 1-||y-x||_x$. That means, if (3.7) is satisfied for all $x \in \mathcal{K}$, all $y \in D_0(x,1)$ and all $v \neq 0$, then automatically (3.8) is true too. Conversely, if $\kappa < 1$, then the two inequalities in (3.8) become tighter than the inequalities in (3.7).

From the right-hand side inequality of (3.8), we can remove the dependence on v by finding an upper bound on $\frac{||v||_y}{||v||_x}$. To simplify notation let us denote for any $x \in \operatorname{int}\mathcal{K}$ the Hessian of F at x by $H(x) = \nabla^2 F(x)$. We get

$$\begin{aligned}\left(\max_{v\neq 0}\frac{||v||_y}{||v||_x}\right)^2 &= \max_{v\neq 0}\frac{||v||_y^2}{||v||_x^2} = \max_{v\neq 0}\frac{v^T H(y) v}{v^T H(x) v} = \max_{v\neq 0}\frac{v^T H(y) v}{\underbrace{v^T H(x)^{\frac{1}{2}}}_{w^T}\cdot\underbrace{H(x)^{\frac{1}{2}}v}_{w\neq 0}}\\ &= \max_{w\neq 0}\frac{w^T H(x)^{-\frac{1}{2}}H(y)H(x)^{-\frac{1}{2}}w}{||w||^2} = \max_{w\neq 0}\frac{||H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}w||^2}{||w||^2}\\ &= \left(\max_{w\neq 0}\frac{||H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}w||}{||w||}\right)^2 = ||H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}||^2,\end{aligned}$$

where the matrix norm in the last expression is induced by the Euclidean norm $||\cdot||_2$. Let us denote

$$\sigma_1(x,y) := ||H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}||.$$

Similarly we consider the lower bound on $\frac{||v||_y}{||v||_x}$, or alternatively an upper bound on $\frac{||v||_x}{||v||_y}$. We get

$$\sigma_2(x,y) := \max_{v\neq 0}\frac{||v||_x}{||v||_y} = ||H(x)^{\frac{1}{2}}H(y)^{-\frac{1}{2}}||.$$

Note that $\sigma_1(x,y)$ is the largest singular value of the matrix $H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}$. On the other hand $\sigma_2(x,y)$ is the largest singular value of $H(x)^{\frac{1}{2}}H(y)^{-\frac{1}{2}} = \left[H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}\right]^{-1}$, or equivalently, the inverse of the smallest singular value of $H(y)^{\frac{1}{2}}H(x)^{-\frac{1}{2}}$.

Our aim is to find the smallest κ such that the inequalities (3.8) are valid. Using the above reformulations, we see that this is equivalent to finding the smallest κ such that

$$\max\{\sigma_1(x,y),\sigma_2(x,y)\} \leq \frac{1}{1-\sqrt{\kappa}\,||y-x||_x}$$

for $x \in \operatorname{int}\mathcal{K}$, $y \in D_0\left(x, 1/\sqrt{\kappa}\right)$, which is the same as

$$\kappa \geq \underbrace{\left(\frac{1-\frac{1}{\max(\sigma_1,\sigma_2)}}{||y-x||_x}\right)^2}_{=:\psi(x,y)}.$$

Let us denote

$$\varphi(\kappa) := \sup_{\substack{x\in \operatorname{int}\mathcal{K} \\ y\in D_0\left(x,\frac{1}{\sqrt{\kappa}}\right)}} \psi(x,y). \tag{3.9}$$

We see that φ is non-increasing in κ, because the radii $r = \frac{1}{\sqrt{\kappa}}$ decrease as κ increases, and as a consequence we are taking suprema over smaller and smaller sets. Our goal is to find the smallest κ such that

$$\kappa \geq \varphi(\kappa). \tag{3.10}$$

This situation is illustrated in Figure 3.1.

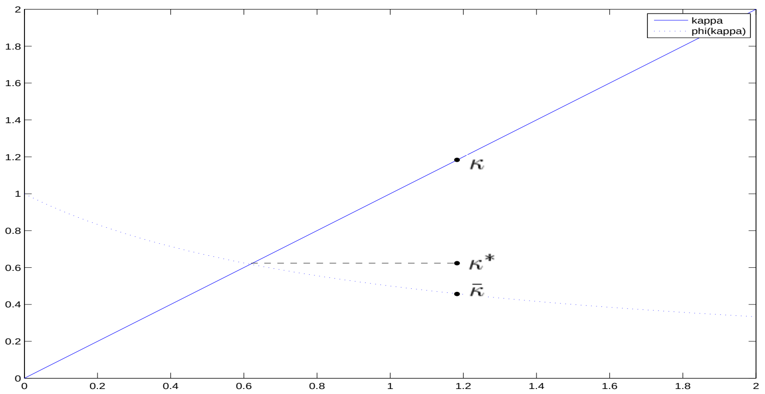


Figure 3.1: solid: κ, dashed: $\varphi(\kappa)$.

We denote by κ^* the smallest value of κ that satisfies (3.10). Note that in order to evaluate φ we need to solve the subproblem (3.9). Therefore we cannot compute explicitly κ^* but we can find it numerically by applying the following simple bisection-like algorithm

For any $\kappa > 0$ we define $r(\kappa) = \frac{1}{\sqrt{\kappa}}$, and for an initial κ_0 (say $\kappa_0 = 1$) we denote $r_0 = r(\kappa_0)$ we consider the problem

$$\bar{\kappa} := \sup_{\substack{x\in \operatorname{int}\mathcal{K} \\ y\in D_0(x,r_0)}} \psi(x,y). \tag{3.11}$$

If $\bar{\kappa} < \kappa_0$, then κ_0 is an upper bound on κ^* (because κ_0 satisfies (3.10)). On the other hand $\bar{\kappa}$ is a lower bound on κ^*, because $\bar{\kappa} < \kappa_0$ implies $\bar{r} = r(\bar{\kappa}) > r(\kappa_0) = r_0$, which means

$$\bar{\kappa} = \sup_{\substack{x\in \operatorname{int}\mathcal{K} \\ y\in D_0(x,r_0)}} \psi(x,y) \leq \sup_{\substack{x\in \operatorname{int}\mathcal{K} \\ y\in D_0(x,\bar{r})}} \psi(x,y).$$

In other words $\bar{\kappa}$ is a lower bound on κ^*. In fact, $\bar{\kappa}$ might be equal to κ^* if the inequality above is tight.

The goal is to find a κ^+ that is smaller than κ_0 but still greater than $\bar{\kappa}$ and that satisfies (3.10). We define κ^+ simply as a convex combination between κ_0 and $\bar{\kappa}$ and check if (3.10) is satisfied. This procedure is summarized in Algorithm 6.

Algorithm 6 Bisection method to find optimal scaling parameter κ

```
Initialize: $\kappa_0 = 1$, $\bar{\kappa}_0 = 1$, $r_0 = \frac{1}{\sqrt{\kappa_0}}$, $k = 0$, $\gamma \in (0,1)$.
  while $\kappa_k - \bar{\kappa}_k \geq 0$ do
    $\bar{\kappa}_k := \sup_{\substack{x\in\mathcal{K} \\ y\in D(x,r_k)}} \psi(x,y)$
    if $\bar{\kappa}_k < \kappa_k$ then
      $\kappa_{k+1} := \gamma\bar{\kappa}_k + (1-\gamma)\kappa_k$
      $r_{k+1} = \frac{1}{\sqrt{\kappa_k}} \quad (\geq 1)$
      $k = k + 1$
    else
      return $\kappa_{k-1}$
    end if
  end while
```

Note that Algorithm 6 generates a decreasing sequence of upper bounds κ_k (because as long as the algorithm runs it takes as new value for κ_{k+1} the convex combination of $\bar{\kappa}_k$ and the previous κ_k). As a result the sequence of radii r_k as well as the sequence of $\bar{\kappa}_k$ will be increasing (the latter because we take the supremum over larger and larger sets of the same function). The return value will be the last computed valid upper bound for the scaling coefficient.

So far we have assumed that the subproblem (3.11) can be solved exactly. As we mentioned before, (3.11) involves implicitly the value of κ that we are looking for. Therefore we propose the following way of approximating the solution of (3.11). For the current κ we generate a random sample of points $x \in \operatorname{int}\mathcal{K}$ and $y \in D_0(x,r)$ and compute the supremum of ψ over that sampling set $D_r \subseteq \{(x,y) : x \in \operatorname{int}\mathcal{K}, y \in D_0(x,r)\}$. Thus, we obtain only a *lower bound* $\hat{\kappa}$ on the actual $\bar{\kappa}$ because

$$\hat{\kappa} = \sup_{(x,y)\in D_r} \psi(x,y) \leq \sup_{\substack{x\in\operatorname{int}\mathcal{K} \\ y\in D_0(x,r)}} \psi(x,y) = \bar{\kappa}.$$

If now $\hat{\kappa} < \kappa$, we conclude that also $\bar{\kappa} < \kappa$ (which is not necessarily true), and we accept κ as a valid upper bound. Therefore, the sequence of upper bounds κ that we compute is *uncertain*, i.e. we might think that κ is a valid scaling coefficient that satisfies (3.10) because we observe that $\hat{\kappa} < \kappa$. Moreover, the value of ψ can only be evaluated *approximately*, because it involves the computation of the Hessian of the universal barrier, which is defined in (3.6) using some integrals that are computed numerically.

We have implemented the above scheme that computes in each iteration an approximation $\hat{\kappa}$ for the subproblem (3.11) for different values of p. We have sampled 10 random points x in the cone $\mathcal{K}$ and for each x 100 random points y in the corresponding scaled Dikin ellipsoid around x. In Tables 3.1 we see the approximate values of the optimal κ.

p	κ	$\frac{p}{p+1}$
2	0.666666	0.667
3	0.752176	0.750
4	0.801353	0.800
5	0.834602	0.833
6	0.859251	0.857
7	0.876871	0.875
8	0.893245	0.889
9	0.904193	0.900
10	0.910155	0.909

p	κ	$\frac{p}{2p-1}$
2	0.666666	0.667
3/2	0.746877	0.750
4/3	0.794590	0.800
5/4	0.823972	0.833
6/5	0.859855	0.857
7/6	0.871697	0.875
8/7	0.874634	0.889
9/8	0.897667	0.900
10/9	0.909984	0.909

Table 3.1: Scaling coefficients for the universal barrier for the 3D p-cone.

We see that for $p = 2$ we obtain a scaling factor of $\kappa = \frac{2}{3}$, which is in fact the optimal scaling factor for the universal barrier for the second order cone (see [28, Lemma 7.1]). That means in that case we obtain a self-concordance parameter of $\nu = 2$. On the other hand for large values of p and for $p \to 1$ the value of κ tends to 1. This effect agree with the theory, since for $p = 1$ and $p = \infty$ the p-cone is a polyhedral cone with optimal self-concordance parameter or $\nu = 3$. In these two extreme cases it cannot be possible to reduce the self-concordance parameter by scaling the universal barrier with a $\kappa < 1$. Furthermore, we observe a strong resemblance of the obtained values for κ with the function $\varphi_1(p) = \frac{p}{p+1}$ for $p \geq 2$ and $\varphi_2(p) = \frac{p}{2p-1}$ for $p \in [1, 2]$ (see Figure 3.2).

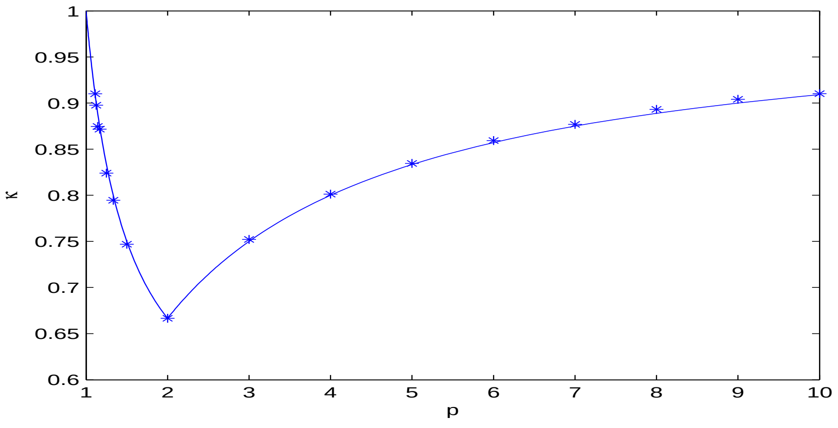


Figure 3.2: Scaling parameter κ for universal barrier for $\mathcal{P}_p^{(2)}$ vs. p. The numerically obtained values are indicated using the stars (*), the plotted functions are $\varphi_1(p) = \frac{p}{p+1}$ for $p \geq 2$ and $\varphi_1(p) = \frac{p}{2p-1}$ for $p \in [0, 1]$.

Based on our numerical experiments we conjecture that the universal barrier for the three-dimensional p-cone $\mathcal{P}_p^{(2)}$ is given by

$$\Phi(x,t) = \varphi(p)\,\Phi_u(x,t),$$

where $\Phi_u(x,t) = \log(\zeta(x,t))$ and $\zeta(x,t)$ is the characteristic function of $\mathcal{P}_p^{(2)}$. The scaling coefficient is given by

$$\varphi(p) = \begin{cases} \frac{p}{p+1}, & p \geq 2 \\ \frac{p}{2p-1}, & p \in [1,2]. \end{cases}$$

Moreover, the optimal self-concordance parameter of $\Phi(x,t)$ is $\nu = 3\,\varphi(p) \in [2,3]$.

CHAPTER 4

Modelling with the power cone

In this chapter we consider the so-called *power cone* $\mathcal{K}_\alpha$ in dimension 3 and we discuss its scope, i.e. we examine which constraints can be expressed using $\mathcal{K}_\alpha$. For $\alpha \in [0,1]$ the power cone is defined as

$$\mathcal{K}_\alpha := \left\{(x,z) \in \mathbb{R}^2_+ \times \mathbb{R} : x_1^\alpha x_2^{1-\alpha} \geq |z|\right\} \subset \mathbb{R}^3.$$

We start with the observation from the previous chapter that for $\alpha = \frac{1}{2}$ the cone $\mathcal{K}_\alpha$ is exactly the rotated second order cone in dimension 3. For the other two extreme cases ($\alpha \in \{0,1\}$) we obtain a polyhedral cone, e.g. for $\alpha = 0$, we get

$$\mathcal{K}_0 = \{(x_1,x_2,z) : x_1 \geq 0, x_2 \geq |z|\} = \{(x_1,x_2,z) : x_1 \geq 0, x_2 \geq z, x_2 \geq -z\}.$$

In Section 2.5.2 we saw that these three cones are in fact symmetric cones (see Definition 2.5.10).

For all other values of $\alpha \in (0,1)$ the cone $\mathcal{K}_\alpha$ is nonsymmetric, and therefore not applicable to practically efficient symmetric primal-dual interior-point methods. However, we showed in Section 2.5.3 that nonsymmetric conic problems can be solved using a primal-dual predictor-corrector method (Algorithm 5), provided that a self-concordant barrier for the dual cone is available. Moreover, we showed in Theorem 3.1.1 that

$$F_\alpha(x,z) = -\log(x_1^{2\alpha}x_2^{2-2\alpha} - z^2) - (1-\alpha)\log(x_1) - \alpha\log(x_2)$$

is a 3-self-concordant barrier for $\mathcal{K}_\alpha$. If we have a dual formulation (D) with a cone $\mathcal{K}^*$ that is a direct product of power cones, then we can derive in view of Section 2.4.3 a ν-self-concordant barrier for $\mathcal{K}^*$, where $\nu = 3N$ and N is the number of power cones in the (D). The complexity of solving the dual problem will then be proportional to $\sqrt{3N}\log(1/\epsilon)$ (see Theorem 2.4.14 and Theorem 2.5.18). That means all convex optimization problems that can be reformulated in dual conic form, where the dual cone is a direct product of power cones $\mathcal{K}_\alpha$, can be efficiently solved using dual or primal-dual interior-point methods.

The main objective of this chapter is therefore the description of the scope of the power cone, i.e. given a convex set $\mathcal{C}$, we are looking for representations of $\mathcal{C}$ in terms of $\mathcal{K}_\alpha$ in the following sense. We call $\mathcal{C}$ *power cone representable* (*α-representable*) if there exist a finite integer M, scalars $\alpha_i \in [0,1], i = 1, \ldots, M$, vectors $c_1, \ldots, c_M \in \mathbb{R}^3$, matrices $A_1, \ldots, A_M$ with 3 columns and appropriate number of rows, a matrix A_f and a vector c_f such that

$$u \in \mathcal{C} \qquad \Leftrightarrow \qquad c_i - A_i^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\alpha_i},\ i = 1, \ldots, M,\ A_f^T \begin{bmatrix} u \\ v \end{bmatrix} = c_f$$

for some vector v. The variables v are denoted *artificial* variables or *modelling* variables. We denote the above finite system of conic inequalities by S and call it the *α-representation* of $\mathcal{C}$. More compactly, we can write

$$u \in \mathcal{C} \qquad \Leftrightarrow \qquad c - A^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_M},\ A_f^T \begin{bmatrix} u \\ v \end{bmatrix} = c_f$$

where c is the vector containing the c_i's and $A = [A_1, \ldots, A_M]$.

Similarly we define α-representability of a convex function f via α-representability of its epigraph, i.e.

$$f \text{ is } \alpha\text{-representable} \qquad \Leftrightarrow \qquad \text{epi}(f) \text{ is } \alpha\text{-representable.}$$

The representability of a given set $\mathcal{C}$ in terms of $\mathcal{K}_\alpha$ can be considered as a *lifting* into a higher-dimensional space while confining the nonlinearities in $\mathcal{C}$ into smaller building blocks, the power cones $\mathcal{K}_{\alpha_i}$. Since we know a self-concordant barrier for $\mathcal{K}_\alpha$, we can then construct a self-concordant barrier of the lifted power cone reformulation of an α-representable set $\mathcal{C}$. As a consequence we can optimize linear functions over such sets $\mathcal{C}$.

In the following two sections we explore sets that are α-representable, first for the power cone itself, then for a limit of the power cone. In Section 4.3 we compute the duals of these cones, while Section 4.4 is devoted to two concrete problem classes making use of earlier reformulations. Finally, we present numerical results of a dual and a primal-dual path-following method for solving three different problem classes and compare these methods with several commercial nonlinear programming solvers in Section 4.5.

4.1 Power cone representability

We build up the class of α-representable functions in a similar fashion as proposed by Ben-Tal and Nemirovski in [3] for sets and functions that are representable in terms of second-order cones and semidefinite cones, namely we present

1. elementary α-representable functions and sets,

2. operations that preserve α-representability.

4.1.1 Elementary α-representable functions

1. **affine function:** $f(x) = a^T x + b$ is α-representable since $a^T x + b \leq t$ if and only if $-a^T x + t - b \geq 0 = |0|$, which is equivalent to $(-a^T x + t - b)^1 1^0 \geq |0|$ or
$$(-a^T x + t - b, 1, 0) \in \mathcal{K}_1.$$

2. **convex p-power:** $f(x) = |x|^p$, $1 \leq p \leq \infty$. Its epigraph is given by $|x|^p \leq t$ which is the same as $(t, 1, x) \in \mathcal{K}_\alpha$ with $\alpha = 1/p$.

3. **concave power:** the set $\{(x,t) : x^\alpha \geq |t|\}$, with $\alpha \in (0, 1]$, $x \geq 0$ can be described by the inequality $x^\alpha 1^{1-\alpha} \geq |t|$, or equivalently $(x, 1, t) \in \mathcal{K}_\alpha$.

4. **inverse of p-power:** $f(x) = x^p$, $p < 0$, $x > 0$. Indeed, we see that the epigraph of f is given by $x^p \leq t$, which is clearly the same as to $1 \leq tx^{-p}$. If we define $\alpha = 1/(1-p) \in (0, 1)$ and take both sides of the inequality to the power of α, then the inequality sign remains the same (since τ^α is monotonically increasing in τ for $\alpha \in (0, 1)$). We get
$$1 = 1^\alpha \leq t^\alpha \cdot x^{-p \cdot \alpha}.$$
It remains to note that $-p\alpha = -(\alpha - 1)/\alpha \cdot \alpha = 1 - \alpha$. That means the epigraph of f can be described by $(t, x, 1) \in \mathcal{K}_\alpha$, where $\alpha = 1/(1-p)$.

We also have the following observation. If f is α-representable then any sublevel set
$$L_c(f) = \{x : f(x) \leq c\}$$
is α-representable too.

4.1.2 Important examples of α-representable sets

High-dimensional power cone $\mathcal{K}_\alpha^{(n)}$

We recall here the definition of the higher-dimensional power cone. For α such that $\alpha_i \geq 0$ and $\sum_{i=1}^n \alpha_i = 1$ we define
$$\mathcal{K}_\alpha^{(n)} = \left\{(x, z) \in \mathbb{R}_+^n \times \mathbb{R} : x_1^{\alpha_1} \cdots x_n^{\alpha_n} \geq |z|\right\}$$

We decompose $\mathcal{K}_\alpha^{(n)}$ into several smaller power cones of dimension 3 each. Indeed, it holds that $(x, z) \in \mathcal{K}_\alpha^{(n)}$ if and only if $\exists v_1 \geq 0$ such that $x_1^{\alpha_1} v_1^{1-\alpha_1} \geq |z|$ and
$$x_2^{\alpha_2} \cdots x_n^{\alpha_n} \geq v_1^{1-\alpha_1}, \tag{4.1}$$
because then we have
$$x_1^{\alpha_1} \underbrace{x_2^{\alpha_2} \cdots x_n^{\alpha_n}}_{\geq v_1^{1-\alpha_1}} \geq x_1^{\alpha_1} v_1^{1-\alpha_1} \geq |z|.$$

Note that the inequality (4.1) can be alternatively written as
$$x_2^{\frac{\alpha_2}{1-\alpha_1}} \cdots x_n^{\frac{\alpha_n}{1-\alpha_1}} \geq |v_1| = v_1, \tag{4.2}$$

where the exponents satisfy

$$\sum_{i=2}^{n} \frac{\alpha_i}{1-\alpha_1} = \frac{\sum_{i=2}^{n} \alpha_2}{1-\alpha_1} = \frac{1-\alpha_1}{1-\alpha_1} = 1.$$

For $i = 2, \dots, n-1$ let us denote $\tilde{\alpha}_i = \frac{\alpha_i}{\alpha_i + \cdots + \alpha_n}$. For $i = 2$ we have then $\tilde{\alpha}_2 = \frac{\alpha_1}{1-\alpha_1}$. Then (4.2) is true if and only if $\exists\, v_2 \geq 0$ such that $x_2^{\tilde{\alpha}_2} v_2^{1-\tilde{\alpha}_2} \geq |v_1| = v_1$ and

$$x_3^{\frac{\alpha_3}{1-\alpha_1}} \cdots x_n^{\frac{\alpha_n}{1-\alpha_1}} \geq v_2^{1-\tilde{\alpha}_2} = v_2^{\frac{\alpha_3+\cdots+\alpha_n}{1-\alpha_1}},$$

or in other words

$$x_3^{\alpha_3} \cdots x_n^{\alpha_n} \geq v_2^{\alpha_3+\cdots+\alpha_n}.$$

We can proceed now in the same manner as above until we get in the last step

$$x_{n-1}^{\tilde{\alpha}_{n-1}} x_n^{1-\tilde{\alpha}_{n-1}} \geq |v_{n-2}| = v_{n-2}.$$

That means we can decompose $\mathcal{K}_\alpha^{(n)}$ in the following way: $(x, z) \in \mathcal{K}_\alpha^{(n)}$ if and only if $\exists\, v_1 \geq 0, \dots, v_{n-2} \geq 0$ such that

$$\begin{aligned} (x_1, v_1, z) &\in \mathcal{K}_{\alpha_1} \\ (x_i, v_i, v_{i-1}) &\in \mathcal{K}_{\tilde{\alpha}_i}, \quad i = 2, \dots, n-2, \\ (x_{n-1}, x_n, v_{n-2}) &\in \mathcal{K}_{\tilde{\alpha}_{n-1}}, \end{aligned} \tag{4.3}$$

where $\tilde{\alpha}_i = \frac{\alpha_i}{\alpha_i + \cdots + \alpha_n}$ for $i = 2, \dots, n-1$.

In other words, $\mathcal{K}_\alpha^{(n)}$ can be decomposed into $n-1$ low-dimensional power cones $\mathcal{K}_\alpha$ by introducing $n-2$ additional variables $v_1, \dots, v_{n-2}$. The self-concordance parameter of the α-representation (4.3) is $\nu = 3(n-1)$.

p-cone

We recall the definition of the p-cone in dimension $n+1$. For $p \geq 1$ we define

$$\mathcal{P}_p^{(n)} = \{(x, t) : ||x||_p \leq t\} \subset \mathbb{R}^{n+1},$$

where $||x||_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$ denotes the p-norm of a point $x \in \mathbb{R}^n$. Note that p may assume its limit value ∞, in which case

$$\mathcal{P}_\infty^{(n)} = \left\{(x, t) : \max_{i=1,\dots,n} |x_i| \leq t\right\}.$$

We now prove that $\mathcal{P}_p^{(n)}$ is α-representable, i.e. $(x, t) \in \mathcal{P}_p^{(n)}$ if and only if $\exists\, y_i \geq 0$ such that

$$(y_i, t, x_i) \in \mathcal{K}_\alpha,\ i = 1, \dots, n \tag{4.4}$$

$$\sum_{i=1}^{n} y_i = t, \tag{4.5}$$

where $\alpha = \frac{1}{p}$ for $1 \leq p < \infty$. For $p \to \infty$ we get asymptotically the infinity norm $||x||_\infty = \max_{i=1,\dots,n} |x_i|$. By convention we define $\alpha = 0$ for $p = \infty$. The parameter of the α-representation above is $\nu = 3n$.

Let us first consider the case $\alpha = 0$. Then (4.4) means $y_i^0 t^1 = t \geq |x_i|$ for all $i = 1, \dots, n$. This is equivalent to $t \geq \max_{i=1,\dots,n} |x_i| = ||x||_\infty$. Reversely, if $||x||_\infty \leq t$, we define $y_i = \frac{t}{n}, i = 1, \dots, n$ and we get directly (4.4) and (4.5).

Let $\alpha > 0$, assume that (4.4) and (4.5) hold. Then

$$\begin{aligned} & (y_i, t, x_i) \in \mathcal{K}_\alpha, \ \forall i \\ \Leftrightarrow \ & y_i^\alpha \cdot t^{1-\alpha} \geq |x_i|, \ \forall i \\ \Leftrightarrow \ & y_i \cdot t^{(1-\alpha)/\alpha} \geq |x_i|^{1/\alpha}, \ \forall i \\ \Rightarrow \ & \underbrace{\sum_{i=1}^n y_i}_{=t} \cdot t^{(1-\alpha)/\alpha} \geq \sum_{i=1}^n |x_i|^{1/\alpha}, \\ \Leftrightarrow \ & t^p \geq \sum_{i=1}^n |x_i|^p, \end{aligned}$$

since $\frac{1-\alpha}{\alpha} = p - 1$. The last inequality means in fact $||x||_p \leq t$, or $(x, t) \in \mathcal{P}_p^{(n)}$.

On the other hand, let $(x, t) \in \mathcal{P}_p^{(n)}$. We define

$$\begin{aligned} \epsilon &:= t^p - \sum_{i=1}^n |x_i|^p \geq 0 \\ y_i &:= \frac{1}{t^{p-1}} \cdot \left(\frac{\epsilon}{n} + |x_i|^p \right). \end{aligned}$$

Then

$$\begin{aligned} \sum_{i=1}^n y_i &= \sum_{i=1}^n \frac{1}{t^{p-1}} \cdot \left(\frac{\epsilon}{n} + |x_i|^p \right) = \frac{\epsilon}{t^{p-1}} + \sum_{i=1}^n \frac{|x_i|^p}{t^{p-1}} \\ &= t - \sum_{i=1}^n \frac{|x_i|^p}{t^{p-1}} + \sum_{i=1}^n \frac{|x_i|^p}{t^{p-1}} = t. \end{aligned}$$

Additionally, we have

$$\begin{aligned} & (y_i, t, x_i) \in \mathcal{K}_\alpha \\ \Leftrightarrow \ & y_i^\alpha \cdot t^{1-\alpha} \geq |x_i|, \ \forall i \\ \Leftrightarrow \ & y_i \cdot t^{(1-\alpha)/\alpha} \geq |x_i|^{1/\alpha}, \ \forall i. \end{aligned}$$

Using the above definition of y_i, we get

$$y_i \underbrace{t^{(1-\alpha)/\alpha}}_{=t^{p-1}} = \frac{1}{t^{p-1}} \left(\frac{\epsilon}{n} + |x_i|^p \right) t^{p-1} = \underbrace{\frac{\epsilon}{n}}_{\geq 0} + |x_i|^p \geq |x_i|^{1/\alpha},$$

which guarantees that $(y_i, t, x_i) \in \mathcal{K}_\alpha$. We conclude that $\mathcal{P}_p^{(n)}$ is α-representable.

The α-representability of $\mathcal{P}_p^{(n)}$ has the following implication: for $p = 2$ we obtain the second-order cone $\mathbb{L}^n = \{(x,t) : ||x||_2 \leq t\} \subset \mathbb{R}^{n+1}$ which becomes also α-representable. That means all sets (and applications derived from them) that are representable in terms of the second-order cone fall into the scope of $\mathcal{K}_\alpha$. Such sets are listed in [3, Chapter 3]. Note, however, that modelling the second order cone $\mathbb{L}^n$ with $\mathcal{K}_\alpha$ is not efficient because its parameter ($\nu = 3n$) is far higher than the optimal parameter for the second order cone ($\nu = 2$, independent of the dimension n).

It follows directly that p-norms are α-representable too, since the epigraph of $f(x) = ||x||_p$ is exactly the p-cone $\mathcal{P}_p^{(n)}$. Moreover, p-unit balls are α-representable since they are sublevel sets of $f(x) = ||x||_p$.

l_p-cone

Given $p \in \mathbb{R}^n_{++}$, the l_p-cone is defined as

$$\mathcal{L}^p := \left\{ (x,t,s) : \sum_{i=1}^n \frac{1}{p_i} \left(\frac{|x_i|}{t} \right)^{p_i} \leq \frac{s}{t} \right\}.$$

This cone has applications in particular in l_p-norm optimization. Glineur (see [23, Section 4.2]) has shown that $\mathcal{L}^p$ is a proper cone .

We claim that $(x,t,s) \in \mathcal{L}^p$ if and only if $\exists\ v \in \mathbb{R}^n_+$ such that

$$\begin{aligned} (v_i, t, x_i) &\in \mathcal{K}_{\alpha_i}, \\ \sum_{i=1}^n p_i v_i &= s, \end{aligned}$$

where $\alpha_i = 1/p_i, i = 1, \dots, n$. The parameter of the above α-representation of the l_p-cone is $\nu = 3n$.

Indeed, let $(v_i, t, x_i) \in \mathcal{K}_{1/p_i}$. This is equivalent to

$$\begin{aligned} && v_i^{\frac{1}{p_i}} t^{\frac{p_i-1}{p_i}} &\geq |x_i|, \\ \Leftrightarrow && v_i t^{p_i-1} &\geq |x_i|^{p_i}, \\ \Leftrightarrow && \frac{v_i}{t} &\geq \left(\frac{|x_i|}{t} \right)^{p_i}. \end{aligned}$$

By multiplying the above inequalities with p_i and take the sum over all i, we get

$$\sum_{i=1}^n p_i \frac{|x_i|^{p_i}}{t} \leq \sum_{i=1}^n p_i \frac{v_i}{t} = \frac{s}{t}.$$

Conversely, let $(x,t,s) \in \mathcal{L}^p$, define

$$\varepsilon_i = \frac{s - \sum_{j=1}^n p_j \frac{|x_j|^{p_j}}{t^{p_j-1}}}{p_i} \geq 0$$

and set

$$v_i := \frac{|x_i|^{p_i}}{t^{p_i-1}} + \frac{\varepsilon_i}{n}.$$

Then $v_i \geq \frac{|x_i|^{p_i}}{t^{p_i-1}}$, or equivalently $(v_i, t, x_i) \in \mathcal{K}_{1/p_i}$. On the other hand, when multiplying each term by p_i and taking the sum, we get

$$\begin{aligned}\sum_{i=1}^{n} p_i v_i = \sum_{i=1}^{n} \left[p_i \frac{|x_i|^{p_i}}{t^{p_i-1}} + \frac{p_i \varepsilon_i}{n} \right] &= \sum_{i=1}^{n} \left[p_i \frac{|x_i|^{p_i}}{t^{p_i-1}} \right] + \sum_{i=1}^{n} \frac{1}{n} \left[s - \sum_{i=1}^{n} p_i \frac{|x_i|^{p_i}}{t^{p_i-1}} \right] \\ &= s.\end{aligned}$$

We conclude that $\mathcal{L}^p$ is α-representable.

Hypograph of geometric mean

The geometric mean of two nonnegative variables x and y is given by

$$f(x, y) = \sqrt{xy}$$

which is a concave function. Its hypograph is given by

$$\mathcal{C}_{GM} = \{(x, y, t) : x \geq 0, y \geq 0, t \leq \sqrt{xy}\}.$$

We claim that $\mathcal{C}_{GM}$ is α-representable. Indeed, we have that $(x, y, t) \in \mathcal{C}_{GM}$ if and only if $\exists$ v such that

$$\begin{aligned}(x, y, v) &\in \mathcal{K}_{1/2}, \\ t &\leq v.\end{aligned}$$

This is true because

$$x^{\frac{1}{2}} y^{\frac{1}{2}} \geq |v| \geq v \geq t.$$

The parameter of the α-representation of the geometric mean is $\nu = 4$.

4.1.3 Operations that preserve α-representability

The convex sets given by the epigraph and the sublevel sets of the α-representable functions listed above can be transformed into other sets that are α-representable too. The list below of operations that preserve α-representability is included in the one presented by Ben-Tal and Nemirovski in [3, Section 3.3]. The difference is that the authors of [3] consider as basic elements functions and sets that are representable in terms of the *second-order cone.* The authors establish representability of many kinds of sets and functions that are quadratic in their nature.

Here the basic atoms are elements that are not restricted to degree two. In fact, any kind of convex constraint with real powers (not necessarily rational) is expressible. Therefore, the scope of α-representable functions and sets is a little wider, as we will see now.

Intersection of α-representable sets

Let $\mathcal{C}_i \subset \mathcal{E}, i = 1, \dots, N$ be α-representable. Then so is their intersection $\mathcal{C} := \bigcap_{i=1}^N \mathcal{C}_i$. Indeed, let S_i, $i = 1, \dots, N$ be the α-representation of $\mathcal{C}_i$ with artificial variables v_i. Then we obtain the α-representation of $\mathcal{C}$ as

$$\{(u, v_i) \text{ that satisfy } S_i\}, i = 1, \dots, N.$$

Example:

Let $f_1(x) = |x|^{p_1}$, $p_1 \geq 1$ and $f_2(x) = x^{p_2}$, where $x > 0$ and $p_2 < 0$. Then the intersection of the epigraphs

$$\text{epi}(f_1) \bigcap \text{epi}(f_2)$$

is α-representable because f_1 and f_2 are in the list of elementary α-representable functions.

Direct product of α-representable sets

Let $\mathcal{C}_i \subset \mathcal{E}_i, i = 1, \dots, N$ be α-representable. Then so is their direct product $\mathcal{C} := \mathcal{C}_1 \times \cdots \times \mathcal{C}_N$. It is clear that if S_i is an α-representation of $\mathcal{C}_i$ in the variables $u_i \in \mathcal{C}_i$ and v_i, then the union of S_i is an α-representation of $\mathcal{C}$.
Example: mix of $\mathcal{K}_\alpha^{(n)}$ and $\mathcal{P}_p^{(n)}$.

We consider a generalization of $\mathcal{K}_\alpha^{(n)}$, where the right-hand side term $|z|$ is replaced by any p-norm of a vector $||z||_p$, i.e. for $\alpha_i \geq 0$, $\sum_{i=1}^{n_1} \alpha_i = 1$ and $p \geq 1$ we define

$$\mathcal{K}_{\alpha,p}^{(n_1,n_2)} = \left\{ (x, z) \in \mathbb{R}_+^{n_1} \times \mathbb{R}^{n_2} : \prod_{i=1}^{n_1} x_i^{\alpha_i} \geq ||z||_p \right\}.$$

We see that $(x, z) \in \mathcal{K}_{\alpha,p}^{(n_1,n_2)}$ if and only if $\exists\, t \geq 0$ such that

$$\begin{aligned} (x, t) &\in \mathcal{K}_\alpha^{(n_1)}, \\ (z, t) &\in \mathcal{P}_p^{(n_2)}. \end{aligned}$$

As we have seen before, the first expression can be modelled as in (4.3) and the second as in (4.4) and (4.5). The artificial variable t that is shared between both cones can be replaced by using the identity (4.5). We obtain then the following α-representation of $\mathcal{K}_{\alpha,p}^{(n_1,n_2)}$: a point $(x, z) \in \mathcal{K}_{\alpha,p}^{(n_1,n_2)}$ if and only if $\exists\, v_i \geq 0, i = 1, \dots, n_1 - 2$ and $w_i \geq 0$, $i = 1, \dots, n_2$ such that

$$\begin{aligned} (x_1, v_1, \sum_{i=1}^{n_2} w_i) &\in \mathcal{K}_{\alpha_1} \\ (x_i, v_i, v_{i-1}) &\in \mathcal{K}_{\tilde{\alpha}_i}, \;\; i = 2, \dots, n_1 - 2, \\ (x_{n_1-1}, x_{n_1}, v_{n_1-2}) &\in \mathcal{K}_{\tilde{\alpha}_{n_1-1}}, \\ \left(w_i, \sum_{i=1}^{n_2} w_i, z_i \right) &\in \mathcal{K}_{1/p}, \; i = 1, \dots, n_2, \end{aligned}$$

where $\tilde{\alpha}_i = \frac{\alpha_i}{\alpha_i + \cdots + \alpha_n}$ for $i = 1, \ldots, n_1 - 1$. We have introduced $n_1 + n_2 - 2$ new variables $v_1, \ldots, v_{n_1-2}$, and $w_1, \ldots, w_{n_2}$, as well as the $n_1 + n_2 - 1$ power cone constraints.

Affine image of α-representable sets

Let $\mathcal{C} \subseteq \mathbb{R}^n$ be α-representable and $f : \mathbb{R}^n \to \mathbb{R}^m$ such that $f(x) = Ax + b$, where $A \in \mathbb{R}^{m,n}$ and $b \in \mathbb{R}^m$. Then

$$f(\mathcal{C}) := \{f(x), x \in \mathcal{C}\} \subseteq \mathbb{R}^m$$

is α-representable. The proof can be found in [3, p. 89], where the authors consider the representability in terms of second-order cones. the arguments for our setting are exactly the same.

Inverse affine image of α-representable sets

Let $\mathcal{C} \subseteq \mathbb{R}^n$ be α-representable and $f : \mathbb{R}^k \to \mathbb{R}^n$ such that $f(z) = Bz + b$, where $B \in \mathbb{R}^{n,k}$ and $b \in \mathbb{R}^n$, then

$$f^{-1}(\mathcal{C}) := \{z : f(z) \in \mathcal{C}\} \subseteq \mathbb{R}^k$$

is α-representable. Indeed, let $u = Bz + b$ and S the α-representation of $\mathcal{C}$ with the vector c and matrix A. Then

$$c - A^T \begin{bmatrix} u \\ v \end{bmatrix} = c - A^T \begin{bmatrix} Bz + b \\ v \end{bmatrix} \in \mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_M}$$

is clearly an α-representation of $f^{-1}(\mathcal{C})$ (because the term in the middle is an affine expression in the variables z and v).

Maximum of α-representable functions

Let $F_i : \mathcal{E} \to \mathbb{R},\ i = 1, \ldots, N$ be α-representable functions. Then

$$F(x) = \max_{i=1,\ldots,N} F_i(x)$$

is α-representable.

Indeed, we have $\text{epi}(F) = \bigcap_{i=1}^N \text{epi}(F_i)$, where $\text{epi}(F_i)$ are α-representable sets. We saw above that the intersection of finitely many α-representable sets is again α-representable.

Nonnegative weighted sum of α-representable functions

Let $F_i : \mathcal{C}_i \subseteq \mathcal{E} \to \mathbb{R}, i = 1, \ldots, m$ be α-representable functions and $\lambda_i \geq 0, i = 1, \ldots, m$. Then

$$F(x) = \sum_{i=1}^m \lambda_i F_i(x)$$

is α-representable on $\bigcap_{i=1}^n \mathcal{C}_i$.

Indeed, the epigraph of F which is given by $F(x) = \sum_{i=1}^m \lambda_i F_i(x) \le t$, can be equivalently written as

$$\begin{aligned} F_i(x) \le v_i,\ i = 1, \dots, m, \\ \sum_{i=1}^m \lambda_i v_i \le t. \end{aligned}$$

This is a direct product of m α-representable sets and the epigraph of a linear function in v, which is α-representable too. The last inequality can in fact be replaced by the linear equation $\sum_{i=1}^m \lambda_i v_i = t$.

Nonnegative weighted sum of separable α-representable functions

Let $F_i : \mathcal{C}_i \subseteq \mathcal{E}_i \to \mathbb{R}, i = 1, \dots, m$ be α-representable functions and $\lambda_i \ge 0, i = 1, \dots, m$. Then

$$F(x) = \sum_{i=1}^m \lambda_i F_i(x_i)$$

is α-representable on $\mathcal{C}_1 \times \ldots \times \mathcal{C}_m$.

We see that the epigraphs of $\tilde{F}_i(x) = F_i(x_i)$ are α-representable (because their epigraphs are inverse images of α-representable sets under affine transformations). Moreover, $F(x) = \sum_{i=1}^m F_i(x_i) = \sum_{i=1}^m \tilde{F}_i(x)$ which is α-representable because of the previous point.

Affine transformation in the arguments

Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$ be α-representable, $B \in \mathbb{R}^{n,p}$ and $c \in \mathbb{R}^n$. Then

$$\tilde{F}(y) = F(By + c)$$

is α-representable on $\operatorname{dom} \tilde{F} = \{y \in \mathbb{R}^p : By + c \in \mathcal{C}\} \subseteq \mathbb{R}^p$.

Indeed, the epigraph of $\tilde{F}$ is the inverse image of the epigraph of F under an affine transformation.

Partial minimization

Let $F : \mathcal{C} \subseteq \mathbb{R}^n \to \mathbb{R}$, such that $(x, y) \mapsto F(x, y)$, be α-representable and bounded from below. Then the partial minimization of F with respect to y

$$G(x) = \inf_{y \in \mathcal{Q}(x)} F(x, y),$$

where $\mathcal{Q}(x) = \{y : (x, y) \in \mathcal{C}\}$, is α-representable on $\operatorname{dom} G = \{x : \mathcal{Q}(x) \neq \emptyset\}$.

Indeed, the epigraph of G, i.e. the set $\{(x, t) : G(x) \le t\}$, is simply the projection of the epigraph of F ($\{(x, y, t) : F(x, y) \le t\}$) onto the (x, t)-plane. It remains to note that this projection is a linear transformation.

4.1.4 Some more α-representable sets

Unhomogenizing the power cone

For $\alpha_i \geq 0, i = 1, 2$ with $\alpha_1 + \alpha_2 \leq 1$ we consider the convex set

$$\mathcal{C}_\alpha = \{(x_1, x_2, z) : x_1^{\alpha_1} \, x_2^{\alpha_2} \geq |z|\}.$$

We see that the definition of $\mathcal{C}_\alpha$ strongly resembles the one of the power cone $\mathcal{K}_\alpha$. The difference is that it is not required that the exponents sum to 1. It is clear that $\mathcal{C}_\alpha$ is α-representable because $(x, y, z) \in \mathcal{C}_\alpha$ if and only if $(x, y, 1, z) \in \mathcal{K}_\alpha^{(3)}$ with $\alpha = (\alpha_1, \alpha_2, 1 - \alpha_1 - \alpha_2)$. We saw that $\mathcal{K}_\alpha^{(3)}$ is α-representable and the argument is an affine transformation of the variables (x, y, z). Concretely, the α-representation becomes: $(x_1 x_2, z) \in \mathcal{C}_\alpha$ if and only if $\exists \; v \geq 0$ such that

$$\begin{aligned} (x_1, v, z) &\in \mathcal{K}_{\alpha_1}, \\ (x_2, 1, v) &\in \mathcal{K}_{\frac{\alpha_2}{1-\alpha_1}}. \end{aligned}$$

Indeed,

$$x^{\alpha_1} \; \underbrace{y^{\alpha_2}}_{\geq v^{1-\alpha_1}} \; \geq x^{\alpha_1} \, v^{1-\alpha_1} \geq |z|.$$

Unhomogenizing the p-cone

Let us consider the convex set

$$\mathcal{C}_p = \left\{ (x, t) : \sum_{i=1}^{n} |x_i|^{p_i} \leq t^p, \; t \geq 0 \right\}$$

where $1 \leq p \leq \min_{i=1,\ldots,n} p_i$. It can be easily seen that $(x, t) \in \mathcal{C}_p$ if and only if there exists $v \geq 0$ such that

$$|x_i|^{p_i} \leq v_i^p, \; i = 1, \ldots, n \tag{4.6}$$

$$\sum_{i=1}^{n} v_i^p \leq t^p, \tag{4.7}$$

because, for $(x, t) \in \mathcal{C}_p$ choose $v_i = |x_i|^{p_i/p}$, then

$$\begin{aligned} |x_i|^{p_i} &= (|x_i|^{\frac{p_i}{p}})^p = v_i^p \; (\leq v_i^p) \\ \sum_{i=1}^{n} v_i^p &= \sum_{i=1}^{n} |x_i|^{p_i} \leq t^p. \end{aligned}$$

The reverse implication is immediate.

Homogenizing the inequalities in (4.6) with $\theta_i = 1$, we get

$$\begin{aligned} \left|\frac{x_i}{\theta_i}\right|^{p_i} &\leq \left(\frac{v_i}{\theta_i}\right)^p \\ \Leftrightarrow \quad |x_i|^{p_i} &\leq v_i^p \cdot \theta_i^{p_i - p} \\ \Leftrightarrow \quad |x_i| &\leq v_i^{p/p_i} \cdot \theta_i^{(p_i-p)/p_i} \\ \Leftrightarrow \quad (v_i, 1, x_i) &\in \mathcal{K}_{\alpha_i}, \ \alpha_i = \frac{p}{p_i}, \ i = 1, \dots, n. \end{aligned}$$

The p-cone constraint (4.7) is modelled again by introducing additional variables w_i, with $\sum_{i=1}^n w_i = t$. As above, t is replaced in the formulation. Finally, we get $(x, t) \in \mathcal{C}_p$ if and only if there are $v \geq 0$ and $w \geq 0$ such that

$$\begin{aligned} (v_i, 1, x_i) &\in \mathcal{K}_{\alpha_i}, \ \alpha_i = \frac{p}{p_i}, \ i = 1, \dots, n, \\ \left(w_i, \sum_{i=1}^n w_i, v_i\right) &\in \mathcal{K}_{\alpha_{n+1}}, \ \alpha_{n+1} = \frac{1}{p}, \ i = 1, \dots, n. \end{aligned}$$

4.2 The exponential cone

In this section we consider the cone that is obtained by taking the conic hull of the epigraph of the exponential function.

$$\mathcal{K}_{\exp} = \operatorname{cl}\left(\left\{z_1 \in \mathbb{R}, z_2 \in \mathbb{R}_+, z_3 \in \mathbb{R}_{++} : \exp\left(\frac{z_1}{z_3}\right) \leq \frac{z_2}{z_3}\right\}\right).$$

Since we work with closed convex cones, we have to take the closure in the above definition. Let us denote

$$\mathcal{K}^0_{\exp} = \left\{z_1 \in \mathbb{R}, z_2 \in \mathbb{R}_+, z_3 \in \mathbb{R}_{++} : \exp\left(\frac{z_1}{z_3}\right) \leq \frac{z_2}{z_3}\right\}.$$

We have then the following description of $\mathcal{K}_{\exp}$.

Lemma 4.2.1. *It holds*

$$\mathcal{K}_{\exp} = \mathcal{K}^0_{\exp} \bigcup (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}.$$

Proof. Let us denote

$$\bar{\mathcal{K}}_{\exp} = \mathcal{K}^0_{\exp} \bigcup (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}.$$

As a first step we show that $\bar{\mathcal{K}}_{\exp} \subseteq \mathcal{K}_{\exp}$. Let $z \in \bar{\mathcal{K}}_{\exp}$. That means either $z \in \mathcal{K}^0_{\exp}$ or $z \in (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}$. If $z \in \mathcal{K}^0_{\exp}$ then it automatically holds that $z \in \mathcal{K}_{\exp}$ since $\mathcal{K}^0_{\exp} \subseteq \operatorname{cl}(\mathcal{K}^0_{\exp}) = \mathcal{K}_{\exp}$. Let now $z = (z_1, z_2, z_3) \in (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}$. In order to prove that $z \in \mathcal{K}_{\exp}$ we need to show that z is an accumulation point

of a sequence $\{z^{(k)}\} \subset \mathcal{K}^0_{\text{exp}}$.
For $k = 1, 2, \ldots$ we define the sequence $z^{(k)} = (z_1^{(k)}, z_2^{(k)}, z_3^{(k)})$ with

$$\begin{aligned} z_1^{(k)} &= -\frac{1}{k} + z_1, \\ z_2^{(k)} &= \frac{1}{k} + z_2 \geq 0, \ \forall \text{ integer } k \geq 1, \\ z_3^{(k)} &= \frac{1}{k^2} > 0, \ \forall \text{ integer } k \geq 1. \end{aligned}$$

We obtain the following chain of inequalities:

$$\begin{aligned} \exp\left(\frac{z_1^{(k)}}{z_3^{(k)}}\right) = \exp\left(\frac{-\frac{1}{k} + z_1}{\frac{1}{k^2}}\right) &= \exp\left(-k \underbrace{+k^2 z_1}_{\leq 0}\right) \\ &\leq \exp(-k) \leq \exp(-1) < 1 \\ &\leq k \leq k + k^2 z_2 = \frac{z_2^{(k)}}{z_3^{(k)}}. \end{aligned}$$

That means the sequence $\{z^{(k)}\} \subset \mathcal{K}^0_{\text{exp}}$ for all $k = 1, 2, \ldots$ and we have $z^{(k)} \to (z_1, z_2, 0)$ as $k \to \infty$.

Conversely, let us show that $\mathcal{K}_{\text{exp}} \subseteq \bar{\mathcal{K}}_{\text{exp}}$. Let $z \in \mathcal{K}_{\text{exp}}$ and assume $z \notin \bar{\mathcal{K}}_{\text{exp}}$, i.e. $z \notin \mathcal{K}^0_{\text{exp}}$ and $z \notin (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}$. One can verify that the only possible situation where this can happen is $z_1 > 0$, $z_2 \geq 0$ and $z_3 = 0$. The other possibilities like $z_2 < 0$ or $z_3 < 0$ or $\exp\left(\frac{z_1}{z_3}\right) > \frac{z_2}{z_3}$ are excluded immediately by looking at $\mathcal{K}^0_{\text{exp}}$. Let $z = (z_1, z_2, z_3)$ be such a potential accumulation point such that $z_1 > 0$, $z_2 \geq 0$ and $z_3 = 0$. We see that any sequence $\{z^{(k)}\} \subset \mathcal{K}^0_{\text{exp}}$ must satisfy

$$\exp\left(\frac{z_1^{(k)}}{z_3^{(k)}}\right) \leq \frac{z_2^{(k)}}{z_3^{(k)}}, \ z_2^{(k)} \geq 0, z_3^{(k)} > 0,$$

or equivalently

$$\frac{\exp\left(-\frac{z_1^{(k)}}{z_3^{(k)}}\right)}{z_3^{(k)}} \geq \frac{1}{z_2^{(k)}}, \ z_2^{(k)} \geq 0, z_3^{(k)} > 0. \tag{4.8}$$

Since $z_1^{(k)} \to z_1 > 0$ we can conclude that $z_1^{(k)}$ must be strictly positive for all $k \geq \bar{k}$, for some $\bar{k}$. On the other hand, for any positive $z_1^{(k)}$ the term on the left-hand side of (4.8) tends to 0 for $k \to \infty$, while the term on the right-hand side of (4.8) tends to a positive constant (if $z_2 > 0$) or to ∞ (if $z_2 = 0$). In both cases it holds

$$\frac{1}{z_2^{(k)}} \geq c > 0$$

for some k. This contradicts (4.8) and we conclude that $z = (z_1, z_2, z_3)$ with $z_1 > 0$, $z_2 \geq 0$ and $z_3 = 0$ cannot be an accumulation point of $\mathcal{K}^0_{\text{exp}}$. It follows that $\mathcal{K}_{\text{exp}} \subseteq \bar{\mathcal{K}}_{\text{exp}}$, which finishes the proof.

□

It is interesting to see that $\mathcal{K}_{\exp}$ can be considered as a limit of a linear transformation of the power cone $\mathcal{K}_\alpha$ for $\alpha \to 0$. Consider

$$\tilde{\mathcal{K}_\alpha} = \{(z_1, z_2, z_3) : z_2^\alpha z_3^{1-\alpha} \geq |z_3 + \alpha z_1|\}.$$

Then $z \in \tilde{\mathcal{K}_\alpha}$ if and only if

$$\frac{z_2}{z_3} \geq \left|\frac{z_3 + \alpha z_1}{z_3}\right|^{1/\alpha} = \left|1 + \alpha\,\frac{z_1}{z_3}\right|^{1/\alpha}.$$

The term on the right-hand side converges to $\exp(z_1/z_3)$ as $\alpha \to 0$. That means

$$\lim_{\alpha \to 0} \tilde{\mathcal{K}_\alpha} = \mathcal{K}_{\exp}$$

in the sense that the indicator functions $I_\alpha(z)$ for $\tilde{\mathcal{K}_\alpha}$ converge pointwise to the indicator function $I_{\exp}(z)$ of $\mathcal{K}_{\exp}$.

In that sense we may extend the definition of α-representable sets $\mathcal{C}$ in the following way: There should exist finite integers M_1 and M_2, matrices A_α, $A_{\exp}$, A_f and vectors c_α, $c_{\exp}$ and c_f in appropriate sizes such that

$$u \in \mathcal{C} \Leftrightarrow \begin{cases} c_\alpha - A_\alpha^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_{M_1}} \\ c_{\exp} - A_{\exp}^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\exp} \times \cdots \times \mathcal{K}_{\exp} \;\; (M_2 \text{ times}), \\ A_f^T \begin{bmatrix} u \\ v \end{bmatrix} = c_f. \end{cases} \tag{α-REP}$$

for some artificial modelling variables v. This representation is essentially the same as the one discussed in Appendix A, where we present our Matlab implementation of a dual path-following interior-point solver for α-representable problems. The only difference in the input format of the solver is the additional distinction of linear constraints. We mentioned before that linear inequalities are α-representable. However, it is more efficient to solve them directly using the log-barrier for the nonnegative orthant.

The operations that preserve α-representability are the same as in Section 4.1.3. However, we can add now some more elementary sets and functions to our list α-representable objects.

4.2.1 Additional elementary α-representable functions

1. **exponential function:** $f : \mathbb{R} \to \mathbb{R}$, $f(x) = \exp(x)$ is α-representable. Indeed, $(x, t) \in \operatorname{epi}(f)$ if and only if $(x, t, 1) \in \mathcal{K}_{\exp}$.

2. **logarithm:** $f : \mathbb{R}_{++} \to \mathbb{R}$, $f(x) = -\log(x)$ is α-representable. Indeed, $(x, t) \in \operatorname{epi}(f)$ if and only if $-\log(x) \leq t$, which is equivalent to

$$\begin{aligned} && \log(x) &\geq -t, \\ \Leftrightarrow && x &\geq \exp(-t), \end{aligned}$$

which we model as $(-t, x, 1) \in \mathcal{K}_{\exp}$.

3. **entropy:** $f : \mathbb{R}_{++} \to \mathbb{R}$, $f(x) = x\log(x)$ is α-representable. Indeed, $(x,t) \in \mathrm{epi}(f)$ if and only if $x\log(x) \le t$. But this is equivalent to

$$\begin{aligned} & -x\log(1/x) \le t, \\ \Leftrightarrow \quad & \log(1/x) \ge \frac{-t}{x} \\ \Leftrightarrow \quad & \frac{1}{x} \ge \exp\left(\frac{-t}{x}\right), \end{aligned}$$

or in other words $(-t, 1, x) \in \mathcal{K}_{\mathrm{exp}}$.

4. The function $f : \mathbb{R}_{++} \to \mathbb{R}$, $f(x) = x\exp(1/x)$ is α-representable. Indeed, $(x,t) \in \mathrm{epi}(f)$ if and only if $x\exp(1/x) \le t$, which means $\exp(1/x) \le \frac{t}{x}$, or in other words $(1, t, x) \in \mathcal{K}_{\mathrm{exp}}$.

4.2.2 Some more examples of α-representable sets using $\mathcal{K}_{\mathrm{exp}}$

$\mathcal{K}_{\mathrm{exp}}$ with concave monomial term

Let us consider the generalization of $\mathcal{K}_{\mathrm{exp}}$. For $\alpha_i \ge 0, \sum_{i=1}^n \alpha_i = 1$ we define the cone

$$\mathcal{K}_{\mathrm{exp},\alpha} = \left\{ (x,y,z) \in \mathbb{R} \times \mathbb{R}^n_{++} \times \mathbb{R}_{++} : \exp\left(\frac{x}{z}\right) \le \frac{\prod_{i=1}^n y_i^{\alpha_i}}{z} \right\}.$$

We see that $\mathcal{K}_{\mathrm{exp},\alpha}$ is α-representable. Indeed, $(x,y,z) \in \mathcal{K}_{\mathrm{exp},\alpha}$ if and only if

$$\begin{aligned} (x,v,z) &\in \mathcal{K}_{\mathrm{exp}}, \\ (y,v) &\in \mathcal{K}_{\alpha}^{(n)}, \end{aligned}$$

with parameter $\nu = 3n$ (if the high-dimensional power cone $\mathcal{K}_{\alpha}^{(n)}$ is modelled using $n-1$ low-dimensional power cones $\mathcal{K}_{\alpha}$) or $\nu = n+4$ (if one uses the conjectured $(n+1)$-self-concordant barrier for $\mathcal{K}_{\alpha}^{(n)}$).

The geometric cone

In [23], Glineur presented a cone for geometric programming problems, namely

$$\mathcal{G}^n := \left\{ (x,t) \in \mathbb{R}^n_+ \times \mathbb{R}_+ : \sum_{i=1}^n \exp\left(-\frac{x_i}{t}\right) \le 1 \right\},$$

where for $t = 0$ we define $\exp\left(-\frac{x_i}{t}\right) = 0$. It turns out that $\mathcal{G}^n$ is a proper (i.e. convex, closed, solid, pointed) cone. For a detailed discussion, see [23, Section 5.2].

We see that $\mathcal{G}^n$ is α-representable. Indeed, $(x,t) \in \mathcal{G}^n$ if and only if $\exists\, v$ such that

$$\begin{aligned} (-x_i, v_i, t) &\in \mathcal{K}_{\mathrm{exp}} \\ \sum_{i=1}^n v_i &= t, \end{aligned}$$

with $\nu = 3n$.

Indeed, $(-x_i, v_i, t) \in \mathcal{K}_{\exp}$ means

$$\exp\left(-\frac{x_i}{t}\right) \leq \frac{v_i}{t},$$

and when taking the sum over all $i = 1, \dots, n$ we get exactly $\mathcal{G}^n$.

Reversely, let $(x, t) \in \mathcal{G}^n$, which means $\sum_{i=1}^n \exp\left(-\frac{x_i}{t}\right) \leq 1$, or equivalently $\sum_{i=1}^n t\exp\left(-\frac{x_i}{t}\right) \leq t$. Let us define $\varepsilon := t - \sum_{i=1}^n t\exp\left(-\frac{x_i}{t}\right) \geq 0$ and

$$v_i = t\exp\left(-\frac{x_i}{t}\right) + \underbrace{\frac{\varepsilon}{n}}_{\geq 0},\ i = 1, \dots, n.$$

Consequently we have

$$\sum_{i=1}^n v_i = \sum_{i=1}^n t\exp\left(-\frac{x_i}{t}\right) + \varepsilon = t,$$

and

$$v_i = t\exp\left(-\frac{x_i}{t}\right) + \frac{\varepsilon}{n} \geq t\exp\left(-\frac{x_i}{t}\right),$$

which means $\exp\left(-\frac{x_i}{t}\right) \leq \frac{v_i}{t}$, or equivalently $(-x_i, v_i, t) \in \mathcal{K}_{\exp}$.

That means $\mathcal{G}^n$ is α-representable.

Posynomial and generalized posynomial constraints

A monomial m_i is a function in N strictly positive variables x_k such that

$$m_i(x) = d_i \prod_{k=1}^N x_k^{a_{i,k}},$$

where $d_i > 0$ and $a_{i,k} \in \mathbb{R}$ for all k and i. A posynomial function f is a sum of monomials m_i,

$$f(x) = \sum_{i=1}^n m_i(x).$$

In general a posynomial is not convex (if it happens to be convex, then it can be modelled using $\mathcal{K}_\alpha$, see Section 4.1). However, under the above assumption that $x > 0$ and $d_i > 0$, we can apply the following change of variables

$$x_k = \exp(u_k) \qquad\qquad d_i = \exp(c_i).$$

for $u_k \in \mathbb{R}$ and $c_i \in \mathbb{R}$. That allows us to write

$$\begin{aligned} f(x) = \sum_{i=1}^{n} m_i(x) &= \sum_{i=1}^{n} d_i \prod_{k=1}^{N} x_k^{a_{i,k}} \\ &= \sum_{i=1}^{n} \exp(c_i) \prod_{k=1}^{N} \exp(u_k)^{a_{i,k}} \\ &= \sum_{i=1}^{n} \exp(c_i) \cdot \exp\left(\sum_{k=1}^{N} a_{i,k} u_k\right) \\ &= \sum_{i=1}^{n} \exp\left(a_i^T u + c_i\right), \end{aligned}$$

where we denote $a_i^T u = \sum_{k=1}^{N} a_{i,k} u_k$ for $i = 1, \ldots, n$. Note that even though f might not be convex in x, the last expression is indeed convex in the new variables u (as it is a sum of convex functions of affine functions in u). Posynomial constraints are typically of the form

$$f(x) \leq 1, \tag{4.9}$$

where f is a posynomial. Using the convex reformulation above, it is clear that constraints of the form

$$f(x) = \sum_{i=1}^{n} \exp\left(a_i^T u + c_i\right) \leq 1$$

can be expressed in terms of $\mathcal{K}_{\text{exp}}$. Indeed, the above inequality is valid if and only if

$$\begin{gathered} (a_i^T u + c_i, v_i, 1) \in \mathcal{K}_{\text{exp}} \\ \sum_{i=1}^{n} v_i = 1, \end{gathered}$$

where $v_i > 0$ are additional modelling variables. We conclude that posynomials of the form $\sum_{i=1}^{n} d_i \prod_{k=1}^{N} x_k^{a_{i,k}}$ are α-representable with parameter $\nu = 3n$.

Boyd et al. [5] have shown that generalizations of standard posynomial constraints (like fractional powers and maxima of posynomials) are expressible using entirely standard posynomial constraints. In order to do that, however, it is necessary to introduce some additional modelling variables. These generalized posynomial constraints lead to so-called generalized geometric programming.

On the other hand the conic setting allows another generalization of posynomial constraints of the form (4.9) which is different from the one proposed by Boyd et al., namely

$$f(x) \leq \log(m(x)), \tag{4.10}$$

where $m(x) = \tilde{d} \prod_{k=1}^{N} x_k^{\tilde{a}_k}$ is any monomial. We see that (4.10) is indeed more general than (4.9); by taking the constant monomial $m \equiv e$, we get exactly (4.9).

But using the same change of variables as above and $\tilde{d} = \exp(\tilde{c})$, we get

$$\begin{aligned}\sum_{i=1}^{n} \exp\left(a_i^T u + c_i\right) &\leq \log\left(\exp\left(\sum_{k=1}^{N} (\tilde{a}_k u_k + \tilde{c})\right)\right) \\ &= \tilde{a}^T u + \tilde{c},\end{aligned}$$

which can be expressed using $\mathcal{K}_{\exp}$ in the following manner

$$\begin{aligned}(a_i^T u + c_i, v_i, 1) &\in \mathcal{K}_{\exp} \\ \sum_{i=1}^{n} v_i &= \tilde{a}^T u + \tilde{c}.\end{aligned}$$

That means generalized posynomial constraints of the form (4.10) are α-representable.

The Lambert W function

The Lambert W function has various applications in different domains (see [12] for a discussion). It is introduced as the function that satisfies the defining equality

$$W(x) \cdot \exp\left(W(x)\right) = x.$$

For real arguments $x \geq 0$ the function W is injective and concave with $W(0) = 0$. $W(x)$ as a function of x is shown in Figure 4.1.

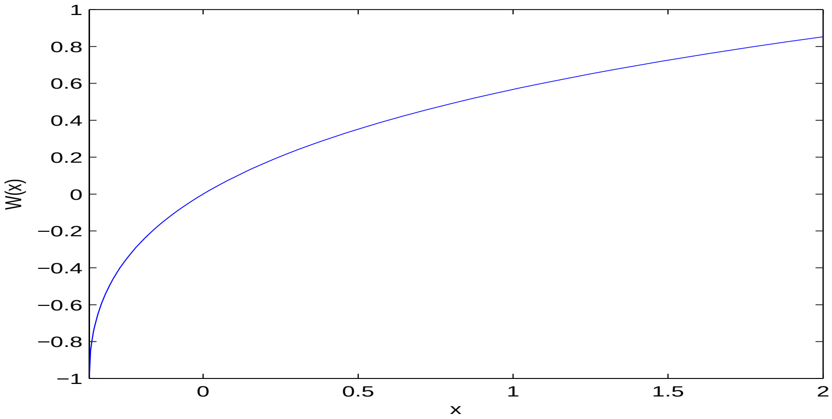


Figure 4.1: Lambert W function.

Note that it is not possible to write down the closed form analytic formula of W. However, we still can consider the convex set

$$W_L = \{(x, y) : x \geq 0, 0 \leq y \leq W(x)\}. \tag{4.11}$$

By applying the monotonic transformation $\tau \exp(\tau)$ to all three terms of $0 \leq y \leq W(x)$, we get

$$0 \leq y \cdot \exp(y) \leq W(x) \cdot \exp\left(W(x)\right) = x.$$

However, the term in the middle can be written as

$$y \cdot \exp(y) = y \cdot \exp\left(\frac{y^2}{y}\right) \leq y \cdot \exp\left(\frac{z}{y}\right),$$

which is true whenever $z \geq y^2$. That means $(x, y) \in W_L$ if and only if $y \exp\left(\frac{z}{y}\right) \leq x$ and $y^2 \leq z$, or in other words

$$\begin{aligned} (z, x, y) &\in \mathcal{K}_{\exp} \\ (z, 1, y) &\in \mathcal{K}_{\frac{1}{2}}. \end{aligned}$$

It follows, W_L is α-representable with parameter $\nu = 6$.

Another mixed set

We see that the cone

$$C = \left\{(x, y, z) : \exp\left(\frac{x^2}{2y^2}\right) \leq \frac{z}{y}\right\}$$

can be decomposed into

$$\begin{aligned} \frac{x^2}{2y^2} &\leq \frac{v}{y}, (\Leftrightarrow x^2 \leq 2vy) \\ \exp\left(\frac{v}{y}\right) &\leq \frac{z}{y}, \end{aligned}$$

using monotonicity of $\exp(\cdot)$. That means $(x, y, z) \in C$ if and only if $\exists\, v$ such that

$$\begin{aligned} (2v, y, x) &\in \mathcal{K}_{\frac{1}{2}}, \\ (v, z, y) &\in \mathcal{K}_{\exp}, \end{aligned}$$

i.e. $\mathcal{C}$ is α-representable with parameter $\nu = 6$.

4.3 The dual cones

As we showed in Section 2.5, the dual cone $\mathcal{K}^*$ is an important object for the design of primal-dual interior-point methods for convex problems in conic form. It is essential that we are able to describe $\mathcal{K}^*$ explicitly, for example to check feasibility of dual points in practical implementations. But it might also be useful or even necessary to derive a self-concordant barrier $\mathcal{K}^*$. Therefore it is important to have a closed form description of the dual cone.

We recall the definition of the dual cone,

$$\mathcal{K}^* = \{s \in \mathcal{E}^* : s^T x \geq 0, \forall\, x \in \mathcal{K}\}.$$

In this section we are going to compute the dual cones of $\mathcal{K}_\alpha^{(n)}$ and $\mathcal{K}_{\exp}$.

4.3.1 The dual of the power cone

Let us recall the definition of the high dimensional power cone. Let $\alpha_i \geq 0, i = 1, \dots, n \sum_{i=1}^n \alpha_i = 1$. Then the $(n+1)$-dimensional power cone is given by

$$\mathcal{K}_\alpha^{(n)} = \left\{ (x, z) \in \mathbb{R}_+^n \times \mathbb{R} : x_1^{\alpha_1} \cdots x_n^{\alpha_n} \geq |z| \right\}.$$

We assume without loss of generality that $0 < \alpha_i < 1$, $i = 1, \dots, n$. In fact, if $\alpha_i = 1$ for some i, then $\alpha_j = 0$ for all other j. On the other hand, if $\alpha_i = 0$ for some i, then $\mathcal{K}_\alpha^{(n)}$ decomposes into the nonnegative orthant and a lower-dimensional power cone.

Theorem 4.3.1. *Let $\alpha \in \mathbb{R}^n$ with $0 < \alpha_i < 1$, $i = 1, \dots, n$. Then it holds*

$$\left(\mathcal{K}_\alpha^{(n)}\right)^* = B_\alpha \cdot \mathcal{K}_\alpha^{(n)}, \tag{4.12}$$

with

$$B_\alpha = \begin{bmatrix} \operatorname{diag}(\alpha) & 0 \\ 0 & 1 \end{bmatrix} \succ 0.$$

Proof. Note that (4.12) is equivalent to

$$\left(\mathcal{K}_\alpha^{(n)}\right)^* = P_\alpha := \left\{ (s, w) : s \geq 0, \prod_{i=1}^n \left(\frac{s_i}{\alpha_i}\right)^{\alpha_i} \geq |w| \right\}.$$

First we will show the inclusion $P_\alpha \subseteq \left(\mathcal{K}_\alpha^{(n)}\right)^*$. Let $(s, w) \in P_\alpha$ and $(x, z) \in \mathcal{K}_\alpha^{(n)}$, i.e. $x \geq 0$ and $\prod_{i=1}^n x_i^{\alpha_i} \geq |z|$. Then

$$s^T x + wz \geq s^T x - |w| \prod_{i=1}^n x_i^{\alpha_i} \geq s^T x - \prod_{i=1}^n \left(\frac{s_i x_i}{\alpha_i}\right)^{\alpha_i} \geq 0.$$

The last inequality follows from the weighted arithmetic-geometric mean inequality. Indeed for any $y \in \mathbb{R}_+^n$ and $\alpha \in \mathbb{R}_+^n$ with $\sum_{i=1}^n \alpha_i = 1$ it holds

$$\sum_{i=1}^n \alpha_i y_i \geq \prod_{i=1}^n y_i^{\alpha_i}.$$

If we take $y_i = \frac{s_i x_i}{\alpha_i}$ (which is nonnegative), we get directly that

$$\sum_{i=1}^n \alpha_i \frac{s_i x_i}{\alpha_i} = \sum_{i=1}^n s_i x_i \geq \prod_{i=1}^n \left(\frac{s_i x_i}{\alpha_i}\right)^{\alpha_i}.$$

This means $(s, w) \in \left(\mathcal{K}_\alpha^{(n)}\right)^*$.

Conversely, let $(s, w) \in \left(\mathcal{K}_\alpha^{(n)}\right)^*$ and assume $(s, w) \notin P_\alpha$, i.e. $s \ngeq 0$ or $|w| > \prod_{i=1}^n \left(\frac{s_i}{\alpha_i}\right)^{\alpha_i}$. Let us first assume $\exists\, s_i < 0$. By assumption $s^T x + wz \geq 0$ for all

$(x,z) \in \mathcal{K}_\alpha^{(n)}$. In particular we can choose $(\bar{x},\bar{z})$ with $\bar{x}_i = 1$, $\bar{x}_j = 0$, $j \neq i$ and $\bar{z} = 0$ and we have $(\bar{x},\bar{z}) \in \mathcal{K}_\alpha^{(n)}$, but also

$$\underbrace{s^T\bar{x}}_{=s_i} + \underbrace{w\bar{z}}_{=0} = s_i < 0,$$

which is a contradiction to the assumption $(s,w) \in \left(\mathcal{K}_\alpha^{(n)}\right)^*$.

Let us then look at the situation where $|w| > \prod_{i=1}^n \left(\frac{s_i}{\alpha_i}\right)^{\alpha_i}$, which means

$$w > \prod_{i=1}^n \left(\frac{s_i}{\alpha_i}\right)^{\alpha_i} \geq 0 \tag{4.13}$$

or

$$w < -\prod_{i=1}^n \left(\frac{s_i}{\alpha_i}\right)^{\alpha_i} \leq 0. \tag{4.14}$$

The first case (4.13) is equivalent to $w \cdot \prod_{i=1}^n \left(\frac{\alpha_i}{s_i}\right)^{\alpha_i} > 1$. If we take $\bar{x} = \frac{\alpha}{s} \geq 0$ (which is to be understood componentwise) and $\bar{z} = -\prod_{i=1}^n (\bar{x}_i)^{\alpha_i} = -\prod_{i=1}^n \left(\frac{\alpha_i}{s_i}\right)^{\alpha_i}$, we get

$$\begin{aligned} s^T\bar{x} + w\bar{z} &= \underbrace{\sum_{i=1}^n s_i \frac{\alpha_i}{s_i}}_{=1} - \underbrace{w \cdot \prod_{i=1}^n \left(\frac{\alpha_i}{s_i}\right)^{\alpha_i}}_{\overset{(4.13)}{>} 1} \\ &< 1 - 1 = 0. \end{aligned}$$

In the latter case (4.14) we can take the same $\bar{x}$ as before and define $\bar{z} = \prod_{i=1}^n (\bar{x}_i)^{\alpha_i} = \prod_{i=1}^n \left(\frac{\alpha_i}{s_i}\right)^{\alpha_i}$ and we get

$$s^T\bar{x} + w\bar{z} = \underbrace{\sum_{i=1}^n \alpha_i}_{=1} + \underbrace{w \cdot \prod_{i=1}^n \left(\frac{\alpha_i}{s_i}\right)^{\alpha_i}}_{\overset{(4.14)}{<} -1} < 1 - 1 = 0.$$

That means in all cases we can define $(\bar{x},\bar{z}) \in \mathcal{K}_\alpha^{(n)}$ such that

$$s^T\bar{x} + w\bar{z} < 0,$$

which contradicts the original assumption $(s,w) \in \left(\mathcal{K}_\alpha^{(n)}\right)^*$. This implies $\left(\mathcal{K}_\alpha^{(n)}\right)^* \subseteq P_\alpha$. □

That means we have the following implication for the self-concordant barrier for $\left(\mathcal{K}_\alpha^{(n)}\right)^*$: The dual cone $\left(\mathcal{K}_\alpha^{(n)}\right)^*$ is linked to the primal cone $\mathcal{K}_\alpha^{(n)}$ by the linear relation (4.12), i.e.

$$\left(\mathcal{K}_\alpha^{(n)}\right)^* = B_\alpha \, \mathcal{K}_\alpha^{(n)},$$

where B_α is the positive definite matrix defined in Theorem 4.3.1. Since B_α is positive definite, it means that $\mathcal{K}_\alpha^{(n)}$ is self-dual in the broader sense, i.e. it is self-dual with respect to the inner product induced by the positive definite matrix B_α. However, as we have mentioned at the beginning of Section 3.1, $\mathcal{K}_\alpha$ is not homogeneous (unless $\alpha \in \{0, \frac{1}{2}, 1\}$).

In view of Section 2.4.3 a we can use the $(n+1)$-self-concordant barrier for $\mathcal{K}_\alpha^{(n)}$ (see Theorem 3.1.1) to construct an $(n+1)$-self-concordant barrier for $\mathcal{K}_\alpha^{(n)}$ in the following way:

$$\bar{F}_\alpha(s, w) = F_\alpha\left(B_\alpha^{-1}\begin{bmatrix} s \\ w \end{bmatrix}\right).$$

Unfortunately $\bar{F}_\alpha$ and F_α are not conjugate to each other. This can be checked for example by (2.46), which is violated for example for the particular point $x = (1, 1, 0) \in \operatorname{int}\mathcal{K}_\alpha$.

4.3.2 The dual of the exponential cone

Let us introduce the following cone

$$P^0_{\exp} = \left\{ s : s_1 < 0, s_2 \geq 0, \exp\left(\frac{s_3}{s_1}\right) \leq \frac{e \cdot s_2}{-s_1} \right\}$$

and its closure

$$P_{\exp} = \operatorname{cl}\left(P^0_{\exp}\right).$$

Lemma 4.3.2. *It holds*

$$P_{\exp} = P^0_{\exp} \bigcup \{0\} \times \mathbb{R}_+ \times \mathbb{R}_+.$$

Proof. This proof is very similar to the one of Lemma 4.2.1.
We denote $\bar{P}_{\exp} = P^0_{\exp} \bigcup \{0\} \times \mathbb{R}^2_+$ and show first that $\bar{P}_{\exp} \subseteq P_{\exp}$. Let $s \in \bar{P}_{\exp}$. If $s \in P^0_{\exp}$ then we automatically have also that $s \in P_{\exp}$ since $P^0_{\exp} \subseteq \operatorname{cl}(P^0_{\exp}) = P_{\exp}$. Let $s = (s_1, s_2, s_3) \in \{0\} \times \mathbb{R}^2_+$. In order to prove that $s \in P_{\exp}$ we need to show that s is an accumulation point of a sequence $\{s^{(k)}\} \subset P^0_{\exp}$.
For $k = 1, 2, \ldots$ we define the sequence $s^{(k)} = (s_1^{(k)}, s_2^{(k)}, s_3^{(k)})$ with

$$\begin{aligned} s_1^{(k)} &= -\frac{1}{k^2} < 0, \ \forall \text{ integer } k \geq 1, \\ s_2^{(k)} &= s_2 + \frac{1}{k} > 0, \ \forall \text{ integer } k \geq 1, \\ s_3^{(k)} &= s_3 + \frac{1}{k}. \end{aligned}$$

We obtain the following chain of inequalities.

$$\begin{aligned}\exp\left(\frac{s_3^{(k)}}{s_1^{(k)}}\right) = \exp\left(\frac{-\frac{1}{k}+s_3}{-\frac{1}{k^2}}\right) &= \exp\Bigg(-k \underbrace{-k^2 s_3}_{\leq 0}\Bigg)\\ &\leq \exp(-k) \leq \exp(-1) < 1\\ &\leq k \leq e(k+k^2 s_2)\\ &= \frac{e(s_2+\frac{1}{k})}{\frac{1}{k^2}} = \frac{e s_2^{(k)}}{-s_1^{(k)}}.\end{aligned}$$

That means the sequence $\{s^{(k)}\} \subset P_{\exp}^0$ for all $k = 1, 2, \dots$ and we have $s^{(k)} \to (0, s_2, s_3)$ as $k \to \infty$.

Conversely, let $s \in P_{\exp}$ and assume $s \notin \bar{P}_{\exp}$. It is easy to see that the only possibility for this to happen is when $s_1 = 0$, $s_2 \geq 0$ and $s_3 < 0$. However, for any sequence $\{s^{(k)}\} \subset P_{\exp}^0$ with $s_1^{(k)} \to s_1 = 0$ and $s_3^{(k)} \to s_3 < 0$, we must have that $s_3^{(k)} < 0$ for some k and also $s_1^{(k)} < 0$ for all k. It follows that the fraction $\frac{s_3^{(k)}}{s_1^{(k)}}$ will be positive and converging to $+\infty$ for k large enough. But this contradicts the assumption $\{s^{(k)}\} \subset P_{\exp}^0$ because the inequality

$$\exp\left(\frac{s_3^{(k)}}{s_1^{(k)}}\right) \leq \frac{e \cdot s_2^{(k)}}{-s_1^{(k)}}$$

must be violated for some k sufficiently large, because the left term tends much faster to $+\infty$ as compared to the right term. That means $P_{\exp} \subseteq \bar{P}_{\exp}$ and finishes the proof. □

In the following theorem we are going to give an explicit description of the dual cone of $\mathcal{K}_{\exp}$. We have shown in Section 4.2 that $\mathcal{K}_{\exp}$ can in fact be seen as a limit of a linear transformation of the power cone $\mathcal{K}_\alpha$, namely

$$\tilde{\mathcal{K}_\alpha} = \{(z_1, z_2, z_3) : z_2^\alpha z_3^{1-\alpha} \geq |z_3 + \alpha z_1|\} = A_\alpha^{-1}\,\mathcal{K}_\alpha,$$

where

$$A_\alpha = \begin{bmatrix} 0 & 1 & 0\\ 0 & 0 & 1\\ \alpha & 0 & 1\end{bmatrix},$$

because $z \in \tilde{\mathcal{K}_\alpha}$ if and only if $A_\alpha\, z = (z_2, z_3, \alpha z_1 + z_3) \in \mathcal{K}_\alpha$, or equivalently $z \in A_\alpha^{-1}\,\mathcal{K}_\alpha$. In Theorem 4.3.1 we have shown that also between $\mathcal{K}_\alpha$ and its dual cone a linear relation exists, namely $\mathcal{K}_\alpha^* = B_\alpha\,\mathcal{K}_\alpha$ where

$$B_\alpha = \begin{bmatrix} \alpha & 0 & 0\\ 0 & 1-\alpha & 0\\ 0 & 0 & 1\end{bmatrix}$$

It would be natural to use these two results with the aim to give a compact description of the dual cone of $\mathcal{K}_{\exp}$. Unfortunately, we see that both A_α and B_α converge to nonsingular matrices as $\alpha \to 0$. Therefore we have to compute directly the dual cone of $\mathcal{K}_{\exp}$.

Theorem 4.3.3. *We have*

$$(\mathcal{K}_{\text{exp}})^* = P_{\text{exp}}. \tag{4.15}$$

Proof. Let us first show the inclusion $P_{\text{exp}} \subseteq \mathcal{K}^*_{\text{exp}}$. Let $s \in P_{\text{exp}}$. According to Lemma 4.3.2 it holds either $s \in P^0_{\text{exp}}$ or $s \in \{0\} \times \mathbb{R}^2_+$. In the latter case we have in view of Lemma 4.2.1 for any $x \in \mathcal{K}_{\text{exp}}$

$$s^T x = \underbrace{s_1}_{=0} x_1 + \underbrace{s_2}_{\geq 0} \underbrace{x_2}_{\geq 0} + \underbrace{s_3}_{\geq 0} \underbrace{x_3}_{\geq 0} \geq 0.$$

Let $s \in P^0_{\text{exp}}$ and $x \in \mathcal{K}_{\text{exp}}$. According to Lemma 4.2.1 we have that $x \in \mathcal{K}^0_{\text{exp}}$ or $x \in (-\mathbb{R}_+) \times \mathbb{R}_+ \times \{0\}$. In the latter case we get

$$s^T x = \underbrace{s_1}_{<0} \underbrace{x_1}_{\leq 0} + \underbrace{s_2}_{\geq 0} \underbrace{x_2}_{\geq 0} + s_3 \underbrace{x_3}_{=0} \geq 0.$$

Let $s \in P^0_{\text{exp}}$ and $x \in \mathcal{K}^0_{\text{exp}}$. In particular we have then $s_2 \geq (-s_1) \exp\left(\frac{s_1 - s_3}{-s_1}\right)$ and also $x_2 \geq x_3 \exp\left(\frac{x_1}{x_3}\right)$. By using these lower bounds on s_2 and x_2, we get

$$\begin{aligned}
s^T x &= s_1 x_1 + \underbrace{s_2}_{\geq 0} x_2 + s_3 x_3 \\
&\geq s_1 x_1 + s_2 \underbrace{x_3 \exp\left(\frac{x_1}{x_3}\right)}_{>0} + s_3 x_3 \\
&\geq s_1 x_1 + (-s_1) \exp\left(\frac{s_1 - s_3}{-s_1}\right) x_3 \exp\left(\frac{x_1}{x_3}\right) + s_3 x_3.
\end{aligned}$$

After merging the two exponential expressions, the middle term can be simplified and we obtain

$$\begin{aligned}
s^T x &\geq s_1 x_1 + (-s_1) x_3 \exp\left(\frac{s_1 - s_3}{-s_1} + \frac{x_1}{x_3}\right) + s_3 x_3 \\
&= s_1 x_1 + \underbrace{(-s_1) x_3}_{>0} \exp\left(\frac{s_1 x_3 - s_3 x_3 - s_1 x_1}{-s_1 x_3}\right) + s_3 x_3.
\end{aligned}$$

Using the fact that $\exp(\tau) \geq \tau + 1$ for all τ, we can bound the exponential term and get

$$\begin{aligned}
s^T x &\geq s_1 x_1 + (-s_1) x_3 \cdot \left(\frac{s_1 x_3 - s_3 x_3 - s_1 x_1}{-s_1 x_3} + 1\right) + s_3 x_3 \\
&= s_1 x_1 + s_1 x_3 - s_3 x_3 - s_1 x_1 + (-s_1) x_3 + s_3 x_3 \\
&= 0.
\end{aligned}$$

This means $s \in \mathcal{K}^*_{\text{exp}}$. Thus, $P_{\text{exp}} \subseteq \mathcal{K}^*_{\text{exp}}$.

Now, let us show the inverse inclusion $\mathcal{K}^*_{\text{exp}} \subseteq P_{\text{exp}}$. Let $s \in \mathcal{K}^*_{\text{exp}}$ and let us assume $s \notin P_{\text{exp}}$. In view of Lemma 4.3.2 we have that P_{exp} is the union of two convex cones. That means $s \notin P_{\text{exp}}$ means $s \notin P^0_{\text{exp}}$ (i.e. $s_1 \geq 0$ or $s_2 < 0$ or $\exp(\frac{s_3}{s_1}) > \frac{e s_2}{-s_1}$) and $s \notin \{0\} \times \mathbb{R}^2_+$ (i.e. $s_1 \neq 0$ or $s_2 < 0$ or $s_3 < 0$). In total we have nine cases to distinguish.

- Among these nine cases there are five cases that include the inequality $s_2 < 0$. In that case we can take $x = (0, 1, 0) \in \mathcal{K}_{\text{exp}}$ and get $s^T x = s_2 < 0$, which means that s cannot be in the dual cone $\mathcal{K}^*_{\text{exp}}$.

- A sixth case is the situation where $s_1 \geq 0$ and $s_1 \neq 0$, i.e. $s_1 > 0$. Then we can take $x = (-1, 0, 0) \in \mathcal{K}_{\text{exp}}$ and get $s^T x = -s_1 < 0$, which means that s cannot be in $\mathcal{K}^*_{\text{exp}}$.

- Another case is the situation where $s_1 \geq 0$ and $s_3 < 0$. If now $s_2 < 0$, we can take again $x = (0, 1, 0) \in \mathcal{K}_{\text{exp}}$ which contradicts again the assumption that $s \in \mathcal{K}^*_{\text{exp}}$.
  If $s_2 = 0$, we can take $x = (0, 1, 1) \in \mathcal{K}_{\text{exp}}$ and get $s^T x = s_2 + s_3 = s_3 < 0$. Also this case is a contradiction to our assumption that $s \in \mathcal{K}^*_{\text{exp}}$.
  If $s_2 > 0$ and $s_1 > 0$ we can take $x = (-1, 0, 0) \in \mathcal{K}_{\text{exp}}$ and get $s^T x = -s_1 < 0$.
  Finally, if $s_2 > 0$ and $s_1 = 0$ we can take $x = \left(\log(\frac{-s_3}{2s_2}), \frac{-s_3}{2s_2}, 1\right)$ which is in $\mathcal{K}_{\text{exp}}$ because $x_2 = \frac{-s_3}{2s_2} \geq 0$ and $x_3 = 1 > 0$ and

  $$\exp\left(\frac{x_1}{x_3}\right) = \exp\left(\log\left(\frac{-s_3}{2s_2}\right)\right) = \frac{-s_3}{2s_2} = x_2 = \frac{x_2}{x_3}.$$

  Moreover,

  $$s^T x = s_2 \frac{-s_3}{2s_2} + s_3 = -\frac{1}{2}s_3 + s_3 = \frac{1}{2}s_3 < 0.$$

  This contradicts the assumption that $s \in \mathcal{K}^*_{\text{exp}}$.

- The last two cases are the situations where $\exp(\frac{s_3}{s_1}) > \frac{es_2}{-s_1}$ and $s_1 \neq 0$.
  If $s_1 > 0$ we can take $x = (-1, 0, 0) \in \mathcal{K}_{\text{exp}}$ and get $s^T x = -s_1 < 0$.
  If $s_1 < 0$ and $s_2 < 0$, we can take $x = (0, 1, 0) \in \mathcal{K}_{\text{exp}}$ and get as before $s^T x < 0$.
  If $s_1 < 0$ and $s_2 = 0$ we can take $x = (1, \frac{\exp(t)}{t}, \frac{1}{t})$ for $t > 0$. We see that $x \in \mathcal{K}_{\text{exp}}$ since $x_2 > 0$ and $x_3 > 0$ for all $t > 0$ and

  $$\exp\left(\frac{x_1}{x_3}\right) = \exp(t) = \frac{\frac{\exp(t)}{t}}{\frac{1}{t}} = \frac{x_2}{x_3}.$$

  Moreover,

  $$s^T x = \underbrace{s_1} < 0 + s_3 \frac{1}{t} < 0$$

  for t sufficiently large.
  If $s_1 < 0$ and $s_2 > 0$, then we can divide

  $$\exp\left(\frac{s_3}{s_1}\right) > \frac{e \cdot s_2}{-s_1}$$

  by e and get the equivalent inequality

  $$\exp\left(\frac{s_1 - s_3}{-s_1}\right) > \frac{s_2}{-s_1}. \tag{4.16}$$

In that case we define $x_1 = \frac{s_1 - s_3}{s_1 s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right)$, $x_2 = \frac{1}{s_2} > 0$ and $x_3 = \frac{1}{s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right) > 0$ and we see that

$$\exp\left(\frac{x_1}{x_3}\right) = \exp\left(\frac{s_1 - s_3}{s_1}\right) = \frac{x_2}{x_3},$$

which means $x \in \mathcal{K}_{\exp}$. On the other hand it holds

$$\begin{aligned} s^T x &= s_1 x_1 + s_2 x_2 + s_3 x_3 \\ &= \frac{s_1 - s_3}{s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right) + 1 + \frac{s_3}{s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right) \\ &= \frac{1}{s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right) \cdot (s_1 - s_3 + s_3) + 1 \\ &= \frac{s_1}{s_2} \cdot \exp\left(\frac{s_1 - s_3}{-s_1}\right) + 1. \end{aligned}$$

The last term is negative if and only if

$$\exp\left(\frac{s_1 - s_3}{-s_1}\right) > \frac{s_2}{-s_1}.$$

But this is exactly (4.16), which means we have found a point $x \in \mathcal{K}_{\exp}$ such that $s^T x < 0$. This contradicts the assumption that $s \in \mathcal{K}^*_{\exp}$.

That means in all cases where $s \notin P_{\exp}$ we were able to find an appropriate point $x \in \mathcal{K}_{\exp}$ such that $s^T x < 0$, which means that $s \notin \mathcal{K}^*_{\exp}$. Thus, $\mathcal{K}^*_{\exp} \subseteq P_{\exp}$. □

In view of the definition of $P^0_{\exp}$ and $\mathcal{K}^0_{\exp}$ we see that there is the following scaling relation between both cones. Let us define the scaling matrix $B_{\exp}$ and its inverse

$$B_{\exp} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & \frac{1}{e} & 0 \\ -1 & 0 & 0 \end{bmatrix} \qquad (B_{\exp})^{-1} = \begin{bmatrix} 0 & 0 & -1 \\ 0 & e & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Then we have

$$P^0_{\exp} = B_{\exp} \mathcal{K}^0_{\exp}.$$

Indeed, let $s \in P^0_{\exp}$. Then the above identity is true if and only if

$$(B_{\exp})^{-1} s \in \mathcal{K}^0_{\exp}.$$

We have that $B^{-1}_{\exp} s = (-s_3, e s_2, -s_1)$ and the above inclusion means that it should hold $e s_2 \geq 0$, $-s_1 > 0$ and

$$\exp\left(\frac{-s_3}{-s_1}\right) \leq \frac{e s_2}{-s_1}.$$

We see that all three inequalities are true since $s \in P^0_{\exp}$.

This observation implies that for points in the interior of $\mathcal{K}_{\exp}^*$ we have the following representation:

$$s \in \operatorname{int} \mathcal{K}_{\exp}^* \qquad \Leftrightarrow \qquad (B_{\exp})^{-1} s \in \operatorname{int} \mathcal{K}_{\exp}. \tag{4.17}$$

Note that nevertheless $\mathcal{K}_{\exp}$ is not self-dual, even not in its broader sense (see e.g. [28]). Self-duality in the less restrictive version means that there should exist an inner product induced by a positive definite matrix M such that

$$\mathcal{K}^* = \{s : \langle s, x \rangle_M \geq 0, \ \forall\, x \in \mathcal{K}\}$$

is equal[1] to $\mathcal{K}$. In other words for any $x_1 \in \operatorname{int} \mathcal{K}_{\exp}$ and any $x_2 \in \operatorname{int} \mathcal{K}_{\exp}$ we should find a matrix $M \succ 0$ such that $x_1^T M x_2 \geq 0$. However, this means that $s := M x_1 \in \mathcal{K}_{\exp}^*$ with respect to the Euclidean inner product. But as we have seen, we have then

$$M x_1 \in \mathcal{K}_{\exp}^* = B_{\exp} \mathcal{K}_{\exp}$$

which implies that $M = B_{\exp}$, because x_1 can be *any* point in $\operatorname{int} \mathcal{K}_{\exp}$. However, we see that $B_{\exp}$ is *not* positive definite and therefore $\mathcal{K}_{\exp}$ cannot be self-dual.

On the other hand, we can use the scaling relation (4.17) to derive a 3-self-concordant barrier for $\mathcal{K}_{\exp}^*$, namely

$$\tilde{F}_{\exp}(s) := F_{\exp}\left(B_{\exp}^{-1} s\right).$$

where $F_{\exp}$ is the 3-self-concordant barrier for $\mathcal{K}_{\exp}$. Unfortunately $\tilde{F}_{\exp}$ and $F_{\exp}$ are not conjugate to each other. This can be checked for example by (2.46), which is violated for example for the particular point $x = (0, e, 1) \in \operatorname{int} \mathcal{K}_{\exp}$.

4.4 Examples of α-representable problem classes

In the previous sections we showed that many sets and functions are representable in terms of the power cone $\mathcal{K}_\alpha$ (and its limit $\mathcal{K}_{\exp}$). Moreover, for both cones and their duals self-concordant barriers are available. We will see now how these results can be used to find the α-representation of two concrete classes of convex optimization problems.

We recall here the definition of the α-representation of a *set* $\mathcal{C} \subset \mathbb{R}^n$ (α-REP): There should exist finite integers M_1 and M_2, matrices A_α, $A_{\exp}$, A_f and vectors c_α, $c_{\exp}$ and c_f in appropriate sizes such that

$$u \in \mathcal{C} \Leftrightarrow \begin{cases} c_\alpha - A_\alpha^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_{M_1}} \\ c_{\exp} - A_{\exp}^T \begin{bmatrix} u \\ v \end{bmatrix} \in \mathcal{K}_{\exp} \times \cdots \times \mathcal{K}_{\exp} \ \ (M_2 \text{ times}), \\ A_f^T \begin{bmatrix} u \\ v \end{bmatrix} = c_f. \end{cases} \tag{α-REP}$$

[1] Note that in Definition 2.5.1 we have chosen the particular Euclidean inner product

for some artificial modelling variables v.

In this section we are seeking a dual *conic α-representation of convex optimization problems*, i.e. problems of the form

$$(P) \qquad \min_u f(u) \\ u \in \mathcal{C},$$

where f is a convex function and $\mathcal{C}$ is a convex set. The α-representation of (P) should satisfy the following three criteria: the conic reformulation of (P) should

1. be equivalent to (P), in the sense that an optimal solution of the reformulation can be used to derive an optimal solution of (P),
2. be in dual conic form,
3. only involve power cones $\mathcal{K}_\alpha$, exponential cones $\mathcal{K}_{\exp}$ or linear equality constraints.

The reformulations are done in two steps. First, we ensure that the objective function becomes linear, because we wish to have a problem in conic form. This can always be done, for example by going to the epigraph form of (P). We obtain then a problem of the form

$$(\tilde{P}) \qquad \min_{u,w} b^T \begin{bmatrix} u \\ w \end{bmatrix} \\ (u,w) \in \tilde{\mathcal{C}},$$

where w is one (or more) epigraph variable(s), $\tilde{\mathcal{C}}$ is the extended feasible set (that is obtained when adding to $\mathcal{C}$ the epigraph constraint(s) of the form $f(u) \leq w$) and b is the new objective vector.
The second step is the process of finding an α-representation of $\tilde{\mathcal{C}}$ as described above in (α-REP).

4.4.1 Location problem

We consider a constrained generalized location problem, where the sum of the distances of a point $u \in \mathbb{R}^N$ to given locations $C_j \in \mathbb{R}^N, j = 1, \ldots, M$, shall be minimized, subject to the constraint that u should not be too far from other given locations $D_j \in \mathbb{R}^N, j = 1, \ldots, R$. Let $a_j > 0, j = 1, \ldots, M$, be some positive weights, $p_j \geq 1, j = 1, \ldots, M$, parameters that determine the norms that define the distance to the locations C_j, $q_j \geq 1, j = 1, \ldots, R$, parameters that determine the norms that define the distance to the locations D_j and $r_j > 0, j = 1, \ldots, R$, given distance values.

$$\begin{aligned} \min_{u \in \mathbb{R}^N} \quad & \sum_{j=1}^{M} a_j ||u - C_j||_{p_j} \\ \text{s.t.} \quad & ||u - D_j||_{q_j} \leq r_j, \ j = 1, \ldots, R. \end{aligned} \qquad (LOC_0)$$

Note that the distances are defined using p-norms, as opposed to standard Euclidean norms.

Introducing epigraph variables w_j for each norm term in the objective, the formulation (LOC_0) can be written as

$$\begin{aligned} \min_{u\in\mathbb{R}^N, w\in\mathbb{R}^M} \quad & \sum_{j=1}^{M} a_j w_j \\ \text{s.t.} \quad & ||u - C_j||_{p_j} \le w_j, \ j = 1, \dots, M \\ & ||u - D_j||_{q_j} \le r_j, \ j = 1, \dots, R. \end{aligned} \tag{LOC_1}$$

Using the α-representability of $\mathcal{P}_p^{(N)}$ (i.e $(x, \tau) \in \mathcal{P}_p^{(N)}$ if and only if $\exists\ v \ge 0 :$ $(v_i, \tau, x_i) \in \mathcal{K}_\alpha$ and $\sum_{i=1}^N v_i = \tau$, where $\alpha = \frac{1}{p}$), we define $\alpha_j = \frac{1}{p_j}$ for $j = 1, \dots, M$ and $\alpha_{M+j} = \frac{1}{q_j}$ for $j = 1, \dots, R$. We arrive at the following α-representation of (LOC_0):

$$\begin{aligned} \min_{u,w,\tilde{v},\bar{v}} \quad & \sum_{j=1}^{M} a_j w_j \\ \text{s.t.} \quad & (\tilde{v}_{i,j}, w_j, u_i - C_{i,j}) \in \mathcal{K}_{\alpha_j}, i = 1, \dots, N, j = 1, \dots, M \\ & (\bar{v}_{i,j}, r_j, u_i - D_{i,j}) \in \mathcal{K}_{\alpha_{M+j}}, i = 1, \dots, N, j = 1, \dots, R \\ & \sum_{i=1}^{N} \tilde{v}_{i,j} = w_j, \ j = 1, \dots, M \\ & \sum_{i=1}^{N} \bar{v}_{i,j} = r_j, \ j = 1, \dots, R. \end{aligned} \tag{LOC_2}$$

We see that (LOC_2) is a dual conic formulation using entirely power cones $\mathcal{K}_\alpha$ and linear equality constraints. Moreover, since the epigraph formulation (LOC_1) is equivalent to (LOC_0) and because p-cones $\mathcal{P}_p^{(N)}$ are α-representable, it holds that (LOC_2) is equivalent to the original formulation (LOC_0).

4.4.2 Geometric programming

An important class of optimization problems are so-called geometric programs (see e.g. [16], [5]). A geometric program is a minimization problem involving a posynomial objective, posynomial inequality constraints and monomial equality constraints. We saw these kind of expressions already in Section 4.2.2. Let us introduce the following piece of notation: for vectors $x \in \mathbb{R}^N_{++}$ and $a \in \mathbb{R}^N$ we denote $x^a = \prod_{i=1}^N x_i^{a_i}$.

Let $x \in \mathbb{R}^N$ be positive variables, $D_{i,j}^{(\text{pos})}$ and $e_j^{(\text{mon})}$ positive coefficients and $\mathbf{K}_{i,j}^{(\text{pos})} \in \mathbb{R}^N$, $i = 1, \dots, n_j, j = 0, \dots, M$ and $K_j^{(\text{mon})} \in \mathbb{R}^N$, $j = 1, \dots, M_{\text{mon}}$ real exponents. We assume that the matrix

$$\mathbf{K}^{(\mathrm{pos})} := \left[\mathbf{K}^{(\mathrm{pos})}_{1,0}, \cdots, \mathbf{K}^{(\mathrm{pos})}_{n_0,0}, \cdots, \mathbf{K}^{(\mathrm{pos})}_{1,M}, \cdots, \mathbf{K}^{(\mathrm{pos})}_{n_M,M}\right] \in \mathbb{R}^{N \times \sum_{j=0}^{M} n_j}$$

has full row rank and the matrix

$$K^{(\mathrm{mon})} := [K^{(\mathrm{mon})}_1, \cdots, K^{(\mathrm{mon})}_{M_{\mathrm{mon}}}]$$

has full column rank (that is, the vectors $K^{(\mathrm{mon})}_j$ are linearly independent).

A geometric program in posynomial form is given by

$$\begin{aligned} \min_{x>0} & \sum_{i=1}^{n_0} D^{(\mathrm{pos})}_{i,0} \cdot x^{\mathbf{K}^{(\mathrm{pos})}_{i,0}} \\ \text{s.t.} & \sum_{i=1}^{n_j} D^{(\mathrm{pos})}_{i,j} \cdot x^{\mathbf{K}^{(\mathrm{pos})}_{i,j}} \leq 1, \ \ j = 1, \ldots, M, \\ & e^{(\mathrm{mon})}_j \cdot x^{K^{(\mathrm{mon})}_j} = 1, \ \ j = 1, \ldots, M_{\mathrm{mon}}. \end{aligned} \tag{GP}$$

Under the above assumptions we apply a change of variables $x_k = \exp(u_k)$ and $D^{(\mathrm{pos})}_{i,j} = \exp\left(C^{(\mathrm{pos})}_{i,j}\right)$, and we get a geometric program in convex form, i.e.

$$\begin{aligned} \min_{u} & \sum_{i=1}^{n_0} \exp\left(u^T \mathbf{K}^{(\mathrm{pos})}_{i,0} + C^{(\mathrm{pos})}_{i,0}\right) \\ \text{s.t.} & \sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}^{(\mathrm{pos})}_{i,j} + C^{(\mathrm{pos})}_{i,j}\right) \leq 1, \ \ j = 1, \ldots, M, \\ & u^T K^{(\mathrm{mon})}_j + \log\left(e^{(\mathrm{mon})}_j\right) = 0, \ \ j = 1, \ldots, M_{\mathrm{mon}}. \end{aligned} \tag{GP_0}$$

We introduce one epigraph variable w and rewrite (GP_0) in its equivalent epigraph form

$$\begin{aligned} \min_{u,w} \ & w \\ \text{s.t.} & \sum_{i=1}^{n_0} \exp\left(u^T \mathbf{K}^{(\mathrm{pos})}_{i,0} + C^{(\mathrm{pos})}_{i,0}\right) \leq w \\ & \sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}^{(\mathrm{pos})}_{i,j} + C^{(\mathrm{pos})}_{i,j}\right) \leq 1, \ \ j = 1, \ldots, M, \\ & u^T K^{(\mathrm{mon})}_j + \log\left(e^{(\mathrm{mon})}_j\right) = 0, \ \ j = 1, \ldots, M_{\mathrm{mon}}. \end{aligned} \tag{GP_1}$$

We showed in Section 4.2.2 that constraints of the form $\sum_{i=1}^{n} \exp(a_i^T u + c_i) \leq w$ are α-representable using the exponential cone $\mathcal{K}_{\exp}$. It follows that the α-representation of (GP_1) is

$$\begin{aligned} \min_{u,w,v} \quad & w \\ \text{s.t.} \quad & \left(u^T \mathbf{K}_{i,j}^{(\text{pos})} + C_{i,j}^{(\text{pos})}, v_{i,j}, 1\right) \in \mathcal{K}_{\exp}, \quad j = 0, \ldots, M, i = 1, \ldots, n_j \\ & \sum_{i=1}^{n_0} v_{i,0} = w, \; \sum_{i=1}^{n_j} v_{i,j} = 1, \quad j = 1, \ldots, M, \\ & u^T K_j^{(\text{mon})} + \log\left(e_j^{(\text{mon})}\right) = 0, \quad j = 1, \ldots, M_{\text{mon}}. \end{aligned} \tag{GP_2}$$

We see that (GP_2) is a dual conic formulation using entirely exponential cones $\mathcal{K}_{\exp}$ and linear equality constraints. Moreover, since the epigraph formulation (GP_1) is equivalent to (GP_0) we have that (GP_2) is equivalent to the original formulation (GP_0).

4.5 Numerical results

We have shown in Section 4.4 that two important classes of convex optimization problems (geometric programs and generalized location problems) can be cast in dual conic form using entirely the power cone $\mathcal{K}_\alpha$ and the exponential cone $\mathcal{K}_{\exp}$. For the feasible set of both conic reformulations we have explicit self-concordant barriers available, which means in view of Section 2.4 that these problems can be solved in polynomial time (compare Theorem 2.4.15 and Theorem 2.5.18). In this section we are going to verify this theoretical result with a practical Matlab implementation of Algorithm 4 and 5. Both algorithms are included in the nonsymmetric conic solver that is presented in more detail in Appendix A.

To improve the practical performance we implemented for both algorithms a linesearch along the Newton directions. For Algorithm 4 we know that $\alpha_0 = \frac{1}{1+\delta}$, where δ denotes the Newton decrement at the current iterate for a step towards the current target point on the central path, is a step size that guarantees feasibility of the step and a decrease of the objective value of the centering problem (compare Theorem 2.3.6). Starting at α_0 we gradually increase this step size (by multiplying the current feasible step size with 2) until we arrive at an infeasible point or until the objective value increases again.

For the centering steps in Algorithm 5 we use the same strategy for the choice of the step size as presented above. In addition, we proposed a safe step size parameter α_0 for the primal-dual affine-scaling direction, that guarantees a sufficient increase in the duality measure t (see Theorem 2.5.16). On the other hand we make sure not to drift too far away from the primal-dual central path (see Theorem 2.5.15). Starting at α_0 we gradually increase this step size (again by multiplying with 2) until we reach a given proximity to the boundary of the primal-dual cone $\mathcal{K} \times \mathcal{K}^*$. In the implementation we have chosen this proximity value to be 90%, that is, if $\alpha_{\max}$ is the largest value so that $(x + \alpha_{\max}\Delta x, s + \alpha_{\max}\Delta s) \in \mathcal{K} \times \mathcal{K}^*$, then we take as step size $\alpha = 0.9 \cdot \alpha_{\max}$. Note that this strategy is not covered anymore by the complexity result because we cannot bound the proximity to the primal-dual central path. However, since we have observed that this strategy is superior to the conservative step size proposed in Algorithm 5, we have decided to use it in the numerical tests.

For all methods tested we have chosen as absolute accuracy in terms of the objective value $\epsilon = 10^{-6}$. In the implementation of Algorithm 4 we have set as centering accuracy $\beta = 0.25$ and as update coefficient of the duality measure $\theta = 10$. In Algorithm 5 we have chosen as centering accuracy $\beta = \frac{-3+\sqrt{17}}{8}$.

4.5.1 Location problems

Let us recall the definition of the generalized location problem (LOC_0), i.e.

$$\begin{aligned} \min_{u\in\mathbb{R}^N} \; & \sum_{j=1}^{M} a_j ||u - C_j||_{p_j} \\ \text{s.t. } & ||u - D_j||_{q_j} \le r_j, \; j = 1, \dots, R. \end{aligned} \qquad (LOC_0)$$

In Section 4.4.1 we saw that (LOC_0) can be cast in dual conic form

$$\begin{aligned} & \max \; b^T y \\ & s.t. \; c - A^T y \in \mathcal{K}^*, \\ & \quad A_f^T y = c_f, \end{aligned} \qquad (D - LOC)$$

with $y = (u, w, v) \in \mathbb{R}^{N+M+NM}$, where $w \in \mathbb{R}^M$ and $v \in \mathbb{R}^{NM}$ are artificial variables, the data A, A_f, c, c_f and b are suitably chosen and $\mathcal{K}^*$ is a direct product of $N(M + R)$ power cones $\mathcal{K}_\alpha$. Using the 3-self-concordant barrier for the power cone $\mathcal{K}_\alpha$, we obtain a ν-self-concordant barrier for the feasible set of $(D - LOC)$, where $\nu = 3N(M + R)$.

Let us verify that assumptions (2)-(4) on page 66 are satisfied.
One can verify that A is given by

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix} \in \mathbb{R}^{N+M+N(M+R),3N(M+R)},$$

where

$$\begin{aligned} A_1 &= \texttt{rep}_{MR}(\texttt{blkdiag}_N([0, 0, -1])) \in \mathbb{R}^{N\times 3\cdot N(M+R)}, \\ A_2 &= [\texttt{blkdiag}_M(\texttt{rep}_N([0, -1, 0])), 0] \in \mathbb{R}^{M\times 3\cdot N(M+R)}, \\ A_3 &= \texttt{blkdiag}_{N(M+R)}([-1, 0, 0]) \in \mathbb{R}^{N(M+R)\times 3\cdot N(M+R)}. \end{aligned}$$

Here $\texttt{rep}_N(X)$ denotes the N times repetition of a matrix X, i.e.

$$\texttt{rep}_N(X) = [\underbrace{X, \cdots, X}_{N \text{ times}}],$$

and $\texttt{blkdiag}_N(X)$ denotes the block diagonal matrix containing N copies of X.

By permutation of the columns we see immediately that A has full row rank.

Further, A_f has full column rank since the linear equality constraints in (LOC_2) are given by

$$B\begin{bmatrix}\tilde{v}\\ \bar{v}\end{bmatrix}+E\begin{bmatrix}u\\ w\end{bmatrix}=d$$

for some matrix E and some vector d and a matrix B that has the following structure

$$B=\texttt{blkdiag}(\mathbf{1}_N)\in\mathbb{R}^{(M+R)\times N(M+R)},$$

where $\mathbf{1}_N$ is a row vector of ones of size N. It is clear that B has full row rank. Therefore the matrix A_f in the conic formulation $(D{-}LOC)$ must have full column rank.

To show the last condition, let $y=(u,w,v)$ such that $c-A^Ty\in\operatorname{int}\mathcal{K}^*$ and $A_f^Ty=c_f$, and let $(\Delta u,\Delta w,\Delta v)\neq 0$ be any direction.

If $\Delta v\neq 0$ or $\Delta w\neq 0$, then it is possible to find a step size γ such that $w+\gamma\Delta w\not> 0$ or $v+\gamma\Delta v\not> 0$. Let $\Delta v=0$ and $\Delta w=0$. Then there exists an index i such that $\Delta u_i\neq 0$ (otherwise $(\Delta u,\Delta w,\Delta v)=0$) and it is possible to find a step size γ such that $|u_i+\gamma\Delta u_i-C_{i,j}|>\tilde{v_{i,j}}^{\alpha_j}w_j^{1-\alpha_j}$ for some (i,j). That means for any feasible point y and any direction $\Delta y\neq 0$ we cannot extend this direction to $\pm\infty$ without leaving the feasible region. In other words, the feasible region of $(D-LOC)$ does not contain straight lines and therefore the Newton directions are defined everywhere (see Theorem 2.2.4).

Comparison with nonlinear solvers with an AMPL interface

For the numerical tests we consider the unconstrained version of (LOC_0), i.e. for $R=0$ we have

$$\min_{u\in\mathbb{R}^N}\sum_{j=1}^{M}a_j||u-C_j||_{p_j},\tag{4.18}$$

where for $j=1,\ldots,M$ the parameters p_j are uniformly distributed on the [1 3] interval, the facilities C_j are uniformly distributed in the $[0\ 1]^N$ box and the weights a_j on the objective terms are set to be all equal to 1. Including additional p-norm constraints (as in the general formulation (LOC_0)) would be straightforward. However, we have chosen to omit these constraints since the unconstrained problem (4.18) is easy to initialize. Indeed, as (4.18) is unconstrained we can choose for example $u^{(0)}=0\in\mathbb{R}^N$, initialize $w^{(0)}\in\mathbb{R}^M$ such that $||C_j||_{p_j}<w_j^{(0)}, j=1,\ldots,M$ (e.g. $w_j=||C_j||_{p_j}+1$) and initialize $v^{(0)}$ as shown in Section 4.1.3 (compare with the α-representability of the p-cone).

We are going to solve random instances of (4.18) for different values of N and M by first reformulating them in conic form $(D-LOC)$, then solving them using the dual long-step path-following method (Algorithm 4) and the nonsymmetric primal-dual predictor-corrector method (Algorithm 5). We compare their practical performance with those of the nonlinear solvers MINOS 5.5 [2], SNOPT 6.1-1[3] and KNITRO 6.0.0[4]. All three solvers directly solve the original problem (4.18).

[2] http://www.sbsi-sol-optimize.com/asp/sol_product _minos.htm

[3] http://www.sbsi-sol-optimize.com/asp/sol_product_snopt.htm

[4] http://www.ziena.com/knitro.htm

MINOS is using a quasi-Newton method to solve nonlinear problems. SNOPT employs a sparse SQP algorithm with limited-memory quasi-Newton approximations to the Hessian of Lagrangian. An augmented Lagrangian merit function promotes convergence from an arbitrary point. KNITRO is based on a direct barrier method to solve a primal-dual KKT system using trust regions and a merit function to promote convergence.

For each choice of problem parameters (N, M) presented here, 10 different instances are solved and the average is reported. Table 4.1 reports the number of iterations (major iterations for KNITRO) carried out by each method. Let us recall here the complexity results for the dual and primal-dual algorithm. In [9] we have analyzed (4.18) and presented a complete complexity analysis for solving it (see [9, Theorem 4.1]), including the initial centering (as proposed in Section 2.4) to generate a point close to the central path. For sake of transparency we recall the result here.

dimension	D-IPM	PD-IPM	MINOS	SNOPT	KNITRO
$N = 2, M = 10$	26.8	21.1	8.1	11.1	8.6
$N = 2, M = 100$	37.0	27.9	7.2	10.1	6.1
$N = 2, M = 1000$	48.6	46.3	7.4	11.3	6.5
$N = 10, M = 10$	39.9	31.7	30.0	31.9	20.9
$N = 10, M = 50$	45.6	43.8	38.3	42.1	2034.2
$N = 10, M = 100$	55.0	49.8	25.4	28.8	13.3
$N = 10, M = 500$	68.3	64.5	31.9	41.8	35.6
$N = 50, M = 10$	51.2	56.8	251.4	131.2	1045.4
$N = 50, M = 50$	67.5	74.9	248.6	170.5	284.2
$N = 50, M = 100$	81.0	80.4	162.7	154.1	211.8
$N = 50, M = 200$	98.4	100.7	208.9	147	128.7

Table 4.1: Number of iterations for each solver (averaged on 10 instances).

Theorem 4.5.1. *Let $a_j \in [a_{\min}, 1]$, $\forall j = 1, \ldots, M$, where $a_{\min} > 0$ and $C_{i,j} \in [0, 1]$, for $j = 1, \ldots, M, i = 1, \ldots, N$. Then Algorithm 4 initialized as described above solves the unconstrained location problem* (4.18) *in*

$$\mathcal{O}(N\,M) \cdot \mathcal{O}\left(\log\left(\frac{N\,M}{\epsilon\, a_{\min}}\right)\right)$$

iterations.

Note that the above theorem is in line with Theorem 2.4.15. The only difference is that Theorem 2.4.15 assumes that the initial point is already close to the central path, while Theorem 4.5.1 does not need such an assumption. The complexity for the inital centering phase is guaranteed by the additional assumptions on the data (i.e. a and C). In accordance with Theorem 2.5.18, the complexity of the primal-dual predictor-corrector method proposed in Algorithm 5 is $\mathcal{O}(\sqrt{\nu} \cdot \log(s_0^T x_0/\epsilon))$. However, since we embed a linesearch along the primal-dual affine-scaling direction as described above, the complexity result is no longer valid. On the other hand we see that the number of iterations is comparable to the dual long-step method, and both are clearly better than the pessimistic bound of $\mathcal{O}(NM)$. Indeed, the number of iterations increases only by a factor of less than 5 when going from $NM = 20$ to $NM = 10000$.

When comparing the number of iterations for the dual and primal-dual method to those of the three nonlinear solvers, we see that for the larger values of N the iteration count becomes comparable to - if not better than - those of the three nonlinear solvers MINOS, SNOPT and KNITRO. The guidance of the central path is thus clearly beneficial here.

It is interesting to see that when fixing N and increasing the value of M, the number of iterations for all the AMPL nonlinear solvers remains constant or even decreases (apart from some instances that were difficult for KNITRO). This effect could be explained by a "smoothing-out" of the objective function. For a large M, non-differentiable terms in the objective become "small" with respect to the complete sum of the norms, and the objective is almost smooth. This potentially explains why the nonlinear solvers find an optimal solution faster for large values of M, even though the problems seem to be more difficult.

Table 4.2 shows the computation times in seconds. Note that this comparison across solvers is not completely fair since we cannot expect MATLAB, an interpreted language, to be as fast as those natively compiled solvers. To somehow support this claim, we report that for a typical $(N, M) = (50, 100)$ run, 50% of the CPU time is spent on building the Hessian (involving a lot of data manipulation within MATLAB) and only 30% on actually computing the Newton step (solving a linear system with a single MATLAB command), while the latter operation should in principle be dominating the CPU cost.

dimension	D-IPM	PD-IPM	MINOS	SNOPT	KNITRO
$N = 2, M = 10$	0.14	0.21	0.02	0.02	0.02
$N = 2, M = 100$	0.37	0.51	0.03	0.02	0.02
$N = 2, M = 1000$	2.98	6.21	0.07	0.06	0.06
$N = 10, M = 10$	0.26	0.37	0.03	0.02	0.02
$N = 10, M = 50$	0.74	1.06	0.09	0.05	3.29
$N = 10, M = 100$	1.67	2.01	0.07	0.04	0.05
$N = 10, M = 500$	9.02	12.18	0.49	0.33	0.70
$N = 50, M = 10$	0.81	1.33	0.46	0.10	1.97
$N = 50, M = 50$	4.24	5.98	2.22	0.55	2.42
$N = 50, M = 100$	10.03	12.53	2.47	0.84	3.62
$N = 50, M = 200$	25.57	48.76	7.51	1.94	4.85

Table 4.2: CPU time in seconds used by each solver (averaged on 10 instances).

Looking at the computation times of both interior-point methods, we observe that the total computation time is higher for the primal-dual method by a factor of up to 2. This can be explained by the fact that for each update of the duality measure t we need to compute the primal-dual affine scaling direction (2.67), whose complexity is the same as the computation of a Newton direction for the centering problems (2.52).

When comparing to the computation times of the AMPL nonlinear solvers we see that the interior-point methods are not competitive. The main explanation for the poor performance of the interior-point scheme seems to be the large number of variables needed for the conic formulation: indeed, instead of working with a vector of N unknowns (and, accordingly, computing a $N \times N$ Hessian), our algorithm requires M additional epigraph variables τ_j and one additional $v_{j,i}$ variable for each of the $M \cdot N$ cones involved, for a total of $N + M + NM$ variables and the corresponding much enlarged Hessian. One could write (LOC_2) in a more compact way and remove the epigraph variables τ by replacing them directly in the conic constraints. But this modification would not have a huge impact because the total number of variables would still be $\mathcal{O}(NM)$ due to the presence of the v's.

Table 4.3 displays the percentage of problems for which the AMPL nonlinear solvers claimed to have found an optimal solution. The need for this table was prompted by the fact that, for a significant number of instances, the AMPL nonlinear solvers could not satisfy their stopping criterion (based on the norm of the gradient) and stopped either because a built-in iteration limit had been reached or insufficient progress after a certain number of iterations had been observed, reporting that the final iterate *might* not be optimal. For large M and N, this problematic behavior even seems to become the norm. However, in nearly all the cases, the solution provided was indeed optimal, meaning that all six requested digits of accuracy matched between the interior-point solution and its nonlinear counterpart. Nonetheless, we still classify these situations as unsuccessful because in general we do not know how close to the optimal solution we are and would like to have a guarantee to be within an ϵ distance of the optimal solution. Note that we have to be careful when comparing the solvers with respect to their reliability since different stopping criteria are used for the different solvers. Out path-following methods `D-IPM` and `PD-IPM` use as stopping criterion the error in terms of the objective value. This optimality error can be bounded in view of Theorem 2.4.15 and Theorem 2.5.18. The stopping criterion for MINOS is the size of the dual

dimension	D-IPM	PD-IPM	MINOS	SNOPT	KNITRO
$N = 2, M = 10$	100%	100%	100%	100%	100%
$N = 2, M = 100$	100%	100%	100%	100%	100%
$N = 2, M = 1000$	100%	100%	100%	100%	100%
$N = 10, M = 10$	100%	100%	90%	90%	90%
$N = 10, M = 50$	100%	100%	80%	60%	60%
$N = 10, M = 100$	100%	100%	100%	100%	100%
$N = 10, M = 500$	100%	100%	70%	40%	30%
$N = 50, M = 10$	100%	100%	0%	80%	60%
$N = 50, M = 50$	100%	100%	0%	10%	10%
$N = 50, M = 100$	100%	100%	0%	10%	10%

Table 4.3: Percentage of solutions for which optimality is guaranteed.

[1] Intel Pentium IV 3.00 GHz; MATLAB version 7.2 (R14)

solution; for SNOPT it is the relative error in the slackness conditions; and for KNITRO is the maximal violation of the KKT conditions. Thus, for all three AMPL solvers a solution is considered to be optimal if the optimality conditions are approximately satisfied. Unfortunately, this condition is not directly comparable to the stopping criterion we have used.

These failures to detect optimality are most probably due to the proximity/equality of the optimal solution to one of the fixed facilities and the (near) non-differentiability of the objective function that it causes. It is remarkable to observe that this non-differentiability has a significant impact on the practical behavior of the AMPL nonlinear solvers tested, even on relatively simple unconstrained problems with a finite number of problematic points. One can therefore conclude here that one of the main advantages of the interior-point solver lies in its insensitivity to these issues.

Comparison with Xue and Ye's algorithm

Xue and Ye [66] present an algorithm to solve the similar - but not identical - problem of minimizing a sum of p-norms, where all norms in the objective are defined by one single value of p and the decision variable x is scaled by a matrix A_j^T in each norm term,

$$\min_{u \in \mathbb{R}^N} \sum_{j=1}^{M} ||B_j - A_j^T u||_p, \tag{4.19}$$

with $B_j \in \mathbb{R}^d$ and $A_j^T \in \mathbb{R}^{d \times N}$, $j = 1, \ldots, M$. They propose a nonsymmetric primal-dual potential-reduction method that relies on the self-concordant barrier for the conic hull of the p-unit ball. Due to this construction the self-concordance parameter becomes relatively large, i.e. $200M(2d+1)$ for the description they chose in the computational examples.

We considered the problem (4.18)

$$\min_{u \in \mathbb{R}^N} \sum_{j=1}^{M} a_j ||u - C_j||_{p_j}$$

which is slightly more general in the sense that it does not require the norm terms to have identical parameters p, but on the other hand does not make use of scaling matrices A_j^T (however, incorporation of these matrices in our model would be trivial).

Although a direct comparison is not possible, problem (4.19) can be rewritten as

$$\min_{u \in \mathbb{R}^n} \sum_{j=1}^{M} ||B_j|| \cdot \left|\left| \frac{B_j}{||B_j||} - \frac{A_j^T}{||B_j||} u \right|\right|_p ,$$

with the constants in each norm term having components in the interval $[0, 1]$. Assuming that $d = N$ and $p_j = p, \forall j = 1, \ldots, M$, we can compare the complexity of both methods in that case. The self-concordance parameter of the barrier used by Xue and Ye in [66] is $200M(2N+1)$, while for our barrier it is only $3NM$, showing a clear advantage for our approach. The iteration complexity of the method used by Xue and Ye in their computational results is $\mathcal{O}(2M\sqrt{200M(2N+1)} \cdot \log(\frac{\max ||B_j||}{\epsilon} \cdot MN))$, and for our method is $\mathcal{O}(N\,M) \cdot \mathcal{O}\left(\log\left(\frac{N\,M}{\epsilon}\right)\right)$. The complexities are equivalent except for $M \gg N$, in which case our method has a better bound. Finally, the cost per iteration for Xue and Ye in [66] is $\mathcal{O}(MN^3)$, which is also the case for our method. Summarizing, both methods have comparable overall algorithmic complexity, although our method has a slight advantage when $M \gg N$.

We now look at the test case considered by Xue and Ye in [66] of finding the shortest network under a given N-Steiner topology, with $L = 10$. Reformulating this problem as a sum of p-norms problem yields the parameters $M = 2L - 3, d = 2, N = 2L - 4$. Xue and Ye consider several values of p, among which $p = 3$. With the algorithms proposed in their work they get a solution with an accuracy of $1.0e - 5$ in 33 iterations, whereas choosing for our formulation some random data in the same dimensions (omitting d), we get a solution with the same accuracy requirement in 31 iterations. This slight improvement is not too surprising because of the smaller self-concordance parameter of our barrier, although both iteration counts are much better than their corresponding pessimistic worst-case bounds.

Comparing CPU times is not possible since no computation times are reported by Xue and Ye [66]. Although the size of the linear systems to be solved at each iteration is smaller ($2M(d+1) + 2M + N = 152$) when compared to ours ($N(M+1) = 288$), the special block-structure and sparsity of our system have to be taken into account, which make it difficult to predict which method will be more efficient in practice.

To conclude this section, we observe that the main advantage of our approach seems to be its simplicity and versatility: a single barrier for the 3-dimensional power cone is all that is needed to derive a polynomial-time algorithm, while Xue and Ye [66] propose in their approach three different barriers for p-cones, to be chosen according to the value of the norm parameter p. Moreover, while the approach of Xue and Ye can in principle be applied to any problem involving p-cones such as (LOC_1), ours can be applied to any problem based on power cones, which encompass all problems with p-cones and many others, such as problems involving sums of p powers.

4.5.2 Geometric programming

We recall the definition of a geometric program in posynomial form (GP). Let $x \in \mathbb{R}^N$ be positive variables, $D^{(\text{pos})}_{i,j}$ and $e^{(\text{mon})}_j$ positive coefficients and $\mathbf{K}^{(\text{pos})}_{i,j} \in \mathbb{R}^N$, $i = 1, \dots, n_j$, $j = 0, \dots, M$ and $K^{(\text{mon})}_j \in \mathbb{R}^N$, $j = 1, \dots, M_{\text{mon}}$ real exponents. Then (GP) is given by

$$\begin{aligned} &\min_x \sum_{i=1}^{n_0} D^{(\text{pos})}_{i,0}\, x^{\mathbf{K}^{(\text{pos})}_{i,0}} \\ &\text{s.t.} \sum_{i=1}^{n_j} D^{(\text{pos})}_{i,j}\, x^{\mathbf{K}^{(\text{pos})}_{i,j}} \leq 1, \;\; j = 1, \dots, M, \\ &\qquad e^{(\text{mon})}_j\, x^{K^{(\text{mon})}_j} = 1, \;\; j = 1, \dots, M_{\text{mon}}. \end{aligned} \tag{GP}$$

We have shown in Section 4.4.2 that (GP) can be cast in dual conic form

$$\begin{aligned} &\max\ b^T y \\ &s.t.\ c - A^T y \in \mathcal{K}^* = \mathcal{K}_{\exp} \times \cdots \times \mathcal{K}_{\exp}, \\ &\quad A_f^T y = c_f. \end{aligned} \qquad (D-GP)$$

The variables are $y = (u, w, v) \in \mathbb{R}^{N+1+\sum_{j=0}^M n_j}$, where $w \in \mathbb{R}$ and $v \in \mathbb{R}^{\sum_{j=0}^M n_j}$ are artificial variables. The data A, A_f, c, c_f and b are suitably chosen and $\mathcal{K}^*$ is a direct product of $\sum_{j=0}^M n_j$ exponential cones $\mathcal{K}_{\exp}$. Using the 3-self-concordant barrier for $\mathcal{K}_{\exp}$, we obtain a ν-self-concordant barrier for the feasible set of $(D-GP)$, where $\nu = 3\sum_{j=0}^M n_j$.

Let us show again that the assumptions on page 66 are satisfied. One can verify that the matrix A in the conic formulation $(D-GP)$ is given by

$$A = \begin{bmatrix} \mathbf{K}_{\text{pos}} \cdot B_1 \\ 0 \cdots 0 \\ B_2 \end{bmatrix} \in \mathbb{R}^{(N+1+\sum_{j=0}^M n_j)\times(3\cdot\sum_{j=0}^M n_j)},$$

where

$$\begin{aligned} \mathbf{K}_{\text{pos}} &= \left[\mathbf{K}_{1,0}^{(\text{pos})} \cdots \mathbf{K}_{n_M,M}^{(\text{pos})}\right] \in \mathbb{R}^{N\times\sum_{j=0}^M n_j} \\ B_1 &= \texttt{blkdiag}_{\sum_{j=0}^M n_j}([-1,0,0]) \in \mathbb{R}^{(\sum_{j=0}^M n_j)\times(3\cdot\sum_{j=0}^M n_j)} \\ B_2 &= \texttt{blkdiag}_{\sum_{j=0}^M n_j}([0,-1,0]) \in \mathbb{R}^{(\sum_{j=0}^M n_j)\times(3\cdot\sum_{j=0}^M n_j)}. \end{aligned}$$

Since $\mathbf{K}_{\text{pos}}$ has full row rank we conclude that A is in fact rank deficient, its rank is equal to $N + \sum_{j=0}^M n_j$. However, the linear equalities are given by $A^T y = c_f$, with

$$A_f^T = \begin{bmatrix} E & B \\ G & 0 \end{bmatrix},$$

where

- $E = \begin{bmatrix} 0 & \cdots & 0 & -1 \\ & & & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{(M+1)\times(N+1)}$
- $B = \texttt{blkdiag}(\mathbf{1}_{n_j}) \in \mathbb{R}^{(M+1)\times(\sum_{j=0}^M n_j)}$, where $\mathbf{1}_{n_j}$ is a row vector of ones of size $n_j, j = 0, \ldots, M$
- $G = \begin{bmatrix} K^{(\text{mon})} \\ 0 \cdots 0 \end{bmatrix}^T \in \mathbb{R}^{M_{\text{mon}}\times(N+1)}$.

We see that the row $N+1$ in $[A, A_f]$ consists of zeros except for the first column of A_f where the component is -1. That means this row is linearly independent from all the other rows and we conclude $[A, A_f]$ has full row rank. Further, we see immediately that A_f has full column rank because both B^T and $K^{(\mathrm{mon})}$ have full column rank.

To see the last assumption, let $y = (u, w, v)$ such that $c - A^T y \in \operatorname{int} \mathcal{K}^*$ and $A_f^T y = c_f$. Let $(\Delta u, \Delta w, \Delta v) \neq 0$ be any direction.

If $\Delta w \neq 0$ or $\Delta v \neq 0$ then it is always possible to find a step size γ such that $w + \gamma \Delta w < 0$ or $v + \gamma \Delta v \not> 0$, which is not possible. Let therefore $\Delta w = 0$ and $\Delta v = 0$, and assume $\Delta u \neq 0$ (otherwise $(\Delta u, \Delta w, \Delta v) = 0$). Since $\mathbf{K}_{\mathrm{pos}}$ has full row rank, it cannot contain any 0-rows, which means there exists a column $\mathbf{K}_{i,j}^{(\mathrm{pos})}$ such that $\Delta u^T \mathbf{K}_{i,j}^{(\mathrm{pos})} \neq 0$. It is clear that we can find a step size γ such that

$$(u + \gamma \Delta u)^T \mathbf{K}_{i,j}^{(\mathrm{pos})} + C_{i,j} > \max\{\log(w), 0\},$$

which violates one of the inequality constraints.

This means for any strictly feasible point y and any direction $\Delta y \neq 0$ we cannot extend along this direction towards $+\infty$ and $-\infty$, so the feasible region of $(D - GP)$ does not contain straight lines and the Newton directions are defined at any feasible point y.

In order to test our algorithm we consider a family of random GPs generated by the script `mkgp`, a Matlab function included in `gpcvx`[5] that has originally been written by Lieven Vandenberghe and later modified by Kwangmoo Koh. The parameters are N (the number of original variables), M (the number of posynomial constraints), M_{mon} (the number of monomial constraints), $n_j,\ j = 0, \ldots, M$ (the number of monomial terms in each posynomial).

We test on random instances with $n_j = 5, j = 0, \ldots, M$, $M_{\mathrm{mon}} = 5$ for different values of N and M. We are comparing the dual path-following method (D-IPM), the nonsymmetric predictor-corrector method (PD-IPM) and `gpcvx`, a dedicated GP solver by Koh et al ([36]). The first two methods solve the conic reformulation $(D - GP)$, while `gpcvx` works directly on the convex reformulation (GP); to be precise, it uses the logarithmic transformation of the objective and the constraints. The constraints of the form

$$\sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}_{i,j} + C_{i,j}\right) \leq 1,$$

are reformulated as

$$\log\left(\sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}_{i,j} + C_{i,j}\right)\right) \leq \log(1) = 0, \tag{4.20}$$

for $j = 1, \ldots, M$. This convex reformulation of (GP_0) admits a dual problem involving entropy functions (for details, see [36]). The authors of GPCVX propose a primal-dual interior-point method which uses barriers of the following form

$$F_j(u) = -\log\left(-\log\left(\sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}_{i,j} + C_{i,j}\right)\right)\right).$$

[5] See `http://www.stanford.edu/~boyd/ggplab/gpcvx.pdf`

These barriers are *not* known to be self-concordant for its domain (unless $n_j = 1$). Therefore, the proposed interior-point method does not exhibit a guaranteed polynomial complexity.

Table 4.4 illustrates the number of iterations for solving (GP) up to accuracy $\epsilon = 10^{-6}$ when using the three above-mentioned methods. The number of iterations of the primal-dual method is lower than for the dual method. However, we also note that the iteration count is much higher as compared to `gpcvx`. This could be explained by the lifting to the higher-dimensional problem, where for each monomial in each posynomial one cone $\mathcal{K}_{\text{exp}}$ is introduced. `gpcvx` on the other hand, does not use a self-concordant barrier, but instead it works directly on the logarithmic transformation of the convex reformulation (GP_0).

dimension	D-IPM	PD-IPM	GPCVX
$N = 50, M = 50$	92	66	35
$N = 50, M = 100$	237	82	36
$N = 50, M = 150$	158	72	36
$N = 100, M = 100$	133	74	36
$N = 100, M = 200$	172	79	38
$N = 100, M = 300$	127	68	38
$N = 100, M = 400$	98	67	37
$N = 100, M = 500$	246	86	40
$N = 200, M = 200$	348	132	38
$N = 200, M = 400$	292	98	39
$N = 200, M = 600$	105	92	43
$N = 200, M = 800$	211	105	43

Table 4.4: Number of iterations used by each solver.

This effect is even more striking when looking at Table 4.5 which displays the computation time. The computation time of our dual and primal-dual method is significantly higher as compared to `gpcvx`. This observation can be explained by investigating the numerical cost per iteration. Indeed, `gpcvx` computes Newton directions in terms of the original variables u which means linear systems of size N have to be solved in each iteration. In the decomposed problem (GP_2) the number of variables in $N + 1 + \sum_{j=0}^{M} n_j = N + 1 + 5(M + 1)$, which is essentially the size of the linear systems to be solved in each iteration. We see in Table 4.5 that when using the decomposing technique the cost of each iteration does depend on the number of posynomial terms M, while the computation time of `gpcvx` is independent of the M. This effect can be observed .

We also observe that the computation time of the primal-dual method is higher as compared to the dual method, even though the number of iterations is lower. This can be explained by the fact that after each dual centering we compute the primal-dual affine-scaling direction (2.67) whose cost is comparable to computing one Newton direction for the centering phase.

dimension	D-IPM	PD-IPM	GPCVX
$N = 50, M = 50$	1.19	2.33	0.27
$N = 50, M = 100$	3.79	3.89	0.20
$N = 50, M = 150$	3.18	4.14	0.23
$N = 100, M = 100$	3.17	4.32	0.36
$N = 100, M = 200$	4.97	5.46	0.41
$N = 100, M = 300$	4.46	6.01	0.45
$N = 100, M = 400$	4.03	7.25	0.48
$N = 100, M = 500$	13.15	10.10	0.57
$N = 200, M = 200$	13.61	11.81	0.88
$N = 200, M = 400$	15.01	10.74	0.89
$N = 200, M = 600$	6.49	14.22	0.99
$N = 200, M = 800$	14.37	19.75	1.03

Table 4.5: CPU time (in seconds) used by each solver.

4.5.3 A problem involving mixed powers

In Section 4.5.1 we analyzed the problem class of generalized location problems and compared the performance of our dual path-following method and primal-dual predictor-corrector method to three nonlinear solvers (MINOS, SNOPT and KNITRO). It turned out that for small instances all solvers are more or less comparable, while for larger problems the three nonlinear solvers are getting more and more unreliable in terms of the number of problems that are solved up to optimality with optimality certificate.

In this section we consider a problem class where even for tiny instances the nonlinear solvers have difficulties with solving the problems. Let us recall the convex set $\mathcal{C}_p$ from Section 4.1.4, which is defined over mixed p-powers, i.e.

$$\mathcal{C}_p = \left\{ (x,t) : \sum_{i=1}^{N} |x_i|^{p_i} \leq t^{p_0},\ t \geq 0 \right\}$$

where $1 \leq p_0 \leq \min_{i=1,\ldots,N} p_i$. We have seen in Section 4.1.4 that $\mathcal{C}_p$ is α-representable and therefore we can optimize over that set.

For an arbitrary vector $d \in \mathbb{R}^N$ let us define the convex problem

$$\begin{aligned} \min_{x,t} \quad & d^T x + t \\ \text{s.t.} \quad & (x,t) \in \mathcal{C}_p \\ & t \leq 1. \end{aligned} \tag{4.21}$$

We see that the feasible set of (4.21) is defined over one single nonlinear constraint in dimension $N + 1$. Additionally, we introduced one linear inequality constraint to bound the feasible set and to make sure that the problem is solvable. Using the conic decomposition of $\mathcal{C}_p$ from Section 4.1.4, we arrive at the following conic reformulation of (4.21)

$$\begin{aligned}
&\min \; d^T x + \sum_{i=1}^{N} w_i \\
&\text{s.t. } (v_i, 1, x_i) \in \mathcal{K}_{\alpha_i}, \; \alpha_i = \frac{p_0}{p_i}, \; i = 1, \ldots, N, \\
&\qquad \left(w_i, \sum_{i=1}^{N} w_i, v_i \right) \in \mathcal{K}_{\alpha_0}, \; \alpha_0 = \frac{1}{p_0}, \; i = 1, \ldots, N, \\
&\qquad 1 - \sum_{i=1}^{N} w_i \in \mathbb{R}_+.
\end{aligned} \tag{4.22}$$

We see that we have reformulated (4.21) by using $2N$ power cone constraints and by introducing $2N$ artificial variables v and w. On the other hand, we have replaced t by $\sum_{i=1}^{N} w_i$.

In the numerical tests we generated instances of (4.21) for different values of N, and random data d and p such that $1 \leq p_0 \leq \min_{i=1,\ldots,N} p_i$. Our goal is not to reach the computational limit in terms of the problem size. We aim to illustrate that even for small and rather well-behaved problems the nonlinear solvers might fail. This is why we are comparing the number of iterations and the reliability of the dual path-following method to the nonlinear solvers MINOS 5.5 [6], SNOPT 6.1-1[7] and KNITRO 6.0.0[8]. All three solvers are given the original problem (4.21) to solve. Our interior-point solver is solving the dual conic reformulation of (4.22).

In Table 4.6 we list the average number of iterations to solve the random instances of (4.21). The number of iterations of the dual interior-point method is increasing only mildly with the problem size. On the other hand, MINOS and SNOPT display a significant increase in the number of iterations. For $N = 100$ it is roughly 10 times the number of iterations needed by the dual interior-point method. KNITRO shows an inexplicable behavior. For $N = 2$ it needs a rather large number of iterations, which gets even worse for increasing problem size. However, for $N = 100$ the number of iterations is becoming rather low, even though the problem seems to be more difficult.

dimension	D-IPM	MINOS	SNOPT	KNITRO
$N = 2$	27.2	33.4	39.5	336.6
$N = 5$	23.3	56.9	46.3	1493.4
$N = 10$	25.3	61.2	84.3	3200.3
$N = 20$	36.2	95.8	94.7	2137
$N = 50$	36.0	174.4	358.8	1021.2
$N = 100$	37.1	247.3	399	25

Table 4.6: Number of iterations for each solver (averaged on 10 instances).

[6] http://www.sbsi-sol-optimize.com/asp/sol_product _minos.htm

[7] http://www.sbsi-sol-optimize.com/asp/sol_product_snopt.htm

[8] http://www.ziena.com/knitro.htm

Table 4.7 shows the reliability (in terms of the average number of problems that could be solved) of the four solvers tested. We remark here that the dual interior-point method could solve all the problems up to optimality, while the three nonlinear solvers have difficulties with solving the problems. MINOS was solving the problems rather robustly, even though the number of iterations were increasing significantly with the problem size. SNOPT was completely failing on the larger instances (even though it claimed to have found an optimal solution, which was clearly worse than the true optimal solution). KNITRO, on the other hand was surprisingly failing on the smaller instances, it stopped after a large number of iterations at solutions that were far from optimal.

dimension	D-IPM	MINOS	SNOPT	KNITRO
$N = 2$	100%	80%	20%	70%
$N = 5$	100%	90%	50%	80%
$N = 10$	100%	100%	30%	70%
$N = 20$	100%	100%	60%	90%
$N = 50$	100%	100%	0%	90%
$N = 100$	100%	100%	0%	100%

Table 4.7: Percentage of solutions for which optimality is guaranteed

CHAPTER 5

Partial minimization

We have shown in earlier chapters that interior-point methods are extremely powerful and reliable for solving convex optimization problems. We have recalled for example that a polynomial complexity can be established in terms of the number of iterations for finding a point in close proximity to an exact optimal solution. Moreover, the theoretical complexity bound is often pessimistic in that the number of iterations to solve the problem increases only mildly when the problem size increases (for details see Section 4.5).

On the other hand interior-point methods have at least three major drawbacks. First, in order to profit from the polynomial complexity safeguard, one needs an explicit self-concordant barrier for the feasible set (and possibly also for the epigraph of the convex objective). To find such a barrier is in general a difficult task because it involves a uniform bounding of the third directional derivative of the barrier by an appropriate power of the second directional derivative. Second, due to their nature as second-order methods, interior-point methods have the practical disadvantages of relatively high memory usage. In each iteration one needs to compute and store the gradient and Hessian of some convex function. Finally, each iteration requires a high computational effort. In order to compute the search directions, one has to solve a linear system of equations, whose size is essentially the number of variables. For dense problems with many variables these tasks are intractable. We have observed those phenomena in Section 4.5, where, although the number of iterations of our interior-point code was competitive with or even better than most of the tested solvers for nonlinear problems, the cost of one single iteration was increasing dramatically when the problem size increased.

However, there is hope for improvement, because often it is necessary to lift the feasible set of an optimization problem to a higher-dimensional space in order to have a self-concordant barrier available. That means the problem we actually want to solve might not be large per se, but has to be embedded into a higher-dimensional formulation in order to use the interior-point machinery.

Consider for example the convex constraint

$$\sum_{i=1}^{n} \exp(u_i) \leq 1, \tag{5.1}$$

for which no explicit self-concordant barrier, expressed only in terms of u, is known. However, it is clear that (5.1) can be reformulated into

$$\begin{aligned} \exp(u_i) &\leq v_i, \ i = 1, \dots, n \\ \sum_{i=1}^{n} v_i &= 1 \end{aligned} \tag{5.2}$$

by introducing n additional variables v. Note that (5.2) can be handled in an interior-point framework since a self-concordant barrier for the epigraph of the exponential function is known. On the other hand we see that we have doubled the number of variables from n to $2n$. This means the more compact but less handy constraint (5.1) is replaced by the augmented ($2n$ instead of n variables) but simpler constraint (5.2) with the benefit of having a self-concordant barrier at hand. In general, a doubling of the number of variables means that the memory storage increases roughly by a factor of 4 and the computational cost of one iteration roughly by a factor of 8 (using e.g. a Cholesky factorization to solve the linear system). Even if - in the particular example shown above - the storage and computational cost will certainly be more favorable than in the general case, there will still be some non-negligible increase of the storage and numerical cost observable.

In this chapter we address the issue of variable reduction. We consider convex problems that can be reformulated into *partially sparse* problems with explicit self-concordant barrier for the feasible set. Based on results by Nesterov ([50]) we use this partial sparsity to derive an implicit barrier for the feasible set of the original convex problem. We will show how this technique can be embedded into a path-following interior-point method, even if the partial minimization subproblem cannot be solved analytically but only approximately via an iterative scheme. We show that polynomial complexity can be maintained and that in some cases a substantial improvement in the computational effort can be achieved. Moreover, the proposed concept gives rise to parallelization in practical implementations. Indeed, we see that the decomposed formulation (5.2) is partially separable, in that the variables v_i only appear in one inquality constraint at a time. In the context of interior-point methods this means that the minimization of a barrier function F with respect to the artificial variables v can be done in parallel, which will result in faster computations.

In Section 5.1 we derive an extended formulation that naturally arises whenever we deal with convex problems for whose feasible set we do not know a self-concordant barrier. The concept of partial minimization is presented in Section 5.2. We show how it can be embedded into an interior-point method and present its complexity result. In Section 5.3 we consider the case where the partial minimization subproblem cannot be solved analytically. We show how to obtain an approximate partial minimizer and how the outer algorithm can be adapted so that the global complexity bound from the previous section is essentially maintained. Finally, in Section 5.4 we consider two classes of optimization problems (discussed earlier in Section 4.4) which are well-suited for the application of the partial minimization technique. We present numerical results for interior-point methods with and without approximate partial minimization.

5.1 Lifting

In this section we are going to formalize the situation that we have encountered in the example above, where the sum-of-exponential expression has been decomposed into several simple expressions which are linked by one linear equality.

We consider a general convex optimization problem of the following form

$$\begin{aligned} &\min_z \; a^T z \\ &\quad z \in \tilde{\mathcal{C}} \subset \mathbb{R}^{n_1}, \end{aligned} \tag{5.3}$$

where $\tilde{\mathcal{C}}$ is a closed convex set which is not necessarily full-dimensional. We have shown in Chapter 2 that in order to use interior-point methods to solve (5.3), we need to have available a self-concordant barrier for the feasible set, which implicitly assumes that $\tilde{\mathcal{C}}$ has interior-points (i.e. it has to be full-dimensional). If $\tilde{\mathcal{C}}$ is not full-dimensional, then it can be written as

$$\tilde{\mathcal{C}} = \mathcal{C} \bigcap \bar{\mathcal{L}},$$

where $\mathcal{C} \subset \mathbb{R}^{n_1}$ is full-dimensional and $\bar{\mathcal{L}} = \{z : Gz = g\} \subset \mathbb{R}^{n_1}$ for some matrix G with full row rank. In that notation (5.3) becomes

$$\begin{aligned} &\min_z \; a^T z \\ &\quad z \in \mathcal{C}, \\ &\quad Gz = g. \end{aligned} \tag{5.4}$$

In Chapter 2 we have shown that if we have access to the value, the gradient and the Hessian of a self-concordant barrier for $\mathcal{C}$, then (5.4) can be solved in polynomial time. If we cannot access this information of a self-concordant barrier for $\mathcal{C}$, then it still might be possible to solve (5.4), provided that $\mathcal{C}$ can be expressed as a projection of some higher-dimensional closed convex set $\tilde{\mathcal{Q}} \subset \mathbb{R}^{n_1+n_2}$ onto the z-variables. Thus, we need that $z \in \mathcal{C}$ if and only if $\exists\, v \in \mathbb{R}^{n_2}$ such that

$$(z, v) \in \tilde{\mathcal{Q}},$$

where $\tilde{\mathcal{Q}}$ is not necessarily full-dimensional. If indeed $\tilde{\mathcal{Q}}$ does not contain interior points, then we can write it as

$$\tilde{\mathcal{Q}} = \mathcal{Q} \bigcap \mathcal{L}$$

with some full-dimensional set $\mathcal{Q} \subset \mathbb{R}^{n_1+n_2}$ and $\mathcal{L} = \{(z, v) : Ez + Bv = d\}$, where $B \in \mathbb{R}^{m_1,n_2}$, $E \in \mathbb{R}^{m_1,n_1}$ and $d \in \mathbb{R}^{m_1}$. We can assume without loss of generality that B has full row rank because $\mathcal{C}$ is full-dimensional. Indeed, if B does contain linearly dependent rows, then we can reformulate the system

$$Ez + Bv = d$$

so that it contains a linear constraint only involving z (and not v), which means that $\mathcal{C}$ cannot have full dimension. Summarizing, if $\mathcal{C}$ admits a lifting as described above, then (5.3) can be solved by the extended formulation

$$\begin{aligned} \min_{z,v} \; & a^T z \\ & (z,v) \in \mathcal{Q}, \\ & Ez + Bv = d, \\ & Gz = g, \end{aligned} \tag{5.5}$$

where B and G have full row rank and $\mathcal{Q}$ is full-dimensional.

If we have a self-concordant barrier available for $\mathcal{Q}$, then (5.5) can be solved in polynomial time and the z-component of the optimal solution is an optimal solution for (5.4). The drawback of this lifting procedure is the introduction of artificial variables v for the sake of having a formulation whose feasible set admits a self-concordant barrier. The technique of partial minimization targets at overcoming the above-mentioned disadvantage. Based on the extended formulation (5.5) with the self-concordant barrier F for $\mathcal{Q}$ we want to derive a self-concordant barrier for the lower-dimensional set $\mathcal{C}$, or at least to approximate such a barrier.

Let us summarize the basic assumptions that we make for the partial minimization framework:

Assumption 4 *[Partial Minimization]*

1. *$\mathcal{C} \subset \mathbb{R}^{n_1}$ and $\mathcal{Q} \subset \mathbb{R}^{n_1+n_2}$ are full-dimensional closed convex sets,*
2. *$B \in \mathbb{R}^{m_1,n_2}$ and $G \in \mathbb{R}^{m_2,n_1}$ have full row rank, $E \in \mathbb{R}^{m_1,n_1}$, $g \in \mathbb{R}^{m_2}$, $d \in \mathbb{R}^{m_1}$,*
3. *let $z \in \mathcal{C} \subset \mathbb{R}^{n_1}$ if and only if $\exists\, v$ such that $(z,v) \in \mathcal{Q} \bigcap \mathcal{L}$,*
4. *$\mathcal{C}$ does not contain straight lines,*
5. *for any $\bar{z} \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$ we have $\mathcal{Q} \bigcap \mathcal{L} \bigcap \{(z,v) : z = \bar{z}\}$ is bounded,*
6. *F is a ν-self-concordant barrier for $\mathcal{Q}$.*

The above assumptions might seem somewhat restrictive. However, we argue (and we will demonstrate later on concrete examples) that these assumptions arise naturally. The full-dimensionality of $\mathcal{C}$ and $\mathcal{Q}$ is necessary for the interior-point framework. If they are not full-dimensional, then it is possible to write them as the intersection of a full-dimensional set with an affine subspace, defined by a matrix with full row rank.

The projection property naturally arises whenever $\mathcal{C}$ is given by convex constraints that are given by combinations (e.g. sums) of elementary convex functions.

We will show later examples of such projections. The fact that $\mathcal{C}$ does not contain straight lines ensures that the Newton directions for the implicit barrier (that we will define later) are defined. The assumption $\mathcal{Q}\bigcap\mathcal{L}\bigcap\{(z,v) : z = \bar{z}\}$ being bounded for a fixed $\bar{z} \in \mathcal{C}\bigcap\bar{\mathcal{L}}$ will guarantee that the partial minimization subproblems do have unique solutions.

5.2 Exact partial minimization

Partial minimization denotes the process of minimizing a self-concordant barrier F with respect to *some* of its variables. In this section we show that partial minimization preserves the self-concordance property. This means that if the partial minimization problem can be solved efficiently, then it can be used as a means to reduce the dimension while preserving polynomial complexity of solving the actual problem.

5.2.1 Partial minimization theorem

Let $z \in \operatorname{int}\mathcal{C}\bigcap\bar{\mathcal{L}}$. The partial minimization problem is given by

$$\begin{aligned} \min_v\ & F(z,v) \\ & Ez + Bv = d. \end{aligned} \tag{PM(z)}$$

We have assumed that for any *fixed* z the set $\mathcal{Q}\bigcap\mathcal{L}$ is bounded. Since F is a barrier for its domain $\operatorname{int}\mathcal{Q}$, and any intersection of $\mathcal{Q}$ with an affine subspace, there exists a unique solution for (PM(z)). Let us denote this *partial minimizer* by $v(z)$. Then we can define the implicit function

$$\varphi(z) = F(z, v(z)),$$

which we call the *partially minimized barrier.* It is clear that $\operatorname{dom}\varphi = \operatorname{int}\mathcal{C}$. Indeed, if $z \in \operatorname{int}\mathcal{C}$, then there exists by assumption a v such that $(z,v) \in \operatorname{int}\mathcal{Q}$, which means that $F(z,v) < \infty$. Thus, $\varphi(z) < \infty$, i.e. $z \in \operatorname{dom}\varphi$. Reversely, let $z \in \operatorname{dom}\varphi$. Then there exists a point $v(z)$ such that $(z, v(z)) \in \operatorname{dom} F = \operatorname{int}\mathcal{Q}$ and $E\,z + B\,v(z) = d$. By assumption we have then $z \in \operatorname{int}\mathcal{C}$.

Note that $v(z)$ is characterized by the optimality conditions: $\exists\ \lambda_v(z)$ such that

$$\begin{aligned} F_v'(z, v(z)) &= -B^T\,\lambda_v(z) \\ B\,v(z) &= d - Ez. \end{aligned} \tag{5.6}$$

When we derive the optimality conditions (5.6) with respect to z, we get

$$\begin{aligned} F_{vv}^{''}(z,v(z))\,v'(z) + F_{vz}^{''}(z,v(z)) &= -B^T\,\lambda_v'(z) \\ B\,v'(z) &= -E, \end{aligned}$$

which can be written in the more compact form

$$\begin{bmatrix} F_{vv}^{''}(z,v(z)) & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} v'(z) \\ \lambda_v'(z) \end{bmatrix} = \begin{bmatrix} -F_{vz}^{''}(z,v(z)) \\ -E \end{bmatrix}. \tag{5.7}$$

The solutions $v'(z)$ and $\lambda_v'(z)$ are the Jacobians of the partial minimizer $v(z)$ and the optimal dual multipliers $\lambda_v(z)$ with respect to z. Let us compute now the derivatives of the implicit barrier φ. When applying the chain rule and the optimality conditions (5.6) and (5.7), we get

$$\begin{aligned}\nabla\varphi(z) &= F_z'(z,v(z)) + v'(z)^T \underbrace{F_v'(z,v(z))}_{\overset{(5.6)}{=}-B^T\lambda_v(z)} \\ &= F_z'(z,v(z)) - \underbrace{v'(z)^T B^T}_{\overset{(5.7)}{=}-E^T}\lambda_v(z) \\ &= F_z'(z,v(z)) + E^T\lambda_v(z). \end{aligned} \tag{5.8}$$

It follows that the Hessian becomes

$$\nabla^2\varphi(z) = F_{zz}''(z,v(z)) + F_{zv}''(z,v(z))\,v'(z) + E^T\lambda_v'(z). \tag{5.9}$$

The following theorem is the main result of this section. It states that minimizing a self-concordant barrier with respect to some of its variables preserves the self-concordance property.

Theorem 5.2.1. *Let the Assumptions 4 be satisfied. Then*

$$\varphi(z) = \min_v\{F(z,v) : Bv = d - Ez\}$$

is a nondegenerate ν-self-concordant barrier for $\mathcal{C}$.

In [50, Theorem 3] Nesterov provides a similar result for the special case where $\mathcal{Q}$ does not contain a straight line, something we do not assume here, and the right-hand side of the linear equalities do not depend on z (i.e. $E = 0$). Theorem 5.2.1 treats therefore a more general case and we have to prove it for our setting. The outline of the proof, however, is strongly related to the one in [50].

As a first step, we need the following technical results.

Lemma 5.2.2. *Let*

$$\bar{A} = \begin{bmatrix} A_{11} & A_{12}^T \\ A_{12} & A_{22} \end{bmatrix} \in \mathbb{R}^{n_1+n_2,n_1+n_2}$$

be symmetric and positive semidefinite and A_{22} positive definite. $\bar{A}$ defines a quadratic form in $y = (z,v)$:

$$\langle \bar{A}y, y\rangle = \langle A_{11}z, z\rangle + 2\langle A_{12}z, v\rangle + \langle A_{22}v, v\rangle. \tag{5.10}$$

Let $B \in \mathbb{R}^{m_1,n_2}$ be a matrix with full row rank, $E \in \mathbb{R}^{m_1,n_1}$. Then the matrix P of the quadratic form

$$\langle Pz, z\rangle = \min_v\{\langle \bar{A}y, y\rangle : Bv = -Ez\} \tag{5.11}$$

is given by

$$P = P_1 + P_2,$$

where

$$\begin{aligned}P_1 &= A_{11} - A_{12}^T A_{22}^{-1} A_{12}\\ P_2 &= (-E + BA_{22}^{-1}A_{12})^T (BA_{22}^{-1}B^T)^{-1}(-E + BA_{22}^{-1}A_{12}).\end{aligned}$$

Proof. The optimality conditions for (5.11) are

$$\begin{aligned}A_{22}v + A_{12}z &= B^T\lambda\\ Bv &= -Ez.\end{aligned}\tag{5.12}$$

Since A_{22} is positive definite, we get

$$v = A_{22}^{-1} \cdot (B^T\lambda - A_{12}z),$$

and substituting the above expression for v in the second equation of (5.12) yields

$$BA_{22}^{-1}B^T\,\lambda = (-E + BA_{22}^{-1}A_{12})\,z.$$

B has full row rank, which implies that the matrix on the left-hand side is positive definite. Therefore, we get

$$\lambda = \underbrace{(BA_{22}^{-1}B^T)^{-1}\,[-E + BA_{22}^{-1}A_{12}]}_{=:P_3}\,z$$

and replacing this expression in the above formula for v, we obtain

$$v = \underbrace{A_{22}^{-1}[B^T(BA_{22}^{-1}B^T)^{-1}\,(-E + BA_{22}^{-1}A_{12}) - A_{12}]}_{=A_{22}^{-1}(B^TP_3 - A_{12})}\,z.$$

Let us plug now this expression into (5.10):

$$\begin{aligned}\langle \bar{A}y, y\rangle =&\langle A_{11}z, z\rangle + 2\langle A_{12}z, A_{22}^{-1}(B^TP_3 - A_{12})\,z\rangle\\ &+ \langle (B^TP_3 - A_{12})z, A_{22}^{-1}(B^TP_3 - A_{12})\,z\rangle,\end{aligned}$$

which is a quadratic form in z. Let us simplify the corresponding matrix P such that $\langle \bar{A}y, y\rangle = \langle Pz, z\rangle$. We get

$$\begin{aligned}P =&A_{11} + 2A_{12}^TA_{22}^{-1}B^TP_3 - 2A_{12}^TA_{22}^{-1}A_{12}\\ &+ \underbrace{P_3^TBA_{22}^{-1}B^TP_3}_{=P_2} - P_3^TBA_{22}^{-1}A_{12} - A_{12}^TA_{22}^{-1}B^TP_3 + A_{12}^TA_{22}^{-1}A_{12}\\ =&\underbrace{A_{11} - A_{12}^TA_{22}^{-1}A_{12}}_{P_1} + P_2 + \underbrace{A_{12}^TA_{22}^{-1}B^TP_3}_{=:M} - \underbrace{P_3^TBA_{22}^{-1}A_{12}}_{=M^T}\\ =&P_1 + P_2 + M - M^T.\end{aligned}$$

It remains to note that $M - M^T$ is skew-symmetric, which implies

$$\langle Pz, z\rangle = \langle (P_1 + P_2)z, z\rangle + \underbrace{\langle (-M + M^T)z, z\rangle}_{=0} = \langle (P_1 + P_2)z, z\rangle$$

□

Remark 5.2.3. *We see that P_1 is the Schur complement of $\bar{A}$ with respect to the block A_{22}. Furthermore, we see that the Schur complement of the augmented matrix*

$$\begin{bmatrix} A_{11} & A_{12}^T & E^T \\ A_{12} & A_{22} & B^T \\ E & B & 0 \end{bmatrix}$$

with respect to the block A_{22} is

$$M_1 = \begin{bmatrix} P_1 & E^T - A_{12}^T A_{22}^{-1} B^T \\ E - BA_{22}^{-1} A_{12} & -BA_{22}^{-1} B^T \end{bmatrix}.$$

Since B has full row rank, we get as the Schur complement of M_1 with respect to the block $-BA_{22}^{-1}B^T$:

$$P = P_1 + P_2.$$

We have seen that (5.9) gives the Hessian of the implicit barrier φ, assuming that we have computed the partial minimizer $v(z)$ and the Jacobians $v'(z)$ and $\lambda_v'(z)$. The following result provides an interpretation of the matrix $H(z,v)$ that we obtain from (5.9) *without* assuming that v is the partial minimizer for z.

Corollary 5.2.4. *Let $B \in \mathbb{R}^{m_1,n_2}$ be a matrix with full row rank and $E \in \mathbb{R}^{m_1,n_1}$. Let $y = (z,v) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$ and $\Delta y = (\Delta z, \Delta v)$ a feasible direction such that $B\Delta v + E\Delta z = 0$. Let the matrices $J(z,v)$ and $L(z,v)$ be the solution of*

$$\begin{bmatrix} F_{vv}^{''}(z,v) & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} J(z,v) \\ L(z,v) \end{bmatrix} = \begin{bmatrix} -F_{vz}^{''}(z,v) \\ -E \end{bmatrix} \tag{5.13}$$

and define $H(z,v)as$

$$H(z,v) = F_{zz}^{''}(z,v) + F_{zv}^{''}(z,v)\, J(z,v) + E^T\, L(z,v).$$

Then we have

$$\langle H(z,v)\Delta z, \Delta z\rangle = \min_{\Delta v}\{\langle \nabla^2 F(y)\Delta y, \Delta y\rangle : B\Delta v = -E\Delta z\}.$$

Proof. We simply apply Lemma 5.2.2 with $A_{11} = F_{zz}^{''}(z,v)$, $A_{12} = F_{vz}^{''}(z,v)$ and $A_{22} = F_{vv}^{''}(z,v)$. Note that Assumption 4 implies that $F_{vv}^{''}(z,v)$ is positive definite. Let us denote $J = J(z,v)$, $L = L(z,v)$ and $H = H(z,v)$. We have to show that

$$P_1 + P_2 = H.$$

Recall that, using the above notation, we have

$$H = A_{11} + A_{12}^T\, J + E^T\, L \tag{5.14}$$

where J and L are the solutions of (5.13) which becomes in the above notation

$$\begin{bmatrix} A_{22} & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} J \\ L \end{bmatrix} = \begin{bmatrix} -A_{12} \\ -E \end{bmatrix}.$$

Since A_{22} is positive definite, we get

$$J = -A_{22}^{-1}(B^T L + A_{12})$$

and substituting the above expression for J in the second equation of the system yields

$$BA_{22}^{-1}B^T L = -BA_{22}^{-1}A_{12} + E.$$

B has full row rank, which implies that the matrix on the left-hand side is positive definite. Therefore, we get

$$L = -(BA_{22}^{-1}B^T)^{-1} \cdot [-E + BA_{22}^{-1}A_{12}] = -P_3, \tag{5.15}$$

where P_3 is defined as in the proof of Lemma 5.2.2. If we replace the expression for L in the above formula for J, we obtain

$$J = A_{22}^{-1}(B^T P_3 - A_{12}). \tag{5.16}$$

Plugging (5.15) and (5.16) into (5.14), we get

$$\begin{aligned} H &= A_{11} - A_{12}^T A_{22}^{-2} A_{12} + A_{12}^T A_{22}^{-1} B^T P_3 - E^T P_3 \\ &= P_1 + (-E + BA_{22}^{-1}A_{12})^T P_3 \\ &= P_1 + P_2. \end{aligned}$$

□

Corollary 5.2.5. *Let $z \in \operatorname{int}\mathcal{C}$ with partial minimizer $v(z)$. Denote $y = (z, v(z))$. Then it holds for any direction $\Delta z \in \mathbb{R}^{n_1}$*

$$\Delta z^T \nabla^2 \varphi(z) \Delta z = \min_{\Delta v}\{\Delta y^T \nabla^2 F(y) \Delta y : B\Delta v = -E\Delta z\}.$$

Proof. The result follows directly from Corollary 5.2.4 and the fact that $v = v(z)$ means that (5.7) is the same system as (5.13). It follows $H(z, v(z)) = v'(z)$, $L(z, v(z)) = \lambda_v'(z)$ and consequently $H(z, v(z)) = \nabla^2\varphi(z)$.

□

We have now all the tools ready to prove Theorem 5.2.1.

Proof of Theorem 5.2.1. Let $z_0 \in \operatorname{int}\mathcal{C}$ with partial minimizer $v(z_0)$, denote $y_0 = (z_0, v(z_0))$. Let $\Delta z \in \mathbb{R}^{n_1}$ such that $\bar{z} = z_0 + \Delta z \in \operatorname{int}\mathcal{C}$. Let $\bar{v}$ such that $\bar{y} = (\bar{z}, \bar{v}) \in \operatorname{int}\mathcal{Q} \bigcap \mathcal{L}$, which always exists by assumption. Finally, denote $\Delta v = \bar{v} - v(z_0)$ and $\Delta y = (\Delta z, \Delta v)$.

Because of Theorem 2.2.15 and equations (5.8) and (5.6) we have

$$\begin{aligned} F(\bar{y}) &\geq F(y_0) + \nabla F(y_0)^T \Delta y + \omega(||\Delta y||_{\nabla^2 F(y_0)}) \\ &= \varphi(z_0) + F_z'(y_0)^T \Delta z + F_v'(y_0)^T \Delta v + \omega(||\Delta y||_{\nabla^2 F(y_0)}) \\ &= \varphi(z_0) + (\nabla\varphi(z_0) - E^T \lambda_v(z_0))^T \Delta z - \lambda_v(z_0)^T B \Delta v + \omega(||\Delta y||_{\nabla^2 F(y_0)}). \end{aligned}$$

Since $\bar{y} \in \mathcal{L}$ and also $y_0 \in \mathcal{L}$, we conclude that $E(\bar{z} - z_0) + B(\bar{v} - v(z_0)) = E\Delta z + B\Delta v = 0$. Therefore, $B\Delta v = -E\Delta z$ and we get

$$F(\bar{y}) \geq \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \omega(||\Delta y||_{\nabla^2 F(y_0)}).$$

Let us minimize now both sides of the inequality for all feasible v, i.e. for all v such that $E\bar{z} + Bv = d$. Then we get

$$\begin{aligned}\varphi(\bar{z}) &\geq \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \omega\left(\min_{v:Bv=d-E\bar{z}} ||\Delta z, v - v^*(z_0)||_{\nabla^2 F(y_0)}\right)\\ &= \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \omega\left(\min_{\Delta v:B\Delta v=-E\Delta z} ||\Delta z, \Delta v||_{\nabla^2 F(y_0)}\right).\end{aligned}$$

It follows, using Corollary 5.2.5

$$\varphi(\bar{z}) \geq \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \omega\left(\left(\Delta z^T\nabla^2\varphi(z)\Delta z\right)^{1/2}\right).$$

Inequality (2.12) implies that φ is indeed a self-concordant function.

Let us show now that φ is also a ν-self-concordant barrier for $\mathcal{C}$. Using again (5.6) and (5.8) and the fact that $B\Delta v = -E\Delta z$, we have

$$\begin{aligned}\nabla F(y_0)^T\Delta y &= F'_z(y_0)^T\Delta z + F'_v(y_0)^T\Delta v\\ &= (\nabla\varphi(z_0) - E^T\lambda_v(z_0))^T\Delta z - \lambda_v(z_0)^T B\Delta v\\ &= (\nabla\varphi(z_0) - E^T\lambda_v(z_0))^T\Delta z + \lambda_v(z_0)^T E\Delta z\\ &= \nabla\varphi(z_0)^T\Delta z.\end{aligned}$$

Using Theorem 2.4.7, we get

$$\begin{aligned}F(\bar{y}) &\geq F(y_0) + \nabla F(y_0)^T\Delta y + \nu\omega_*\left(\frac{1}{\nu}\nabla F(y_0)^T\Delta y\right)\\ &= \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \nu\omega_*\left(\frac{1}{\nu}\nabla\varphi(z_0)^T\Delta z\right).\end{aligned}$$

Minimizing the left-hand side of the inequality with respect to all feasible v we get

$$\varphi(\bar{z}) \geq \varphi(z_0) + \nabla\varphi(z_0)^T\Delta z + \nu\omega_*\left(\frac{1}{\nu}\nabla\varphi(z_0)^T\Delta z\right).$$

According to Theorem 2.4.7 this means that φ is a ν-self-concordant barrier for $\mathcal{C}$. □

5.2.2 Two examples of partial minimization

Explicit partial minimizer

Let us consider the convex constraint

$$e^{u^2} \leq w. \tag{5.17}$$

We see that (5.17) can be written equivalently as

$$\begin{aligned} e^v &\leq w \\ u^2 &\leq v \end{aligned} \tag{5.18}$$

by introducing an artificial variable v. Moreover, we know that

$$F_1(v,w) = -\log(w) - \log(\log(w) - v)$$

is a 2-self-concordant barrier for the set $\{e^v \leq w\}$ (see e.g. [44]), and

$$F_2(u,v) = -\log(v - u^2)$$

is a 1-self-concordant barrier for $\{u^2 \leq v\}$ (see e.g. [46, Example 4.2.1(4)]). We conclude that

$$F(u,w,v) = -\log(w) - \log(\log(w) - v) - \log(v - u^2)$$

is a 3-self-concordant barrier for (5.18). On the other hand, we do not need the artificial variable v. Minimizing F with respect to v gives as optimal solution

$$v^* = \frac{1}{2}(u^2 + \log(w)).$$

Therefore we can conclude that

$$\varphi(u,w) = -\log(w) - 2\log(\log(w) - u^2) + 2\log(2)\,.$$

is a 3-self-concordant barrier for (5.17).

Unfortunately, it is not always possible to compute analytically the partial minimizer. In the above example this was possible because the optimality conditions for minimizing F with respect to v were particularly simple. However, this is not always the case, as the following example illustrates.

Implicit partial minimizer

Let us consider the convex constraint

$$\sum_{i=1}^{n} \exp(u_i) \leq u_0 \tag{5.19}$$

which is equivalent to

$$\begin{aligned} \exp(u_i) &\leq v_i \\ \sum_{i=1}^{n} v_i &= u_0. \end{aligned} \tag{5.20}$$

The second formulation has the advantage of having an explicit barrier available. Indeed,

$$F(u,v) = \sum_{i=1}^{n} (-\log(v_i) - \log(\log(v_i) - u_i))$$

is a $2n$-self-concordant barrier for (5.20). On the other hand, if we want to minimize a linear function over (5.19), we only need a self-concordant barrier for this reduced set. According to Theorem 5.2.1 the implicit function

$$\varphi(u) := \min_{v:\sum v_i = u_0} F(u,v)$$

is a $2n$-self-concordant barrier for (5.19). The optimality conditions for the above partial minimization subproblem are: $\exists\ \lambda$:

$$\begin{aligned} -\frac{1}{v_i} - \frac{1}{(\log(v_i)-u_i)\cdot v_i} &= \lambda\ \forall i \\ \sum_{i=1}^{n} v_i &= u_0. \end{aligned}$$

It seems difficult to find an analytic solution to these optimality conditions. However, we might try to find an approximate solution $\bar{v}$, as it will be described in Section 5.3.

5.2.3 Analytic centering using partial minimization

We consider now the analytic centering problem of minimizing F restricted to linear subspaces defined by $\mathcal{L}$ and $\bar{\mathcal{L}}$, i.e.

$$\begin{aligned} \min_{z,v}\ & F(z,v) \\ Ez + Bv &= d \\ Gz &= g. \end{aligned} \tag{AC}$$

Using Theorem 5.2.1, the analytic centering problem (AC) can be written as

$$\begin{aligned} \min_{z} \varphi(z) \\ Gz = g. \end{aligned} \tag{5.21}$$

It is clear that if we want to make sure that a solution to (5.21) exists, we have to add the assumption that $\mathcal{C}\bigcap\bar{\mathcal{L}}$ is bounded.

In order to solve (5.21) we employ a damped Newton scheme as discussed in Section 2.3. The Newton directions are computed by solving the linear system

$$\begin{bmatrix} \nabla^2\varphi(z) & G^T \\ G & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta z \\ \lambda_z \end{bmatrix} = \begin{bmatrix} -\nabla\varphi(z) \\ 0 \end{bmatrix}, \tag{5.22}$$

where the gradient and Hessian of φ are given by (5.8) and (5.9).

Using Theorem 2.3.7, we conclude that we can solve (5.21) up to accuracy ϵ in no more than

$$\frac{\varphi(z_0)-\varphi(z^*)}{\omega(\bar{\beta})} + \mathcal{O}\left(\log_2\left(\log_2\left(1/\bar{\epsilon}\right)\right)\right)$$

iterations, where $\bar{\beta} \in \left(0, \frac{3-\sqrt{5}}{2}\right)$ and $\bar{\epsilon} := \omega_*^{-1}(\epsilon)$. Let $\bar{z}$ such an approximate solution for (5.21). Then we have that $(\bar{z}, v(\bar{z}))$ is an ϵ-solution for the full problem (AC).

In view of the observations from Section 2.3.3, we have that the numerical cost of solving (5.22) is

$$\mathcal{O}\left(\frac{1}{3}(n_1+m_2)^3\right)$$

flops, while the complexity of solving the Newton system corresponding to the full analytic centering problem (AC) would be

$$\mathcal{O}\left(\frac{1}{3}(n_1+n_2+m_1+m_2)^3\right)$$

flops, since the number of variables is $n = n_1 + n_2$ and the number of constraints is $m = m_1 + m_2$. We see that the improvement is significant if n_2 and/or m_1 are large as compared to n_1 and m_2. Note that the above complexity does not take sparsity into account, in which case the complexity of solving (AC) would be lower.

5.2.4 Solving convex problems using partial minimization

We consider now the convex optimization problem of minimizing a linear function a over $\mathcal{C} \bigcap \bar{\mathcal{L}}$.

$$\begin{aligned} &\min a^T z \\ &\text{s.t. } z \in C \\ &\quad Gz = g. \end{aligned} \tag{5.23}$$

Since the implicit barrier $\varphi(z)$ is a nondegenerate ν-self-concordant barrier for $\mathcal{C}$, according to Theorem 2.4.14 the complexity of solving (5.23) is bounded by

$$N \leq \mathcal{O}\left(\sqrt{\nu} \cdot \log\left(\frac{\nu}{\epsilon}\right)\right)$$

iterations. Moreover, the cost per iteration is significantly lower as compared to the lifted formulation involving the higher-dimensional set $\mathcal{Q}$ (see Section 5.2.3).

5.3 Approximate partial minimization

In the previous section we have showed that minimizing a self-concordant barrier with respect to some variables that are not needed, preserves the self-concordance property. As a consequence, we are able to solve the original problem in the lower-dimensional space using that implicit barrier. However, this result is only of limited direct use, because often we will not be able to compute the partial minimizer analytically as in the first example in Section 5.2.2. In this section we illustrate how the concept of partial minimization can be used even if the partial minimizer is only computed approximately.

Let us recall the partial minimization subproblem for a fixed strictly feasible point $z \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$:

$$\begin{aligned} \varphi(z) = &\min_v F(z,v) \\ &\text{s.t. } Bv = d - Ez. \end{aligned} \tag{PM(z)}$$

As we have argued in Section 5.2, Assumptions 4 guarantee that for any $z \in \operatorname{int}\mathcal{C} \bigcap \bar{\mathcal{L}}$ a solution for (PM(z)) exists and is unique.

5.3.1 Computing the approximate partial minimizer

In general it is not possible to find analytically a solution for (PM(z)). However, since F is a self-concordant function, we can efficiently compute an approximate solution for (PM(z)) by a damped Newton scheme. Let $(z, v) \in \operatorname{int}\mathcal{Q} \bigcap \mathcal{L}$ be a strictly feasible starting point. The Newton directions can be obtained by solving the following linear system

$$\begin{bmatrix} F^{''}_{vv}(z,v) & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta v \\ \lambda_v \end{bmatrix} = \begin{bmatrix} -F^{'}_v(z,v) \\ 0 \end{bmatrix}. \tag{5.24}$$

For the sake of simplifying notation we denote $F^{''}_{vv} = F^{''}_{vv}(z,v)$ and $F^{'}_v = F^{'}_v(z,v)$. For a *fixed* $z \in \operatorname{int}\mathcal{C} \bigcap \mathcal{L}$ the domain of F is bounded by Assumption 4. Therefore the v-block of the Hessian ($F^{''}_{vv}$) is nondegenerate and we get as solution for (5.24)

$$\Delta v = -(F^{''}_{vv})^{-1}\,(F^{'}_v + B^T \lambda_v).$$

Since B has full row rank we can pre-multiply the above expression by B and get

$$\lambda_v = -\left[B(F^{''}_{vv})^{-1}B^T\right]^{-1} B(F^{''}_{vv})^{-1}F^{'}_v,$$

(see also Section 2.3.3).

In order to solve (PM(z)) we apply several damped Newton steps $v^+ = v + \gamma \cdot \Delta v$, where γ is a a suitable step size parameter. Lemma 2.3.6 provides us with a safeguard step length $\gamma = \frac{1}{1+\delta_v}$ that guarantees a decrease in the objective value (in practice we might try to find a larger step size). Here, δ_v denotes the Newton decrement $\delta_v := ||\Delta v||_v = (\Delta v^T F^{''}_{vv} \Delta v)^{1/2}$. The stopping criterion for the damped Newton scheme is the size of the Newton decrement: as soon as $\delta_v \leq \beta_v < 1$ we call the current iterate $\bar{v}$ a β_v-approximation to the exact partial minimizer $v(z)$ (and $\bar{\lambda}$ the corresponding multiplier that approximates the exact multiplier $\lambda_v(z)$). Then we have that $F(z, \bar{v})$ is an approximation for the value of φ at z.

5.3.2 Analytic centering using approximate partial minimization

In this section we consider again the analytic centering problem (AC), i.e.

$$\begin{aligned} \min_{z,v}\ & F(z,v) \\ Ez + Bv &= d \\ Gz &= g. \end{aligned} \tag{AC}$$

As in Section 5.2.3, in order to guarantee that a solution to (AC) exists we impose the additional assumption that $\mathcal{C} \bigcap \bar{\mathcal{L}}$ is bounded. Note, however, that this additional assumption is not needed when there is additionally a nontrivial linear term in the objective present. This situation occurs whenever we solve the sequence of centering problems as it is described in Section 5.3.3.

Since F is a self-concordant function, we can solve this problem using a damped Newton scheme, whose complexity is

$$\mathcal{O}\left(F(y_0) - F(y^*)\right) + \mathcal{O}\left(\log_2\left(\log_2\left(1/\omega_*^{-1}(\epsilon)\right)\right)\right)$$

iterations to generate an approximate solution $\bar{y}$ such that

$$F(\bar{y}) - F(y^*) \leq \epsilon,$$

(see Theorem 2.3.7). The cost of computing the Newton directions is

$$\mathcal{O}\left(\frac{1}{3}(n+m)^3\right),$$

where $n = n_1 + n_2$ and $m = m_1 + m_2$ (see Section 2.3.3). We see that the complexity of one iteration increases dramatically when the number of variables n, or the number of constraints m, gets large.

The main objective of this section is to modify the standard damped Newton method (Algorithm 2) so that the cost per iteration is reduced. The comment at the end of Section 5.2.3 indicates that there is indeed hope for improvement, provided that the partial minimization subproblem (PM(z)) can be solved efficiently.

Newton steps in a sequence of affine subspaces

For a given strictly feasible point $z \in \operatorname{int}\mathcal{C} \bigcap \bar{\mathcal{L}}$ and an approximate partial minimizer $\bar{v} \approx v(z)$ (that we compute as described in Section 5.3.1), we can approximate the *value* of the implicit barrier φ at z. Indeed, let δ_v be the Newton decrement for the partial minimization subproblem (PM(z)) at the point $(z, \bar{v})$. If $\delta_v < 1$, then we have according to Theorem 2.3.4

$$F(z, \bar{v}) - \varphi(z) \leq \omega_*(\delta_v).$$

However, in order to work in an outer Newton scheme only in terms of the variables z, it is necessary to approximate the gradient and Hessian of φ at z. We showed that (5.8) and (5.9) give the gradient and Hessian of φ, provided that $\bar{v}$ is indeed the *exact* partial minimizer for z. We could simply replace $v(z)$ by $\bar{v}$ in formulae (5.8) and (5.9), but then we would only get *approximations* for $\nabla\varphi(z)$ and $\nabla^2\varphi(z)$. Unfortunately, we cannot say anything a priori about the quality of these approximations. However, we will show that for $\bar{v} = v(z)$ (5.8) and (5.9) can be thought of as the gradient and Hessian of a *restriction* of F to an affine subspace, which is tangent to the surface of partial minimizers at the point $(z, v(z))$. Moreover, for $\bar{v} \neq v(z)$ we get a whole *family of restrictions* of the barrier F.

Let us consider a fixed point $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$. We could take for example as $\bar{v}$ an approximate partial minimizer for $z \in \operatorname{int}\mathcal{C} \bigcap \bar{\mathcal{L}}$ that we compute as described in Section 5.3.1. Let us define the matrices $J(z, \bar{v})$ and $L(z, \bar{v})$ to be the unique solutions to (5.13), i.e.

$$\begin{bmatrix} F''_{vv}(z, \bar{v}) & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} J(z, \bar{v}) \\ L(z, \bar{v}) \end{bmatrix} = \begin{bmatrix} -F''_{vz}(z, \bar{v}) \\ -E \end{bmatrix}.$$

We have pointed out in the proof of Corollary 5.2.5, that if $\bar{v}$ is the partial minimizer for z, i.e. $\bar{v} = v(z)$, then $J(z, v(z)) = v'(z)$ and $L(z, v(z)) = \lambda'_v(z)$.

Let us fix $J = J(z, \bar{v})$ and define for points $\tilde{z}$ around z the function

$$\psi(\tilde{z}) = F(\tilde{z}, \bar{v} + J\,(\tilde{z} - z)),$$

relative to the fixed point $(z, \bar{v})$.

Lemma 5.3.1. *For any* $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$ *we have*

$$\operatorname{dom} \psi \subseteq \operatorname{int} \mathcal{C}.$$

Proof. Let $\tilde{z} \in \operatorname{dom} \psi$ and denote $\tilde{v} = \bar{v} + J\,(\tilde{z} - z)$. We have by definition of ψ that

$$F(\tilde{z}, \tilde{v}) < \infty,$$

that is, $(\tilde{z}, \tilde{v}) \in \operatorname{dom} F = \operatorname{int} \mathcal{Q}$.

Further, in view of (5.13) we have

$$B\tilde{v} + E\tilde{z} = B\bar{v} + \underbrace{BJ}_{=-E}(\tilde{z} - z) + E\tilde{z} = B\bar{v} + Ez = d,$$

which means $(\tilde{z}, \tilde{v}) \in \mathcal{L}$. By Assumption 4 we conclude that $\tilde{z} \in \operatorname{int} \mathcal{C}$. □

Let us compute now the derivatives of ψ.

$$\begin{aligned}
\nabla\psi(\tilde{z}) =& F'_z(\tilde{z}, \bar{v} + J\,(\tilde{z} - z)) + J^T\,F'_v(\tilde{z}, \bar{v} + J\,(\tilde{z} - z)),\\
\nabla^2\psi(\tilde{z}) =& F''_{zz}(\tilde{z}, \bar{v} + J \cdot (\tilde{z} - z)) + F''_{zv}(\tilde{z}, \bar{v} + J\,(\tilde{z} - z))\,J\\
&+ J^T\,F''_{vz}(\tilde{z}, \bar{v} + J\,(\tilde{z} - z)) + J^T\,F''_{vv}(\tilde{z}, \bar{v} + J\,(\tilde{z} - z))\,J.
\end{aligned}$$

For $\tilde{z} = z$ we denote

$$\begin{aligned}
h(z, \bar{v}) := \nabla\psi(z) =& F'_z(z, \bar{v}) + J^T\,F'_v(z, \bar{v}), \qquad (5.25)\\
H(z, \bar{v}) := \nabla^2\psi(z) =& F''_{zz}(z, \bar{v}) + F''_{zv}(z, \bar{v})\,J\\
&+ \underbrace{J^T\,F''_{vz}(z, \bar{v}) + J^T\,F''_{vv}(z, \bar{v})\,J}_{\overset{(5.13)}{=} -J^T B^T L \overset{(5.13)}{=} E^T L}\\
=& F''_{zz}(z, \bar{v}) + F''_{zv}(z, \bar{v})\,J + E^T L. \qquad (5.26)
\end{aligned}$$

In view of Lemma 5.3.1 and Assumption 4 we have that $\operatorname{dom} \psi$ does not contain straight lines. According to Theorem 2.2.4 the Hessian of ψ is then nonsingular for every point $\tilde{z} \in \operatorname{dom} \psi$ (and in particular for z from above). This justifies that $H(z, \bar{v})$ is a positive definite matrix.

It is easy to see that the function ψ can be considered as a restriction of F to the affine subspace

$$\mathcal{L}_J = \{(\tilde{z}, \tilde{v}) : \tilde{v} = \bar{v} + J\,(\tilde{z} - z)\}$$

relative to $(z, \bar{v})$. From the proof of Lemma 5.3.1 it follows directly that $\mathcal{L}_J \subseteq \mathcal{L}$ for all $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$.

The approach of restricting F to $\mathcal{L}_J$ is justified by the following two facts. First, we can remove the dependence on the variables v while still maintaining feasibility with respect to the linear constraints $\mathcal{L}$. Second, if we assume that $\bar{v}$ is the exact partial minimizer (i.e. $\bar{v} = v(z_0)$), then according to the optimality conditions (5.6) we have necessarily

$$F'_v(z, v(z)) = -B^T \lambda_v(z),$$

and also $J = v'(z)$ and $L = \lambda'_v(z)$ (because (5.13) and (5.7) are the same system), which implies that $\nabla\psi(z) = \nabla\varphi(z)$ and $\nabla^2\psi(z) = \nabla^2\varphi(z)$. That means also that, at the point z_0, the Newton direction for minimizing ψ is *equal* to the Newton direction for the implicit barrier φ.

For $\bar{v} \neq v(z)$ we have the following interpretation of the Newton direction for minimizing ψ.

Lemma 5.3.2. *Let $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$ and $\Delta y = (\Delta y_1, \Delta y_2)$ the Newton direction for minimizing F restricted to $\mathcal{L}$, at the point $(z, \bar{v})$. Then Δy_1 is the Newton direction for minimizing ψ at z.*

Proof. We consider the analytic centering problem

$$\min_{z,v} \; F(z, v)$$
$$Ez + Bv = d.$$

Throughout this proof, all partial derivatives are taken at the point $(z, \bar{v})$, and to simplify notation we write F'_v for $F'_v(z, \bar{v})$ etc. The Newton direction $\Delta y = (\Delta y_1, \Delta y_2)$ is the unique solution of

$$\begin{bmatrix} F''_{zz} & F''_{zv} & E^T \\ F''_{vz} & F''_{vv} & B^T \\ E & B & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \lambda_y \end{bmatrix} = \begin{bmatrix} -F'_z \\ -F'_v \\ 0 \end{bmatrix}. \tag{5.27}$$

Let us compute explicitly Δy_1. From the second equation of (5.27) it follows

$$\Delta y_2 = -(F''_{vv})^{-1} \cdot [F'_v + F''_{vz}\Delta y_1 + B^T \lambda_y] \tag{5.28}$$

and by we replacing this expression for Δy_2 in (5.27), we get

$$\begin{bmatrix} S & M^T \\ M & -B(F''_{vv})^{-1}B^T \end{bmatrix} \cdot \begin{bmatrix} \Delta y_1 \\ \lambda_y \end{bmatrix} = \begin{bmatrix} -F'_z + F''_{zv}(F''_{vv})^{-1}F'_v \\ B(F''_{vv})^{-1}F'_v \end{bmatrix}, \tag{5.29}$$

where $M = E - B(F''_{vv})^{-1}F''_{vz}$ and $S = F''_{zz} - F''_{zv}(F''_{vv})^{-1}F''_{vz}$. Using the fact that B has full row rank, we isolate λ_y in the the second equation of (5.29) and get

$$\lambda_y = \left(B(F''_{vv})^{-1}B^T\right)^{-1} [M \cdot \Delta y_1 - B(F''_{vv})^{-1}F'_v]. \tag{5.30}$$

Let us replace this expression for λ_y in (5.29), which yields

$$\begin{aligned} &\left[S + M^T \left(B(F''_{vv})^{-1}B^T\right)^{-1} M\right] \cdot \Delta y_1 \\ &\qquad = -F'_z + \left[F''_{zv} + M^T \left(B(F''_{vv})^{-1}B^T\right)^{-1} B\right] \cdot (F''_{vv})^{-1}F'_v. \end{aligned} \tag{5.31}$$

On the other hand, the matrices J and L are the solutions of the system (5.13) relative to the point $(z, \bar{v})$, i.e.

$$\begin{bmatrix} F_{vv}^{''} & B^T \\ B & 0 \end{bmatrix} \cdot \begin{bmatrix} J \\ L \end{bmatrix} = \begin{bmatrix} -F_{vz}^{''} \\ -E \end{bmatrix}.$$

We obtain as solutions

$$L = -(BF_{vv}^{''\,-1}B^T)^{-1} \cdot \underbrace{[-E + BF_{vv}^{''\,-1}F_{vz}^{''}]}_{=-M} = \left(BF_{vv}^{''\,-1}B^T\right)^{-1} \cdot M \tag{5.32}$$

and

$$\begin{aligned} J &= -(F_{vv}^{''})^{-1}B^T\, L - (F_{vv}^{''})^{-1}F_{vz}^{''} \\ &= -(F_{vv}^{''})^{-1}B^T \left(BF_{vv}^{''\,-1}B^T\right)^{-1} M - (F_{vv}^{''})^{-1}F_{vz}^{''} \\ &= -(F_{vv}^{''})^{-1} \left[B^T \left(BF_{vv}^{''\,-1}B^T\right)^{-1} M + F_{vz}^{''}\right]. \end{aligned} \tag{5.33}$$

Let us have a closer look at the system (5.31). Using (5.33), its right-hand side term can be written as

$$-F_z' + \underbrace{\left[F_{zv}^{''} + M^T \left(B(F_{vv}^{''})^{-1}B^T\right)^{-1} B\right] \cdot (F_{vv}^{''})^{-1}}_{=-J^T} F_v' = -F_z' - J^T F_v',$$

and by comparing with (5.25) it follows that the above expression is exactly $-\nabla\psi(z)$.

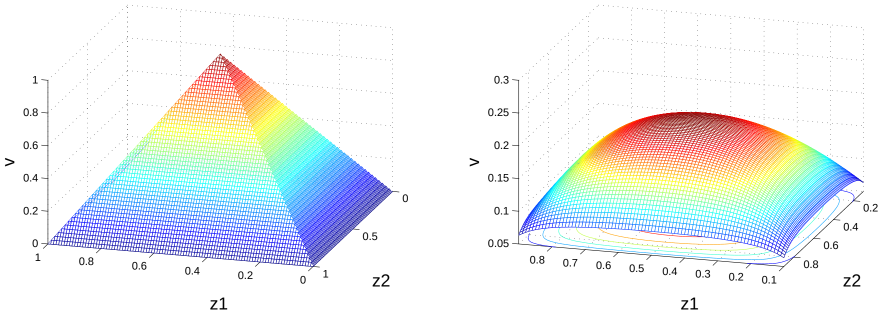
Further, if we combine (5.26) with (5.32) and (5.33), we get

$$\begin{aligned} \nabla^2\psi(z) &= F_{zz}^{''} + F_{zv}^{''} \cdot J + E^T L \\ &= \underbrace{F_{zz}^{''} - F_{zv}^{''}(F_{vv}^{''})^{-1}F_{vz}^{''}}_{=S} - F_{zv}^{''}(F_{vv}^{''})^{-1}B^T \left(B(F_{vv}^{''})^{-1}B^T\right)^{-1} M + E^T L \\ &= S + \underbrace{\left[-F_{zv}^{''}(F_{vv}^{''})^{-1}B^T + E^T\right]}_{=M^T} \left(B(F_{vv}^{''})^{-1}B^T\right)^{-1} M \\ &= S + M^T \left(B(F_{vv}^{''})^{-1}B^T\right)^{-1} M. \end{aligned}$$

The last expression is exactly the system matrix of (5.31). That means the system (5.31) can be written as

$$\nabla^2\psi(z) \cdot \Delta y_1 = -\nabla\psi(z).$$

which is exactly the Newton system for minimizing ψ at the point z. □



(a) (b)

Figure 5.1: (a) The pyramid $\mathcal{Q}$, (b) the surface of partial minimizers $v(z)$.

Graphical illustration of partial minimization

We consider the convex set

$$\mathcal{Q} = \{(z_1, z_2, v) : v \leq 2z_1, v \leq 2z_2, v \leq 2 - 2z_1, v \leq 2 - 2z_2, v \geq 0\}.$$

$\mathcal{Q}$ is the pyramid which is illustrated in Figure 5.1(a).

We know that a self-concordant barrier for $\mathcal{Q}$ is given by

$$F(z, v) = -\log(2z_1 - v) - \log(2z_2 - v) - \log(2 - 2z_1 - v) - \log(2 - 2z_1 - v) - \log(v).$$

It is evident that the z-domain of F is simply the unit box $\mathcal{B} = \{z : 0 \leq z_i \leq 1, i = 1, 2\} \subseteq \mathbb{R}^2$. If we minimize F with respect to v for all strictly feasible z (that is $0 < z_i < 1$), we obtain the surface $v(z) \subset \mathcal{Q}$ which is visualized in Figure 5.1(b). The global minimizer of F is the analytic center of $\mathcal{C}$, which is situated on the peak of $v(z)$. In Figure 5.2 we illustrate again the surface $v(z)$ and additionally two hyperplanes. The lower hyperplane $\mathcal{L}_{v'(\bar{z})}$ is tangent to $v(z)$ going through the point $(\bar{z}, v(\bar{z}))$ for some $\bar{z} \in \operatorname{int} \mathcal{B}$. The upper hyperplane is $\mathcal{L}_J$, which is going through $(\bar{z}, \bar{v})$, where $\bar{v}$ is an approximation for the exact partial minimizer $v(\bar{z})$. We see that $\mathcal{L}_J$ is approximately tangent to $\mathcal{L}_{v'(\bar{z})}$. The Newton step in $\mathcal{L}_J$ is depicted by the arrow.

Bounds on the approximation for the reduced Hessian

As we have mentioned above, if we only have an approximation $\bar{v}$ for the partial minimizer $v(z)$ for some $z \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$, then (5.26) yields only an approximation for the Hessian of the implicit barrier φ. The following result gives a bound on the quality of that approximation $H = \nabla^2 \psi(z)$. In the following, for the fixed point $(z, \bar{v})$ and some displacement $\Delta v \in \mathbb{R}^{n_2}$ we denote $||\Delta v||_{\bar{v}} = ||\Delta v||_{F''_{vv}(z,\bar{v})}$.

Lemma 5.3.3. *Let $z \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$ and $\bar{v}$ such that $||\bar{v} - v(z)||_{\bar{v}} \leq r < 1$. Then we have for any direction Δz*

$$(1 - r) \, ||\Delta z||_{\nabla^2 \varphi(z)} \leq ||\Delta z||_H \leq \frac{1}{1 - r} \, ||\Delta z||_{\nabla^2 \varphi(z)}.$$

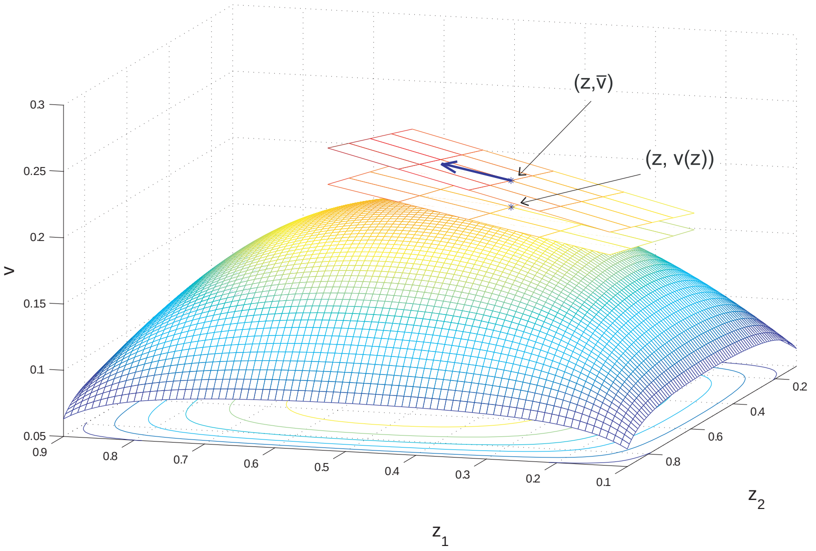


Figure 5.2: Surface of partial minimizers $v(z)$ with the tangent hyperplane $\mathcal{L}_{v'(z)}$ and the approximately tangent hyperplane $\mathcal{L}_J$ containing the Newton direction in $\mathcal{L}_J$, indicated as the arrow.

Proof. First, note that $||\bar{v} - v(z)||_{\bar{v}} \leq r < 1$ implies

$$\left|\left| \begin{pmatrix} 0 \\ \bar{v} - v(z) \end{pmatrix} \right|\right|_{\nabla^2 F(z,\bar{v})} = \left|\left| \begin{pmatrix} z \\ \bar{v} \end{pmatrix} - \begin{pmatrix} z \\ v(z) \end{pmatrix} \right|\right|_{\nabla^2 F(z,\bar{v})} \leq r < 1.$$

Let us fix Δz and choose Δv to be any feasible direction with respect to the subspace $\mathcal{L}$, i.e. such that $B\Delta v + E\Delta z = 0$. Then according to Theorem 2.2.10 it holds

$$(1-r)^2 \left\langle \nabla^2 F(z, v(z)) \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix}, \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix} \right\rangle \leq \left\langle \nabla^2 F(z, \bar{v}) \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix}, \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix} \right\rangle. \quad (5.34)$$

If we minimize the right-hand side term of inequality (5.34) with respect to all feasible directions Δv, we get according to Corollary 5.2.4

$$\begin{aligned} \langle H\Delta z, \Delta z \rangle &= \min_{\Delta v : B\Delta v + E\Delta z = 0} \left\{ \left\langle \nabla^2 F(z, \bar{v}) \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix}, \begin{pmatrix} \Delta z \\ \Delta v \end{pmatrix} \right\rangle \right\} \\ &= \left\langle \nabla^2 F(z, \bar{v}) \begin{pmatrix} \Delta z \\ \Delta v_1 \end{pmatrix}, \begin{pmatrix} \Delta z \\ \Delta v_1 \end{pmatrix} \right\rangle, \end{aligned}$$

assuming that the minimum is attained at Δv_1. It follows, using (5.34)

$$\begin{aligned}\left\langle \nabla^2 F(z,\bar v)\begin{pmatrix}\Delta z\\ \Delta v_1\end{pmatrix},\begin{pmatrix}\Delta z\\ \Delta v_1\end{pmatrix}\right\rangle &\geq (1-r)^2\left\langle \nabla^2 F(z,v(z))\begin{pmatrix}\Delta z\\ \Delta v_1\end{pmatrix},\begin{pmatrix}\Delta z\\ \Delta v_1\end{pmatrix}\right\rangle\\ &\geq \min_{B\Delta v=-E\Delta z}\left\{(1-r)^2\left\langle \nabla^2 F(z,v(z))\begin{pmatrix}\Delta z\\ \Delta v\end{pmatrix},\begin{pmatrix}\Delta z\\ \Delta v\end{pmatrix}\right\rangle\right\}\\ &= (1-r)^2\,\langle \nabla^2\varphi(z)\Delta z,\Delta z\rangle,\end{aligned}$$

using again Corollary 5.2.4. In other words, we have shown

$$||\Delta z||_H \geq (1-r)\,||\Delta z||_{\nabla^2\varphi(z)}.$$

With exactly the same arguments, the converse inequality holds, i.e.

$$||\Delta z||_{\nabla^2\varphi(z)} \geq (1-r)\,||\Delta z||_H.$$

□

Link between the Newton decrements

The following two results are crucial for the design of efficient algorithms using approximate partial minimization, as they provide the link between the δ_y, δ_v and δ_z, where δ_y is the Newton decrement for the full analytic centering problem (AC), δ_v is the Newton decrement for the partial minimization subproblem (PM(z)) and δ_z is the Newton decrement for the problem

$$\min_{Gz=g}\ \psi(z).$$

First let us consider the situation where there are no linear constraints that only involve z. In that case, the analytic centering problem (AC) reduces to

$$\begin{aligned}\min_{z,v}\ &F(z,v)\\ Ez+Bv&=d.\end{aligned}\tag{AC1}$$

For any $\bar y=(z,\bar v)\in \operatorname{int}\mathcal{Q}\bigcap\mathcal{L}$ we denote the full Newton decrement by $\delta_y = ||\Delta y||_{\bar y} = ||\Delta y||_{\nabla^2 F(\bar y)}$, where $\Delta y=(\Delta y_1,\Delta y_2)\in\mathbb{R}^{n_1+n_2}$ is the solution of (5.27).

The Newton decrement for the partial minimization subproblem (PM(z)) is denoted by $\delta_v = ||\Delta v||_{\bar v} = ||\Delta v||_{F''_{vv}(\bar y)}$, where Δv is the solution of (5.24), i.e.

$$\begin{bmatrix}F''_{vv} & B^T\\ B & 0\end{bmatrix}\cdot\begin{bmatrix}\Delta v\\ \lambda_v\end{bmatrix}=\begin{bmatrix}-F'_v\\ 0\end{bmatrix}.$$

Given $\bar v$, we compute J and L from (5.13) and build the gradient and Hessian of ψ from (5.25) and (5.26). The Newton direction Δz for minimizing ψ is given by the solution of

$$\nabla^2\psi(z)\,\Delta z = -\nabla\psi(z).\tag{5.35}$$

We denote the Newton decrement by $\delta_z = ||\Delta z||_{\nabla^2\psi(z)}$.

Theorem 5.3.4. *Let $\bar{y} = (z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$, where B has full row rank. Then*

$$\delta_y = \sqrt{\delta_z^2 + \delta_v^2}.$$

Proof. We get

$$\begin{aligned} \delta_y^2 = ||\Delta y||^2_{\nabla^2 F(\bar{y})} &= -\nabla F(\bar{y})^T \Delta y \\ &= -\Delta y_1^T F_z' - \Delta y_2^T F_v'. \end{aligned}$$

Using Lemma 5.3.2 it holds $\Delta y_1 = \Delta z$. Further, since $\nabla \psi(z) = F_z' + J^T F_v'$, we get

$$\begin{aligned} -\Delta y_1^T F_z' &= -\Delta z^T F_z' \\ &= -\Delta z^T \left(\nabla \psi(z) - J^T F_v'\right) \\ &= \delta_z^2 + \Delta z^T J^T F_v', \end{aligned} \tag{5.36}$$

where J is given by (5.33), i.e.

$$J = -(F_{vv}^{''})^{-1} \left[F_{vz}^{''} + B^T \left(B (F_{vv}^{''})^{-1} B^T \right)^{-1} M \right].$$

Replacing the above expression for J in the second term of (5.36), we get

$$\begin{aligned} &\Delta z^T J^T F_v' \\ &= -\Delta z^T F_{zv}^{''} (F_{vv}^{''})^{-1} F_v' - \Delta z^T M^T \left(B (F_{vv}^{''})^{-1} B^T \right)^{-1} B (F_{vv}^{''})^{-1} F_v'. \end{aligned} \tag{5.37}$$

On the other hand, according to (5.28) we have

$$\Delta y_2 = -(F_{vv}^{''})^{-1} \left[F_v' + F_{vz}^{''} \Delta z + B^T \lambda_y \right],$$

therefore

$$\begin{aligned} -\Delta y_2^T F_v' &= \left[F_v' + F_{vz}^{''} \Delta z + B^T \lambda_y \right]^T (F_{vv}^{''})^{-1} F_v' \\ &= F_v'^T (F_{vv}^{''})^{-1} F_v' + \Delta z^T F_{zv}^{''} (F_{vv}^{''})^{-1} F_v' + \lambda_y^T B (F_{vv}^{''})^{-1} F_v', \end{aligned}$$

and in accordance with (5.30) it becomes

$$\begin{aligned} -\Delta y_2^T F_v' =& F_v'^T (F_{vv}^{''})^{-1} F_v' + \Delta z^T F_{zv}^{''} (F_{vv}^{''})^{-1} F_v' \\ &+ \Delta z^T M^T \left(B (F_{vv}^{''})^{-1} B^T \right)^{-1} B (F_{vv}^{''})^{-1} F_v' \\ &- F_v'^T (F_{vv}^{''})^{-1} B^T \left(B (F_{vv}^{''})^{-1} B^T \right)^{-1} B (F_{vv}^{''})^{-1} F_v'. \end{aligned} \tag{5.38}$$

Summing (5.37) and (5.38) it gives

$$\begin{aligned} (J \Delta z - \Delta y_2)^T F_v' =& F_v'^T (F_{vv}^{''})^{-1} \left[F_{vv}^{''} - B^T \left(B (F_{vv}^{''})^{-1} B^T \right)^{-1} B \right] (F_{vv}^{''})^{-1} F_v' \\ =& - F_v'^T \Delta v \\ =& \delta_v^2, \end{aligned}$$

which follows from the fact that Δv is the solution of (5.24), i.e.

$$\Delta v = -(F_{vv}^{''})^{-1}F_v^{'} + (F_{vv}^{''})^{-1}B^T \left(B(F_{vv}^{''})^{-1}B^T\right)^{-1} B(F_{vv}^{''})^{-1}F_v^{'}.$$

Combining all the results from above, we get

$$\begin{aligned}\delta_y^2 &= -\Delta z^T\, F_z' - \Delta y_2^T\, F_v' \\ &= \delta_z^2 + \Delta z^T J^T F_v' - \Delta y_2^T\, F_v' \\ &= \delta_z^2 + (J\Delta z - \Delta y_2)^T\, F_v' \\ &= \delta_z^2 + \delta_v^2.\end{aligned}$$

□

We consider now the general situation, where in (AC) there are linear equality constraints only involving z, i.e.

$$\begin{aligned}\min\ & F(z,v) \\ Ez + Bv &= d \\ Gz &= g.\end{aligned} \tag{AC}$$

We denote $\delta_y = ||\Delta y||_{\bar{y}}$, where $\Delta y = (\Delta y_1, \Delta y_2)$ is the solution of the following linear system

$$\begin{bmatrix} F_{zz}^{''} & F_{zv}^{''} & E^T & G^T \\ F_{vz}^{''} & F_{vv}^{''} & B^T & 0 \\ E & B & 0 & 0 \\ G & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y_1 \\ \Delta y_2 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -F_z' \\ -F_v' \\ 0 \\ 0 \end{bmatrix}. \tag{5.39}$$

The Newton decrement for the partial minimization subproblem (PM(z)) is unchanged because additional linear constraints on the outer level have no effect on the minimization in terms of v. Let $\delta_v = ||\Delta v||_{\bar{v}}$, where Δv is the solution of (5.24). Moreover, we consider the problem of minimizing ψ restricted to $\bar{\mathcal{L}}$, i.e.

$$\begin{aligned}\min_z\ & \psi(z) \\ Gz &= g.\end{aligned} \tag{5.40}$$

At the strictly feasible point $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}_J$ we denote $\delta_z = ||\Delta z||_{\nabla^2\psi(z)}$, where Δz is the solution of

$$\begin{bmatrix} \nabla^2\psi(z) & G^T \\ G & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta z \\ \lambda_z \end{bmatrix} = \begin{bmatrix} -\nabla\psi(z) \\ 0 \end{bmatrix}. \tag{5.41}$$

Corollary 5.3.5. *Let $\bar{y} = (z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$ such that $z \in \bar{\mathcal{L}}$, where G and B have full row rank. Then*

$$\delta_y = \sqrt{\delta_z^2 + \delta_v^2}.$$

Proof. The only thing that changes compared to the proof of Theorem 5.3.4 is the definition of Δy and Δz (and as a consequence δ_y and δ_z), because both directions have to take into account the linear constraints $Gz = g$.

It follows from the second equation of (5.39) that

$$\Delta y_2 = -(F''_{vv})^{-1}\,[F'_v + F''_{vz}\Delta y_1 + B^T\lambda_1] \tag{5.42}$$

which reduces the system (5.39) to

$$\begin{bmatrix} S & M^T & G^T \\ M & -B(F''_{vv})^{-1}B^T & 0 \\ G & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y_1 \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -F'_z + F''_{zv}(F''_{vv})^{-1}F'_v \\ B(F''_{vv})^{-1}F'_v \\ 0 \end{bmatrix}, \tag{5.43}$$

where M and S are defined as in the proof of Lemma 5.3.2. Using the fact that B has full row rank, it follows from the second equation of (5.43)

$$\lambda_1 = \left(B(F''_{vv})^{-1}B^T\right)^{-1}\,[M\,\Delta y_1 - B(F''_{vv})^{-1}F'_v], \tag{5.44}$$

and if we replace this in (5.43), we get immediately

$$\begin{bmatrix} H & G^T \\ G & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta y_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} -F'_z - J^T F'_v \\ 0 \end{bmatrix}, \tag{5.45}$$

where $H = \nabla^2\psi(z)$. We see that - analogous to before - $\Delta y_1 = \Delta z$ and $\lambda_2 = \lambda_z$.

Additionally we have

$$\delta_z^2 = ||\Delta z||^2_{\nabla^2\psi(z)} = -\nabla\psi(z)^T\Delta z,$$

and

$$\begin{aligned} \delta_y^2 &= ||\Delta y||^2_{\nabla^2 F(\bar{y})} \\ &= -\nabla F(\bar{y})^T\Delta y \\ &= -F_z'^T\Delta y_1 - F_v'^T\Delta y_2 \\ &= -(\nabla\psi(z) - J^T F'_v)^T\Delta z - F_v'^T\Delta y_2 \\ &= \delta_z^2 + (J\Delta z - \Delta y_2)^T F'_v \\ &= \delta_z^2 + \delta_v^2, \end{aligned}$$

just like in the proof of Theorem 5.3.4. □

The above result provides us with the key for embedding approximate partial minimization in a Newton scheme. Indeed, as soon as we have found a point $\bar{y} = (\bar{z}, \bar{v})$ such that both δ_z and δ_v are small, we can conclude that δ_y is small, without actually knowing the full Newton direction Δy. Additionally, in view of Theorem 2.3.4 we know that $\bar{y}$ is close to the minimizer of F. The damped Newton method using approximate partial minimization is presented in Algorithm 7.

As a first step for the complexity analysis of Algorithm 7 we show that eventually a point such that $\delta_z \leq \beta_z$ will be reached. Indeed, this is guaranteed by showing that in each iteration we decrease the function value of ψ by a nontrivial amount.

Algorithm 7 Damped Newton method for minimizing a self-concordant function due to linear equality constraints, using approximate partial minimization

```
Input: $G$, $B$, $E$, $g$, $d$ as as in Assumption 4, $F : \mathbb{R}^{n_1+n_2} \to \mathbb{R}$ self-concordant on int $\mathcal{Q}$.
Parameter: $0 < \beta_y < \frac{1}{\sqrt{2}}$ the desired accuracy. Set $\beta_z = \beta_v = \frac{1}{\sqrt{2}}\beta_y$, choose $0 < \kappa < (1 - \omega^{-1}(\omega_*(\beta_v)))^3$ and set $\lambda = \frac{\kappa(1-2\beta_v)}{1-\beta_v}$.
Initialize: $(z_0, v_0) \in$ int $\mathcal{Q}$ such that $Gz_0 = g$ and $Bv_0 + Ez_0 = d$, $k = 0$.
  loop
    1) starting at $v_k$, go damped Newton steps (5.24) until $\delta_v = ||\Delta v||_v \leq \beta_v$, output: approximate partial minimizer $\bar{v}$
    2) compute $J_k$ and $L_k$ from (5.13)
    3) define $\nabla^2\psi(z_k) = F''_{zz} + F''_{zv}J_k + E^T L_k$, $\nabla\psi(z_k) = F'_z + J_k^T F'_v$.
    4) compute the Newton direction $\Delta z$ from (5.41)
    5) define the Newton decrement: $\delta_z := ||\Delta z||_{\nabla^2\psi(z_k)}$
    if $\delta_z < \beta_z$ then
      RETURN
    end if
    6) set step size $\alpha = \frac{\lambda}{1+\delta_z}$
    7) update $z_{k+1} = z_k + \alpha\,\Delta z$ and $v_{k+1} = \bar{v} + \alpha\, J_k\,\Delta z$
    8) $k = k + 1$
  end loop
```

Note that the step size in the algorithm is $\alpha = \lambda\,\frac{1}{1+\delta_z}$, where $\lambda = \frac{\kappa(1-2\beta_v)}{1-\beta_v}$ for some $0 < \kappa < [1 - \omega^{-1}(\omega_*(\beta_v))]^3$. That means $\kappa = \tau\,[1 - \omega^{-1}(\omega_*(\beta_v))]^3$ for some $\tau \in (0,1)$ and we obtain

$$\lambda = \tau \cdot \underbrace{[1 - \omega^{-1}(\omega_*(\beta_v))]^3 \cdot \frac{1 - 2\beta_v}{1 - \beta_v}}_{=:\phi(\beta_v)}$$

for some $\tau \in (0,1)$. ϕ is illustrated in Figure 5.3. We see that ϕ is positive and monotonically decreasing from 1 to 0 in the interval $[0, 0.5)$. That means for all $\beta_v \in [0, 0.5)$ the value of ϕ is an upper bound on the parameter λ (depending on our choice of τ). The smaller β_v, the larger λ can be. Reversely, if β_v is close to $\frac{1}{2}$, then λ will be close to 0. In any case it holds $\lambda \in (0,1)$.

Let us analyze which improvement in terms of the objective value ψ we can guarantee when going a damped Newton step in direction Δz with step size $\alpha = \frac{\lambda}{1+\delta_z}$, where $\lambda \in (0,1)$. We know that ψ is self-concordant (see the list of self-concordance preserving operations in Section 2.2.2). This means we can bound the value of ψ at the new iterate $z^+ = z + \alpha\,\Delta z$ in the following way (see

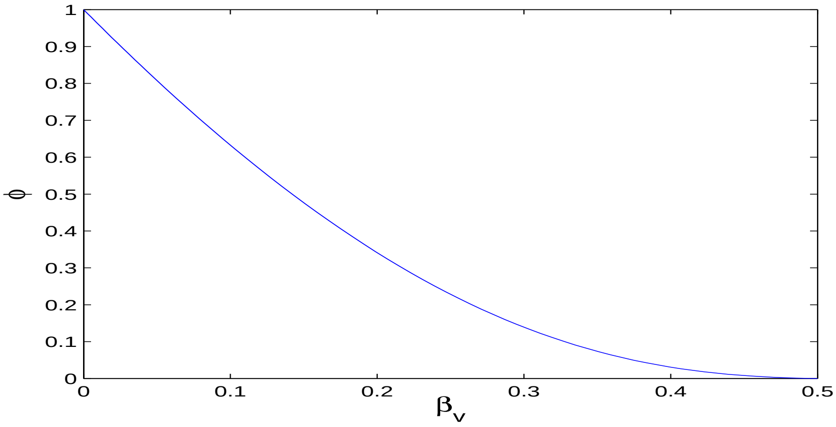


Figure 5.3: Graph of $\phi(\beta_v) = [1 - \omega^{-1}(\omega_*(\beta_v))]^3 \cdot \frac{1-2\beta_v}{1-\beta_v}$.

Theorem 2.2.16):

$$\begin{aligned}
\psi(z^+) &\leq \psi(z) + \alpha\, \nabla\psi(z)^T \Delta z + \omega_*(||z^+ - z||_z) \\
&= \psi(z) - \lambda\frac{\delta_z^2}{1+\delta_z} - \lambda\frac{\delta_z}{1+\delta_z} - \log\left(1 - \lambda\frac{\delta_z}{1+\delta_z}\right) \\
&= \psi(z) - \lambda\delta_z - \log\left(\frac{1+(1-\lambda)\delta_z}{1+\delta_z}\right) \\
&= \psi(z) \underbrace{-\lambda\delta_z + \log\left(1 + \frac{\lambda\delta_z}{1+(1-\lambda)\delta_z}\right)}_{=:-\omega_\lambda(\delta_z)}.
\end{aligned}$$

Figure 5.4 shows that $\omega_\lambda(\delta_z)$ is nonnegative and increasing. Moreover, when λ is close to 1, which corresponds to β_v close to 0 (see Figure 5.3), the guaranteed functional decrease of $\omega_\lambda(\delta_z)$ is also close to $\omega(\delta_z)$, which is the guaranteed functional decrease in the case where partial minimization is done exactly. It also means that if we are far from satisfying the stopping criterion (i.e. if $\delta_z \gg \beta_z$), we reduce the optimality gap by a large amount $\omega_\lambda(\delta_z)$. The number of outer iterations is therefore bounded by

$$N \leq \frac{F(z_0, v_0) - F(z^*, v^*)}{\omega_\lambda(\beta_z)},$$

where $y^* = (z^*, v^*)$ is the optimal solution for (AC).

When the algorithm stops at some point $(\bar{z}, \bar{v})$, we have by construction $\delta_z \leq \beta_z$ and also $\delta_v \leq \beta_v$. That implies according to Theorem 5.3.4

$$\delta_y = \sqrt{\delta_z^2 + \delta_v^2} \leq \sqrt{\beta_z^2 + \beta_v^2} = \sqrt{\frac{1}{2}\beta_y^2 + \frac{1}{2}\beta_y^2} = \beta_y.$$

Let us conclude with a bound on the number of inner iterations, i.e. the

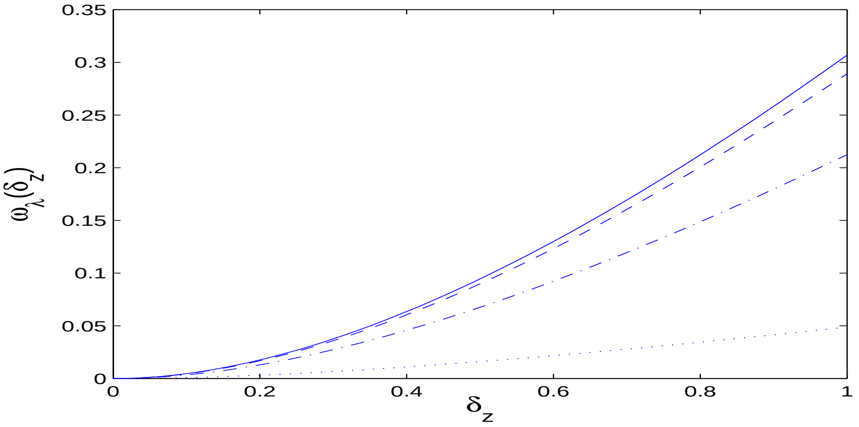


Figure 5.4: Graph of $\omega_\lambda(\delta_z)$ for different values of λ. solid: $\lambda = 1$, dashed: $\lambda = .8$, dash-dot: $\lambda = .5$, dotted: $\lambda = .1$.

number of steps that are needed to generate a point close to the surface of partial minimizers $v(z)$, after one outer step $z^+ = z + \alpha\,\Delta z$.

Let $0 < \beta_v < \frac{1}{2}$ and denote $\gamma = \omega_*(\beta_v) < 1 - \log(2)$ and $r = \omega'_*(\beta_v)$. Then, using Theorem 2.3.4, we get that *before* each z-step it holds

$$F(z, \bar{v}) - \varphi(z) \leq \gamma.$$

Nesterov [50] has shown that, given a point $\bar{v}$ close to $v(z)$, one can update z and v in a such a way that the new v-iterate v^+ is not too far from the new partial minimizer $v(z^+)$, where z^+ is the new z-iterate. Let us formally recall this result.

Theorem 5.3.6. *Let* $(z, \bar{v}) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$ *such that*

$$F(z, \bar{v}) - \varphi(z) \leq \gamma,$$

where $\gamma < 1 - \log(2)$. *Let* $\Delta z \in \mathbb{R}^{n_1}$ *be any direction such that*

$$||\Delta z||_{\nabla^2 \varphi(z)} \leq \kappa < (1 - \omega^{-1}(\gamma))^3$$

and define $\Delta v = J\,\Delta z$ *and the new iterates* $z^+ = z + \Delta z$ *and* $v^+ = \bar{v} + \Delta v$. *Then we have*

$$F(z^+, v^+) - \varphi(z^+) \leq \bar{\gamma},$$

with $\bar{\gamma} = \omega_* \left(\frac{\kappa}{[1-\omega^{-1}(\gamma)]^2} + \omega^{-1}(\gamma) \right)$.

Proof. [50, Theorem 5]. □

Let us apply Theorem 5.3.6 to our setting. We mentioned above that $\beta_v < \frac{1}{2}$ ensures that $\gamma = \omega_*(\beta_v) < 1 - \log(2)$. Let $0 < \kappa < (1 - \omega^{-1}(\gamma))^3$ (for example we can choose $\tau \in (0, 1)$ and define $\kappa = \tau \cdot (1 - \omega^{-1}(\gamma))^3$). As particular direction we are taking the Newton direction Δz from (5.41).

It remains to ensure that

$$||\alpha \, \Delta z||_{\nabla^2 \varphi(z)} \leq \kappa. \tag{5.46}$$

In view of Theorem 2.3.4, we have

$$||\bar{v} - v(z)||_{\bar{v}} \leq \omega_*'(\beta_v) = r.$$

We do not have access to the Hessian $\nabla^2 \varphi(z)$, but using Lemma 5.3.3, we get the bound $||\Delta z||_{\nabla^2 \varphi(z)} \leq \frac{1}{1-r} ||\Delta z||_H = \frac{\delta_z}{1-r}$. We conclude that (5.46) is satisfied if

$$\alpha \, \frac{\delta_z}{1-r} \leq \kappa$$

and since $r = \frac{\beta_v}{1-\beta_v}$ it is necessary to ensure

$$\alpha \leq \frac{(1-r)\kappa}{\delta_z} = \frac{\kappa(1-2\beta_v)}{\delta_z(1-\beta_v)}.$$

This is in particular satisfied for $\alpha \leq \frac{\kappa(1-2\beta_v)}{(\delta_z+1)(1-\beta_v)}$. That means according to Theorem 2.2.24 (where we only enter the first stage of the damped Newton method) that we need no more than

$$N \leq \frac{\bar{\gamma} - \gamma}{\omega(\beta_v)}$$

damped Newton steps in each inner loop. Thus, we have proved the following theorem.

Theorem 5.3.7. *Let Assumptions 4 be satisfied. Let $0 < \beta_y < \frac{1}{\sqrt{2}}$, set $\beta_z = \beta_v = \frac{1}{\sqrt{2}} \beta_y$, and $0 < \kappa < [1 - \omega^{-1}(\omega_*(\beta_v))]^3$. Moreover let $y_0 = (z_0, v_0) \in \operatorname{int} \mathcal{Q}$ such that $G z_0 = g$ and $B v_0 + E z_0 = d$. Then the output of Algorithm 7 is a point $\bar{y} = (\bar{z}, \bar{v}) \in \mathcal{Q}$ such that $E\bar{z} + B\bar{v} = d$ and $G\bar{z} = g$ and*

$$\delta_y = ||\Delta y||_{\bar{y}} \leq \beta_y.$$

The number of outer iterations is bounded by

$$N_{out} \leq \frac{F(y_0) - F(y^*)}{\omega_\lambda(\beta_z)}.$$

Moreover, the number of iterations at each inner loop to generate the approximate partial minimizers $\bar{v}$ is bounded by

$$N_{in} \leq \frac{\bar{\gamma} - \gamma}{\omega(\beta_v)},$$

where $\gamma = \omega_*(\beta_v)$ and $\bar{\gamma} = \omega_* \left(\frac{\kappa}{[1-\omega^{-1}(\gamma)]^2} + \omega^{-1}(\gamma) \right)$.

Remark 5.3.8. *In practice we can choose more aggressive step lengths for Δz than the one presented in the algorithm ($\alpha = \frac{\kappa(1-2\beta_v)}{(1+\delta_z)(1-\beta_v)}$). But in that case we are not able to guarantee anymore that we can bound the centering steps that are needed to compute a point close to the surface of partial minimizers $v(z)$.*

Remark 5.3.9. *We want to point out that the complexity result is essentially the same as for the standard case (compare Theorem 2.3.7), except for the quadratically convergent phase that we could not establish here. The number of iterations depends essentially on the initial optimality gap $F(y_0) - F(y^*)$ that will be reduced in each iteration by a constant which is slightly worse than the standard one (see Figure 5.4).*

The number of iterations to generate an approximation for the partial minimizers is a constant that only depends on β_v (which is chosen a priori). However, the upper bound on N_{in} is rather pessimistic, in particular when we desire a high accuracy for the partial minimizer, but even for moderate values of β_v. For example for $\beta_v = 0.1$ and $\kappa = 0.9 \cdot [1 - \omega^{-1}(\omega_*(\beta_v))]^3$, the upper bound on N_{in} is greater than 300. In practice, however, we typically observe a number of inner iterations of less than 10.

Theorem 5.3.7 only gives useful upper bounds on the number of outer and inner iterations when the desired accuracy is moderate. If β_y is very close to 0 (which implies by construction that β_z and β_v are also close to 0), then the upper bounds for N_{in} and N_{out} tend to ∞. Moreover, if β_v is close to $\frac{1}{2}$, then λ is close to 0 (because λ is bounded from above by $\phi(\beta_v)$, see Figure 5.3). This, in turn, means that $\omega_\lambda(\beta_z)$ is close to 0 even for large β_z (see Figure 5.4). We want to point out that the choice of the parameters $\beta_v = \beta_z = \frac{1}{\sqrt{2}}\beta_y$ was somewhat arbitrary. Any choice of the parameters $\beta_v = \alpha_v \beta_y$ and $\beta_z = \alpha_z \beta_y$ with α_v and α_z such that $\alpha_v^2 + \alpha_z^2 \leq 1$ would give essentially the same result.

Let us conclude with a further justification of computing good approximate partial minimizers $\bar{v}$.

Remark 5.3.10. *If $\bar{v}$ approximates the exact partial minimizer $v(z)$ well, then $\mathcal{L}_J$ is nearly parallel to the tangent subspace $\mathcal{L}_{v'(z)}$ (see Figure 5.2). Furthermore, if the surface of partial minimizers $v(z)$ (as a function of z) has a low curvature, then $\mathcal{L}_{v'(z)}$ is a good approximation for $v(z)$. By consequence, also $\mathcal{L}_J$ could approximate $v(z)$ well, which means we can do relatively large steps in terms of z before leaving the set $\mathcal{C}$.*

Coordinate-descent-like analytic centering

Another much simpler technique of solving (AC) using approximate partial minimization can be described as follows: for a given $z \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$ we compute its partial minimizer $\bar{v}$ as described in Section 5.3.1. Then we execute one Newton step for minimizing F with respect to z along the subspace $\{(z, v) : v = \bar{v}\}$, i.e. we fix the value of the partial minimizer $\bar{v}$. The corresponding Newton system for minimizing F in terms of z is

$$\begin{bmatrix} \nabla^2_{zz} F(z, \bar{v}) & G^T \\ G & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta z \\ \lambda_z \end{bmatrix} = \begin{bmatrix} -\nabla_z F(z, \bar{v}) \\ 0 \end{bmatrix}. \tag{5.47}$$

After going a Newton step along Δz,

$$z^+ = z + \alpha \Delta z$$

```
Algorithm 8 Analytic centering method via coordinate descent
Input: $G$, $B$, $E$, $g$, $d$ as as in Assumption 4, $F : \mathbb{R}^{n_1+n_2} \to \mathbb{R}$ self-concordant on int $\mathcal{Q}$.
Parameter: $0 < \beta_v < 1$, $0 < \beta_z$.
Initialize: $(z_0, v_0) \in \operatorname{int}\mathcal{Q}$ such that $Gz_0 = g$, $Bv_0 + Ez_0 = d$, $k = 0$.
  loop
    1) starting at $v_k$, go damped Newton steps (5.24) until $\delta_v = ||\Delta v||_v \leq \beta_v$,
    output: approximate partial minimizer $\bar{v}$
    2) compute the Newton direction $\Delta z$ from (5.47)
    3) define the Newton decrement: $\delta_z := ||\Delta z||_{F''_{zz}(z_k,\bar{v})}$
    if $\delta_z < \beta_z$ then
      RETURN
    end if
    4) set step size $\alpha = \frac{1}{1+\delta_z}$
    5) update $z_{k+1} = z_k + \alpha\,\Delta z$ and keep $v_{k+1} = \bar{v}$
    6) $k = k + 1$
  end loop
```

with a suitable step size α, we compute again the approximate partial minimizer for the new point z^+, and so on. The method is summarized in Algorithm 8.

As a first step we have to guarantee that a damped Newton step in the z- space yields a decrease in the function value, even if the partial minimization is done only approximately.

Lemma 5.3.11. *If $\beta_v < \omega_*^{-1}(\omega(\beta_z))$, then as long as the above algorithm runs, the outer iterates z generate a strictly monotonically decreasing sequence in the implicit barrier with at least a constant functional decrease of $\tau = \omega(\beta_z) - \omega_*(\beta_v) > 0$, i.e.*

$$\varphi(z^+) \leq \varphi(z) - \tau.$$

Proof. At iteration k, let $z = z_k$ with approximate partial minimizer $\bar{v}$. Then in view of Theorem 2.3.6 and Theorem 2.3.4 we have the following chain of inequalities:

$$\begin{aligned}\varphi(z^+) = F(z^+, v(z^+)) \leq F(z^+, \bar{v}) &\leq F(z, \bar{v}) - \omega(\beta_z) \\ &\leq F(z, v(z)) + \omega_*(\beta_v) - \omega(\beta_z) \\ &= \varphi(z) - \tau.\end{aligned}$$

□

The apparent advantage of Algorithm 8 is its simplicity. We minimize F coordinatewise: first we fix z and minimize with respect to v (partial minimization), then we fix the partial minimizer $\bar{v}$ and compute one Newton step in terms of z.

However, since the Newton directions in terms of z are defined by system (5.47), we cannot use Corollary 5.3.5 to relate δ_z (in combination with δ_v) to the Newton decrement δ_y for the full analytic centering problem (AC). Therefore one cannot provide bounds for the quality of the solution that we obtain when the algorithm stops. That means we cannot guarantee polynomial complexity of the method, which is why we reject this coordinate-descent approach.

To illustrate this claim, let us consider the following example. For $\tau \in [0, 1)$ we define the degenerate simplex

$$\mathcal{C}_\tau = \{(z, v) : z \geq 0, z + v \leq 1, \tau(1 - z) \leq v\}.$$

For $\tau = 0$ the set $\mathcal{C}_0$ is the standard two-dimensional simplex. It is well-known that a 3-self-concordant barrier for $\mathcal{C}_\tau$ is given by

$$F_\tau(z, v) = -\log(z) - \log(1 - z - v) - \log(\tau(z - 1) + v).$$

In this particular example the surface of partial minimizers can be easily computed by solving the optimality conditions $F'_v(z, v) = 0$ with respect to v. One obtains then $v(z) = \frac{1+\tau}{2}(1 - z)$, which is in fact simply a straight line. The set $\mathcal{C}_\tau$ with the level curves of the barrier F_τ and the set of partial minimizers is illustrated in Figure 5.5. We will show now that for points on the curve $v(z)$ the Newton decrement for the coordinate descent method goes to 0, as $\tau \to 1$ even though $(z, v(z))$ is far from the analytic center of $\mathcal{C}_\tau$. Let us introduce some notation. In the following example we write $F(z, v) = F_\tau(z, v)$. Furthermore, for $\tau \in [0, 1)$ we denote by $\delta_z^{CD}(\tau)$ the Newton decrement for the coordinate descent method

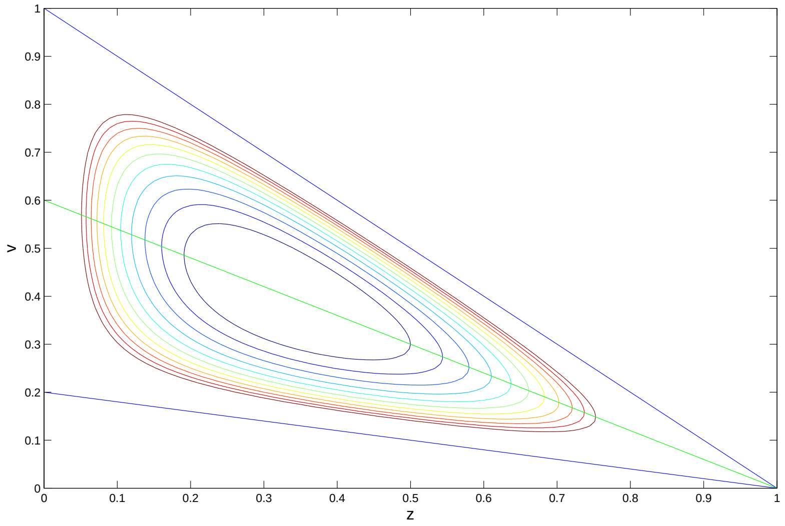


Figure 5.5: $\mathcal{C}_\tau = \{(z, v) : z \geq 0, z + v \leq 1, \tau(1 - z) \leq v\}$ with level curves of $F_\tau(z, v) = -\log(z) - \log(1 - z - v) - \log(\tau(z - 1) + v)$ for $\tau = 0.2$. The intermediate line indicates the set of partial minimizers $v(z)$.

(Algorithm 8) at some fixed point $(z, v(z))$, i.e.

$$\left(\delta_z^{CD}(\tau)\right)^2 = ||\Delta z||^2_{F''_{zz}(z,v(z))} = F'_z(F''_{zz})^{-1}F'_z = \frac{(F'_z)^2}{F''_{zz}}.$$

The gradient and Hessian of F with respect to z can be computed as

$$\begin{aligned} F'_z(z,v) &= -\frac{1}{z} + \frac{1}{1-z-v} + \frac{\tau}{\tau(z-1)+v} \\ F''_{zz}(z,v) &= \frac{1}{z^2} + \frac{1}{(1-z-v)^2} + \frac{\tau^2}{(\tau(z-1)+v)^2}. \end{aligned}$$

Using the fact that v is on the surface $v(z)$, i.e. $v = \frac{1+\tau}{2}(1-z)$ we get

$$\begin{aligned} F'_z(z,v(z)) &= -\frac{1}{z} + \frac{1}{1-z-v(z)} - \frac{\tau}{\tau(z-1)+v(z)} \\ &= -\frac{1}{z} + \frac{1}{(1-z)(1-\frac{1+\tau}{2})} - \frac{\tau}{(1-z)(1-\frac{1+\tau}{2})} \\ &= -\frac{1}{z} + \frac{2}{(1-z)(1-\tau)} - \frac{2\tau}{(1-z)(1-\tau)} \\ &= -\frac{1}{z} + \frac{2}{1-z}, \end{aligned}$$

which means that $F'_z(z,v(z))$ does *not* depend on τ. However, the Hessian F''_{zz} at $(z,v(z))$ can be computed as

$$\begin{aligned} F''_{zz}(z,v(z)) &= \frac{1}{z^2} + \frac{1}{(1-z-v(z))^2} + \frac{\tau^2}{(\tau(z-1)+v(z))^2} \\ &= -\frac{1}{z^2} + \frac{4}{(1-z)^2(1-\tau)^2} + \frac{4\tau^2}{(1-z)^2(1-\tau)^2} \\ &= -\frac{1}{z^2} + \frac{4(1+\tau^2)}{(1-z)^2(1-\tau)^2}. \end{aligned}$$

One can see that $F''_{zz}(z,v(z)) \to \infty$ as $\tau \to 1$. This implies that $\delta_z^{CD}(\tau) \to 0$ as $\tau \to 1$. Moreover, by optimality conditions we have that $F'_v(z,v(z)) = 0$, which implies that $\delta_v = ||F'_v(z,v(z))||_{F''_{vv}} = 0$.

On the other hand, we see in Figure 5.5 that for $z \neq \frac{1}{3}$ the point $(z,v(z))$ is far from the analytic center $(z^*,v^*) = (1/3, v(1/3))$ of $\mathcal{C}_\tau$. This geometric observation can be supported by computing the Newton decrement δ_y for the problem of minimizing F in terms of both variables z and v, or equivalently, the Newton decrement δ_z of Algorithm 7. Indeed, since $\delta_v = 0$ we have in view of Theorem 5.3.4 $\delta_y = \delta_z$, where

$$\delta_z = ||\nabla\psi(z)||_{\nabla^2\psi(z)} = \left(\nabla\psi(z)^T(\nabla^2\psi(z))^{-1}\nabla\psi(z)\right)^{1/2}.$$

Moreover, since $F'_v(z,v(z)) = 0$ by optimality conditions, we get $\nabla\psi(z) = F'_z(z,v(z))$ (see (5.25)). The Hessian becomes in view of (5.26)

$$\nabla^2\psi(z) = F''_{zz}(z,v) - F''_{zv}(z,v)\left(F''_{vv}(z,v)\right)^{-1}F''_{vz}(z,v),$$

where

$$\begin{aligned}
F''_{zv}(z,v) = F''_{vz}(z,v) &= \frac{1}{(1-z-v)^2} + \frac{\tau}{(\tau(z-1)+v)^2},\\
F''_{vv}(z,v) &= \frac{1}{(1-z-v)^2} + \frac{1}{(\tau(z-1)+v)^2}.
\end{aligned}$$

Using again the fact that v is on the straight line $v(z) = \frac{1+\tau}{2}(1-z)$, we can simplify and get

$$\begin{aligned}
F''_{zv}(z,v(z)) = F''_{vz}(z,v(z)) &= \frac{1}{(1-z-v(z))^2} + \frac{\tau}{(\tau(z-1)+v(z))^2},\\
&= \frac{4(1+\tau)}{(1-z)^2(1-\tau)^2},\\
F''_{vv}(z,v(z)) &= \frac{1}{(1-z-v(z))^2} + \frac{1}{(\tau(z-1)+v(z))^2}\\
&= \frac{8}{(1-z)^2(1-\tau)^2}.
\end{aligned}$$

This implies

$$\begin{aligned}
\nabla^2\psi(z) &= -\frac{1}{z^2} + \frac{4(1+\tau^2)}{(1-z)^2(1-\tau)^2} - \frac{(F''_{zv}(z,v(z)))^2}{F''_{vv}(z,v(z))}\\
&= -\frac{1}{z^2} + \frac{4(1+\tau^2)}{(1-z)^2(1-\tau)^2} - \frac{2(1+\tau)^2}{(1-z)^2(1-\tau)^2}\\
&= -\frac{1}{z^2} + \frac{2(1-\tau)^2}{(1-z)^2(1-\tau)^2}\\
&= -\frac{1}{z^2} + \frac{2}{(1-z)^2}.
\end{aligned}$$

We see that also $\nabla^2\psi(z)$ is independent from τ. That means that δ_z is unchanged when $\tau \to 1$. We conclude that for points $(z, v(z))$ on the surface of partial minimizers the Newton decrement δ_y is independent from $\tau \in [0,1)$ (it only depends on the choice of $z \in (0,1)$), while the Newton decrement δ_z^{CD} tends to 0 as $\tau \to 1$. This means we might stop Algorithm 8 at a point (z, v) because δ_v and δ_z^{CD} are small, while in fact (z, v) is far from an optimal solution.

5.3.3 Path-following using approximate partial minimization

Let us come back to the general convex problem (5.4), i.e.

$$\begin{aligned}
&\min a^T z\\
&\text{s.t. } z \in \mathcal{C}\\
&\quad Gz = g.
\end{aligned}$$

Under Assumptions 4 problem (5.4) is equivalent to

$$\begin{aligned} \min\ & a^T z \\ \text{s.t. } & (z,v) \in \mathcal{Q} \\ & Ez + Bv = d \\ & Gz = g. \end{aligned} \tag{5.48}$$

Since we have a self-concordant barrier F for $\mathcal{Q}$ at hand, we can efficiently solve (5.48). In this section we demonstrate how the technique of partial minimization can be embedded in a path-following scheme to solve (5.48).

Note that for any $t \geq 0$ the function

$$f_t(z,v) = t \cdot a^T z + F(z,v)$$

is self-concordant with domain $\operatorname{int} \mathcal{Q}$. Moreover, since the linear term $a^T z$ does not depend on the artificial variables v, for a fixed point $z \in \operatorname{int} \mathcal{C} \bigcap \bar{\mathcal{L}}$ the partial minimization subproblem (PM(z)) is not affected by adding such a linear term. Therefore we directly apply the results from Section 5.3.2 and obtain the same approximate partial minimizer $\bar{v}$ and the same matrices $J(z,\bar{v})$ and $L(z,\bar{v})$ (by solving (5.13)) as in the case of analytic centering (without the linear term).

Following the ideas of Section 5.3.2, let us introduce the function

$$\psi_t(\tilde{z}) := t \cdot a^T \tilde{z} + \psi(\tilde{z}),$$

where ψ is defined just as in Section 5.3.2 relative to the point $(z,\bar{v})$. For the same arguments as above ψ_t is self-concordant. Its derivatives at the point $\tilde{z} = z$ are

$$\begin{aligned} \nabla \psi_t(z) &= t\, a + \nabla \psi(z) = t\, a + h(z,\bar{v}), \\ \nabla^2 \psi_t(z) &= \nabla^2 \psi(z) = H(z,\bar{v}). \end{aligned}$$

Putting these observations together, Theorem 5.3.7 guarantees that for any fixed $t > 0$ one can solve

$$\begin{aligned} \min\ & t \cdot a^T z + F(z,v) \\ \text{s.t. } & Ez + Bv = d \\ & Gz = g \end{aligned} \tag{5.49}$$

efficiently up to any accuracy $\beta_y > 0$ by computing the iterates in the two-level strategy proposed in Algorithm 7.

The only part that changes as compared to Algorithm 7 is the fact that the Newton directions in the outer level do depend on t. Indeed, we consider now the minimization of $\psi_t(\tilde{z})$ over $\bar{\mathcal{L}}$, i.e.

$$\begin{aligned} \min_z\ & t \cdot a^T z + \psi(z) \\ & Gz = g. \end{aligned}$$

Its Newton direction $\Delta z(t)$ at some point z (with partial minimizer $\bar{v}$) can be obtained from the system

$$\begin{bmatrix} H(z,\bar{v}) & G^T \\ G & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta z(t) \\ \lambda_z \end{bmatrix} = \begin{bmatrix} -t\, a - h(z,\bar{v}) \\ 0 \end{bmatrix}. \tag{5.50}$$

Large updates of t

We consider the following update of the duality measure, namely

$$t^+ = \theta \cdot t,$$

where $\theta > 1$. In accordance with the proof of Theorem 2.4.15 the number of updates of t, starting at some $t_0 > 0$ in order to guarantee

$$a^T(z - z^*) \leq \epsilon$$

can be bounded by

$$N_{up} = \mathcal{O}\left(\log\left(\frac{\nu}{t_0 \epsilon}\right)\right)$$

iterations.

We are ready now to present the long-step path-following algorithm using approximate partial minimization (Algorithm 9).

Algorithm 9 Long-step path-following method with partial minimization

Input: G, B, E, g, d as as in Assumption 4, $F : \mathbb{R}^{n_1+n_2} \to \mathbb{R}$ ν-self-concordant barrier for $\mathcal{Q}$.

Parameter: $\epsilon > 0$ desired absolute accuracy, $0 < \beta_y \leq \frac{1}{4}$ the centering accuracy, $\theta > 1$ updating coefficient.

Set $\beta_z = \beta_v = \frac{1}{\sqrt{2}}\beta_y$, choose $0 < \kappa < [1 - \omega^{-1}(\omega_*(\beta_v))]^3$. Define $\lambda = \frac{\kappa(1-\beta_v)}{1-2\beta_v}$ and $\rho(\beta_y, \nu) = \nu + \frac{(\beta_y + \sqrt{\nu})\beta_y}{1-\beta_y}$.

Initialize: $(z_0, v_0) \in \operatorname{int} \mathcal{Q}$ such that $Gz_0 = g$ and $Bv_0 + Ez_0 = d$, $k = 0$, $i = 0$, $t_0 > 0$.

```
while $\epsilon \cdot t_k < \rho(\beta_y, \nu)$ do
  loop
    1) starting at $v_k$, go damped Newton steps (5.24) until $\delta_v = ||\Delta v||_v \leq \beta_v$,
    output: approximate partial minimizer $\bar{v}$
    2) compute $J_k$ and $L_k$ from (5.13)
    3) define $H_k = F''_{zz} + F''_{zv} J_k + E^T L_k$, $h_k = F'_z + J_k^T F'_v$.
    4) compute the Newton direction $\Delta z(t_i)$ from (5.50)
    5) define the Newton decrement: $\delta_z := ||\Delta z(t_i)||_{H_k}$
    if $\delta_z < \beta_z$ then
      BREAK
    end if
    6) set step size $\alpha = \frac{\lambda}{1+\delta_z}$
    7) update $z_{k+1} = z_k + \alpha\, \Delta z(t_i)$ and $v_{k+1} = \bar{v} + \alpha\, J_k\, \Delta z(t_i)$
    8) $k = k + 1$
  end loop
  update $t_{i+1} := \theta\, t_i$
  update $i := i + 1$
end while
```

We get the following complexity result for Algorithm 9.

Theorem 5.3.12. *Let Assumptions 4 be satisfied. Choose* $\epsilon > 0$*. Let* $(z_0, v_0) \in \operatorname{int} \mathcal{Q}$ *such that* $Gz_0 = g$ *and* $Bv_0 + Ez_0 = d$*. Then Algorithm 9 terminates after at most*

$$N = \mathcal{O}\left(\nu \log\left(\frac{\nu}{t_0 \epsilon}\right)\right)$$

iterations with a point $z_N \in \mathcal{C} \bigcap \bar{\mathcal{L}}$ *such that*

$$\langle a, z_N - z^* \rangle \leq \epsilon.$$

Proof. As we outlined above, the number of updates of t to ensure $a^T(z - z^*) \leq \epsilon$ is at most

$$N_{up} = \mathcal{O}\left(\log\left(\frac{\nu}{t_0 \epsilon}\right)\right).$$

Moreover, when updating t to $t^+ = \theta\, t$, the value of the objective function of (5.49) changes to

$$t^+\, a^T z + F(z, v) = f_{t^+}(y).$$

It turns out that the functional difference

$$f_{t^+}(\bar{y}_t) - f_{t^+}(y(t^+)),$$

where $y(t^+)$ is the solution of (5.49) for the new duality measure t^+, can be bounded by $\theta \cdot (\nu + \sqrt{\nu})$ (see Section 2.4). In view of Theorem 5.3.7 the number of z-steps in the affine subspaces $\mathcal{L}_J$ can be bounded by

$$N_{out} \leq \frac{\theta \cdot (\nu + \sqrt{\nu})}{\omega_{\lambda(\beta_z)}},$$

while the number of inner iterations to generate approximations to the partial minimizers $v(z)$ can be bounded by

$$N_{in} \leq \frac{\bar{\gamma} - \omega_*(\beta_v)}{\omega(\beta_v)},$$

(see Theorem 5.3.7), which is a constant that does not depend on the problem size. It only depends on the choice of the accuracy β_v of the partial minimization subproblem (PM(z)).

We conclude that the total number of iterations is

$$N = \mathcal{O}\left(\nu \log\left(\frac{\nu}{t_0 \epsilon}\right)\right).$$

□

We see that the complexity result in Theorem 5.3.12 is the same as the one in Theorem 2.4.15. However, it is important to point out that the constants are worse than in the standard case, where for example the number of iterations N_{in} to generate an approximate partial minimizer $\bar{v}$ is a constant that does not appear in the complexity estimate. This constant could be rather large (see Remark 5.3.9). On the other hand, it is possible that the cost of one iteration (consisting of the computation of the approximate partial minimizer $\bar{v}$ and the Newton direction $\Delta z(t)$) is cheaper than computing directly the full Newton direction Δy (see Lemma 5.3.13). It could additionally or alternatively provide a better direction towards the current target point $y(t)$ on the central path (see Remark 5.3.10).

5.3.4 Partial minimization in primal-dual framework

In this section we show how the technique of partial minimization can be embedded into a primal-dual framework. Algorithm 9 is designed to solve convex problems of the form (5.48), i.e.

$$\begin{aligned} \min a^T z& \\ \text{s.t. } (z,v) &\in \mathcal{Q} \\ Ez + Bv &= d \\ Gz &= g. \end{aligned}$$

where $\mathcal{Q}$ is a full-dimensional closed convex set with ν-self-concordant barrier F. The matrices B and E are assumed to have full row rank (see Assumption 4).

In order to apply a modified version of Algorithm 5 to (5.48), we have to cast this problem in dual conic form

$$\begin{aligned} \max_{y,s} b^T y& \\ s + A^T y &= c, \\ A_f^T y &= c_f, \\ s &\in \mathcal{K}^*, \end{aligned} \tag{D}$$

where $\mathcal{K}^*$ is a proper cone, $[A, A_f]$ has full row rank, A_f has full column rank, and the dual feasible set $\{y : c - A^T y \in \mathcal{K}^*, A_f^T y = c_f\}$ does not contain a straight line.

If we denote $y = (z, v)$ it is clear that $s + A^T y = c$ and $s \in \mathcal{K}^*$ is equivalent to $c - A^T y \in \mathcal{K}^*$. Let us define

$$b = \begin{bmatrix} -a \\ 0 \end{bmatrix} \in \mathbb{R}^{n_1+n_2}$$

and

$$A_f = \begin{bmatrix} E^T & G^T \\ B^T & 0 \end{bmatrix} \in \mathbb{R}^{n_1+n_2, m_1+m_2}, \qquad c_f = \begin{bmatrix} d \\ g \end{bmatrix}.$$

It is clear that (5.48) can be brought in dual form (D) if we find $c \in \mathbb{R}^n$ and $A \in \mathbb{R}^{m,n}$ (for $m = n_1 + n_2$ and some n) such that $[A, A_f]$ has full row rank and $y = (z, v) \in \mathcal{Q}$ if and only if $c - A^T y \in \mathcal{K}^*$, for some proper cone $\mathcal{K}^*$ (note that by Assumption 4 we have that the feasible set of (5.48) does not contain a straight line). Let us assume that such a dual reformulation is possible, i.e. that we can find A and c as above. Examples of such conic reformulations of convex problems were presented in Section 4.4

Let us recall here the main steps of the primal-dual predictor-corrector method (Algorithm 5). These are

(A) the computation of a point $\bar{y}_t$ close to the dual central path,

(B) the primal-dual lifting (2.54),

(C) the computation of the primal-dual affine-scaling direction (2.67),

(D) the update of the duality measure t.

For step (A) we can use the damped Newton method with partial minimization (Algorithm 7). Theorem 5.3.7 guarantees that we can solve the centering problem efficiently as long as the desired accuracy β_y is not too high. However, for the primal-dual lifting (B) we do not only need a point close to the dual central path $\bar{y}_t$, but also the *full* Newton direction Δy and the multipliers λ_1 and λ_2 (see (2.54)). This means after having computed $\bar{y}_t = (\bar{z}_t, \bar{v}_t)$ close to a point $y(t)$ on the dual central path, we have to solve *once* the full Newton system (2.52) in order to have the ingredients for the primal-dual lifting. Similarly, for the primal-dual affine-scaling direction (C) it is not clear if or how we can make use of partial minimization. Instead, we need to solve (2.67) just like in Algorithm 5.

This means that the embedding of partial minimization in Algorithm 5 is only possible to some extent. For the computation of the primal-dual central point (B) and the primal-dual affine-scaling direction partial minimization can possibly not be applied.

5.3.5 Cost per iteration

Let us analyze now the numerical cost of using partial minimization. We recall here that the original motivation of partial minimization was the reduction of the numerical cost of Newton's method (embedded into an interior-point framework). We showed in the previous sections that the complexity of such a path-following method with approximate partial minimization does have essentially the same complexity as in the standard setting where no partial minimization is used.

In the analysis we restrict ourselves to the analytic centering problem (AC). We will estimate the numerical cost of one iteration of Algorithm 7. To simplify the analysis, let us make the following assumptions:

- $F^{''}_{vv}$ is positive definite and diagonal,
- B has full row rank and at most n_2 nonzeros.

In fact these two assumptions are not too restrictive. The first assumption is valid whenever the barrier F is separable in terms of the variables v. This is the case for example when the convex set $\mathcal{C}$ is decomposed into its elementary convex building blocks. Then we would introduce one modelling variable at a time for each elementary block. This would mean that the elementary convex sets do not share modelling variables. The second assumption is reasonable, as we will see in Section 5.4. It means that each artificial variable v_i appears at most once in the linear equality constraints.

The two assumptions above imply (a) that $F^{''}_{vv}$ is easy to invert (it only involves $2n_2$ flops, see Section 2.3.3) and (b) $B(F^{''}_{vv})B^T$ is positive definite and diagonal, and therefore also easy to invert (see again Section 2.3.3).

We see that the numerically dominating computations in Algorithm 7 are the following 5 operations.

1. **Computation of the approximate partial minimizer $\bar{v}$.** In Theorem 5.3.7 we established an upper bound on the number of damped Newton steps to compute $\bar{v}$. But as we mentioned in Remark 5.3.9 and as we will see in Section 5.4, this upper bound is rather pessimistic. Let us denote the number of iterations to solve the partial minimization subproblem (PM(z)) by N_{PM}. In view of Section 2.3.3, the complexity of computing $\bar{v}$ is
$$c_{PM} = N_{PM}\,(8n_2 + m_1) \text{ flops.}$$

2. **The computation of J and L.** J and L can be computed by solving (5.13). In view of Section 2.3.3, the complexity of this operation is
$$c_{J,L} = 3n_2 + 5n_1n_2 + 2n_1m_1 \text{ flops.}$$

3. **Building of H and h.** The gradient h is defined by (5.25). It requires the multiplication $J^T F'_v$ ($2n_1n_2$ flops) and the addition of this product with the vector F'_z (n_1 flops). The Hessian H is defined by (5.26), which requires the multiplication $F''_{zv}J$ ($2n_1^2n_2$ flops), the multiplication E^TL ($2n_1^2m_1$ flops) and the addition of 3 matrices of size $n_1 \times n_1$ ($2n_1^2$ flops). In total this yields
$$c_{H,h} = 2n_1^2(n_2 + m_1 + 1) + n_1(2n_2 + 1) \text{ flops.}$$

4. **Computation of Δz.** The direction Δz can be obtained by solving (5.41). This is a general KKT system with positive definite Hessian block H of size $n_1 \times n_1$ and m_2 linear equality constraints. Its complexity is given by (2.28), i.e.
$$c_{\Delta z} = \frac{1}{3}(n_1 + m_2)^3 + 2(n_1 + m_2)^2 + n_1^2m_2 + n_1m_2^2 + n_1 \text{ flops.}$$

5. **Computation of Δv.** The direction Δv is defined by $\Delta v = J\,\Delta z$. The complexity of this operation is
$$c_{\Delta v} = 2n_1n_2 \text{ flops.}$$

If we sum all the operations we get a total complexity per iteration in Algorithm 7 of

$$N_{PM}\,(8n_2 + m_1) + \frac{1}{3}(n_1 + m_2)^3 + 2(n_1 + m_2)^2 + n_1^2(2n_2 + 2m_1 + m_2 + 2) \\ + n_1(9n_2 + 2m_1 + m_2^2 + 2) + 3n_2 \quad (5.51)$$

floating-point operations.

Note that the above value is only a rough estimation of the complexity of one iteration of Algorithm 7. We are not considering the cost of

1. computing the partial derivatives of the barrier F (this depends on the concrete barrier we have chosen),

2. evaluating the barrier. The evaluation of the barrier will be needed if a line search is used to compute a better step size parameter than the one proposed in step 6 of Algorithm 7. One could for example (starting at the safe value $\alpha = \frac{\lambda}{1+\delta_z}$) gradually increase the step size parameter to find an approximate minimizer of the barrier F along the direction $(\Delta z, J\Delta z)$. This procedure would require the evaluation of F along this direction at several points. In a similar fashion we could employ a line search procedure to compute a step size for the partial minimization subproblem (PM(z)) that is larger than the one that guarantees a certain decrease in the function value, which is $\alpha = \frac{1}{1+\delta_v}$.

   From the theoretical point of view a line search is not necessary, but in practical implementations it is highly recommended in order to improve the overall efficiency of the method. However, it is not possible to bound a priori the number of times the barrier has to be evaluated in such a line search procedure.

Comparison to standard Newton method

Let us compare now the complexity (5.51) of one iteration of Algorithm 7 with the complexity of one iteration of a standard damped Newton method (Algorithm 2) for solving the analytic centering problem (AC). The computationally most expensive operation of Algorithm 2 is the computation of the Newton direction from (5.39). In the proof of Corollary 5.3.5 we saw that solving (5.39) involves solving (5.45) which is exactly the fourth item in the above list of operations ("Computation of Δz"). But this operation implicitly requires the computation of H, h, J and L. This means both Newton methods (with and without partial minimization) share operations 2, 3 and 4 in the above list. The difference is the presence of operation 1 (computation of the partial minimizer $\bar{v}$) in the case of Newton method with partial minimization, and the different v-steps.

We want to stress here that both methods generate different iterates even though the complexities of both algorithms are essentially the same. In order to see this, let $y_k = (z_k, v_k) \in \operatorname{int} \mathcal{Q} \bigcap \mathcal{L}$. Then Algorithm 2 computes directly the full Newton direction $\Delta y = (\Delta y_1, \Delta y_2)$ at y_k. On the other hand, Algorithm 7 first computes an approximate partial minimizer $\bar{v}$ for z_k (and therefore changes the v-components), and then the directions Δz and Δv are computed as in operation 4 and 5 as described above. Even if no partial minimization steps are necessary (i.e. $N_{PM} = 0$) both methods are not the same. In view of the proof of Corollary 5.3.5 the Newton direction Δz in Algorithm 7 is then equal to Δy_1 (that is, the first n_1 components of the full Newton direction Δy). However, Δv is given by $J\,\Delta z$, while Δy_2 is defined as (5.42).

Let us compute now the additional numerical cost that arises when computing Δy_2. We see that it requires first the computation of λ_1 from (5.44), i.e.

$$\lambda_1 = \left(B(F''_{vv})^{-1}B^T\right)^{-1} [M\,\Delta y_1 - B(F''_{vv})^{-1}F'_v],$$

where $M \in \mathbb{R}^{m_1,n_1}$ is defined as in the proof of Lemma 5.3.2. Note that the matrix $\left(B(F''_{vv})^{-1}B^T\right)^{-1} M$ is equal to L (see (5.32)) The complexity of computing λ_1 reduces to the multiplication $L\,\Delta y_1$ ($2n_1m_1$ flops), the diagonal scaling $(F_{vv}^{''})^{-1}F_v'$ (n_2 flops), the multiplication of the latter vector from the left with B^T (n_2 flops, since B has at most n_2 nonzeros), diagonal scaling of the last vector with $\left(B(F_{vv}^{''})^{-1}B^T\right)^{-1}$ (m_1 flops) and the sum of 2 vectors of size m_1 (m_1 flops).

Once λ_1 is computed, we plug it together with Δy_1 in (5.42), that is

$$\Delta y_2 = -(F_{vv}^{''})^{-1}\,[F_v' + F_{vz}^{''}\Delta y_1 + B^T\lambda_1].$$

The matrices $(F_{vv}^{''})^{-1}F_{vz}^{''}$ and $(F_{vv}^{''})^{-1}B^T$ are already computed in the process of defining L (see (5.32)). That means the complexity of computing Δy_2 reduces to the multiplications of $(F_{vv}^{''})^{-1}F_{vz}^{''}$ with Δy_1 ($2n_1n_2$ flops) and $(F_{vv}^{''})^{-1}B^T$ with λ_1 (n_2 flops). Finally, we have to sum 3 vectors of size n_2 ($2n_2$ flops).

That means the computational cost of computing Δy_2 is

$$2n_1(n_2+m_1) + 5n_2 + 2m_1.$$

We have essentially proved the following Lemma.

Lemma 5.3.13. *If in Algorithm 7 no partial minimization steps are needed (i.e. $\delta_v \leq \beta_v$ at the beginning of the current iteration), then one iteration of Algorithm 7 is cheaper than one iteration of Algorithm 2 by*

$$2n_1m_1 + 5n_2 + 2m_1$$

floating-point operations.

Proof. If no partial minimization steps are needed in Algorithm 7, then $N_{PM} = 0$. The only difference between both algorithms with respect to the computational effort is the computation of Δv and Δy_2, respectively. The first costs $2n_1n_2$ flops (see above), the latter $(2n_1(n_2+m_1)+5n_2+2m_1)$ flops which proves the result. □

If we compare the improvement in Lemma 5.3.13 with the total complexity of one iteration (5.51), then we see that the *relative* improvement is rather small, as the total complexity depends cubically on the sum $n_1 + m_2$.

If now some partial minimization steps are needed (because $\delta_v > \beta_v$), then one iteration of Algorithm 7 is numerically cheaper than one iteration of Algorithm 2 only if

$$N_{PM}(8n_2 + m_1) < 2n_1m_1 + 5n_2 + 2m_1, \tag{5.52}$$

where $N_{PM} \neq 0$ is the number of partial minimization steps.

We see that (5.52) cannot be true if n_2 is much larger than n_1 and m_1. Indeed, if $n_2 = \mathcal{O}(n_1 \cdot m_1)$, then (5.52) requires essentially that $N_{PM} < 1$. This means that one standard Newton iteration is cheaper than the one iteration of Algorithm 7. However, for example for $n_1 = n_2 = m_1 = 100$ we find that (5.52) is true if $N_{PM} < 23$.

On the other hand, if N_{PM} is large at iteration k, it means that v_k is far from the partial minimizer $v(z_k)$. Therefore we have improved the function value of the objective by going from (z_k, v_k) to $(z_k, \bar{v})$ by a constant of at least $N_{PM}\omega(\beta_v)$. Moreover, this re-centering justifies hope that we can do large steps in terms of the z-variables (see Remark 5.3.10).

5.3.6 Choosing κ

In this section we are going to propose a reasonable choice for the parameter κ in Algorithm 7. In view of Theorem 5.3.7 we have that the number of partial minimization steps (inner iterations) is bounded by

$$N_{in} \le \frac{\bar{\gamma} - \gamma}{\omega(\beta_v)},$$

where $\gamma = \omega_*(\beta_v)$ and $\bar{\gamma} = \omega_*\left(\frac{\kappa}{[1-\omega^{-1}(\gamma)]^2} + \omega^{-1}(\gamma)\right)$. Moreover, the number of main iterations (outer iterations) is bounded by

$$N_{out} \le \frac{F(y_0) - F(y^*)}{\omega_\lambda(\beta_z)}$$

where

$$\omega_\lambda(\beta_z) = -\lambda\beta_z + \log\left(1 + \frac{\lambda\beta_z}{1 + (1-\lambda)\beta_z}\right)$$

and $\lambda = \frac{\kappa(1-2\beta_v)}{1-\beta_v}$.

We have to choose κ such that

$$0 < \kappa < (1 - \omega^{-1}(\gamma))^3.$$

It is clear that we can write $\kappa = \tau \cdot (1 - \omega^{-1}(\gamma))^3$ for some $\tau \in (0, 1)$.

Then it follows

$$\bar{\gamma} = \omega_*\left(\frac{\kappa}{[1-\omega^{-1}(\gamma)]^2} + \omega^{-1}(\gamma)\right) = \omega_*(\tau + (1-\tau)\omega^{-1}(\gamma)),$$

in other words the argument of the last term is a convex combination of 1 and $\omega^{-1}(\gamma)$. The last term is strictly less than 1 because $\gamma < \omega(1) = 1 - \log(2)$. We see that if τ is close to 1, then the argument tends to 1 too, which means that $\bar{\gamma}$ tends to ∞.

On the other hand, if τ is close to 0, then also κ and λ are close to 0. This implies that $\omega_\lambda(\beta_z)$ is close to 0 even for large β_z (see Figure 5.4). In other words, we have a trade-off between the number of inner and outer iterations when choosing $\tau \in (0, 1)$. Let us consider therefore the total cost which is given by

$$N_{out}(c_{out} + N_{in}\, c_{in}),$$

where c_{in} denotes the cost of one partial minimization step and c_{out} denotes the computational cost of one outer iteration (without the cost of computing the partial minimizer $\bar{v}$). In Section 5.3.5 we have computed these values as

$$\begin{aligned} c_{in} &= 8n_2 + m_1, \\ c_{out} &= c_{J,L} + c_{H,h} + c_{\Delta z} + c_{\Delta v} \end{aligned}$$

in terms of the problem parameters n_1, n_2, m_1 and m_2.

For the number of inner and outer iterations we have the upper bounds presented above. Both upper bounds only depend on β_v, β_z, τ and the initial optimality gap $F(y_0) - F(y^*)$, which we consider here as an absolute constant. Let

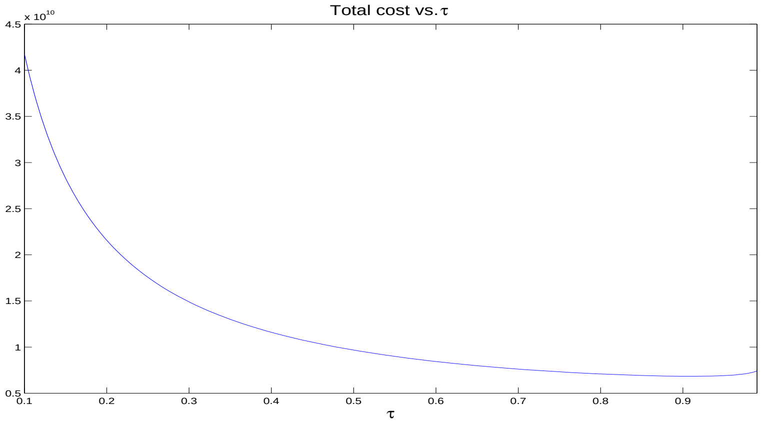


Figure 5.6: Total cost versus τ. Parameter values: $n_1 = m_1 = 100$, $n_2 = 1000$, $m_2 = 0$, $\beta_v = \beta_z = 0.1$.

us fix $\beta_v = \beta_z = 0.1$ and consider the cost function as a function of τ and the problem parameters n_1, n_2, m_1 and m_2.

Figure 5.6 illustrates the upper bound on the cost as a function of τ, for a particular choice of the parameters.

We observe that, when choosing $n_1 = m_1 = 100$, $n_2 = 1000$, $m_2 = 0$ and $\beta_v = \beta_z = 0.1$, the total cost is minimized for a τ-value of approximately 0.9. For different combinations of the parameters, we obtain similar values (around 0.9).

As we mentioned above, the cost function relies on the *upper bounds* for N_{in} and N_{out}. However, we want to stress here that these upper bounds are clearly pessimistic. For example for $\beta_v = \beta_z = 0.1$ and $\tau = 0.9$ we get upper bounds of $N_{in} \leq 320$ and $N_{out} \leq 264$. In numerical experiments we observe a number of inner iterations of no more than 10 (and as we approach the optimal solution it is typically only 1 or 0) and a number of outer iterations of typically no more than 100. But also in these numerical tests we found that $\tau = 0.9$ is a choice that provides a good compromise between the number of inner and outer iterations.

We conclude that a reasonable choice for κ is

$$\kappa = 0.9 \cdot [1 - \omega^{-1}(\omega_*(\beta_v))]^3.$$

5.4 Numerical results

We showed in the previous sections that it is possible to solve convex optimization problems of the form (5.4), for whose feasible set $\mathcal{C}$ we do not have a self-concordant barrier at hand, but where $\mathcal{C}$ is the projection of a higher-dimensional convex set $\mathcal{Q}$ for which we do have a ν-self-concordant barrier. Moreover, we have shown how to compute an optimal solution for the original problem by following a 2-level strategy that does not compute Newton directions for the extended formulation in terms of $\mathcal{Q}$, but instead it follows the surface of partial minimizers to an optimal solution.

Numerical results in Section 4.5 show how Algorithm 4 (and Algorithm 5) can be used to solve convex problems in dual conic form (D), i.e.

$$\begin{aligned} &\max_y b^T y \\ &c - A^T y \in \mathcal{K}^*, \\ &A_f^T y = c_f. \end{aligned} \tag{D}$$

While our methods are reliable and the number of iterations is competitive with the solvers that we compared our implementation with, we also observe that the total computation time increases significantly with the problem size. This effect is essentially due to the decomposition approach. In fact, the original problems in Section 4.5 were of much smaller size than their conic reformulations (D), where all the nonlinear terms are confined to low-dimensional cones. On the other hand, the competing methods (MINOS, SNOPT and KNITRO for the location problems and `gpcvx` for the GP problems) were working directly on the original problems. The negative effect of the conic decomposition was that many artificial variables had to be introduced which increased dramatically the cost per iteration.

In the following two sections we consider again random instances of the generalized location problems and geometric programs. We compare the standard long-step path-following method (Algorithm 4) to its variant that is making use of the technique of partial minimization (Algorithm 9).

As in Section 4.5 we have chosen parameter values of $\epsilon = 10^{-6}$, $\beta_y = \frac{1}{4}$, $\theta = 10$ and for Algorithm 9 the parameter $\kappa = 0.9\,[1 - \omega^{-1}(\omega_*(\beta_v))]^3$. Both methods are making use of a line search on the outer level along the search direction Δy: starting at the safeguard step length ($\frac{1}{1+\delta_y}$, and respectively $\frac{\lambda}{1+\delta_z}$) we gradually increase the step size as long as the objective value of the centering problem improves. For the partial minimization subproblems (PM(z)) that have to be solved in each iteration of Algorithm 9 we use a similar line search procedure.

5.4.1 Location problems

In Section 4.5.1 we considered the unconstrained version of generalized location problems where the distance of a point to given locations (measured in terms of p-norms) has to be minimized. The formulation is the following (compare to (LOC_0) with $R = 0$).

$$\min_{u \in \mathbb{R}^N} \quad \sum_{j=1}^{M} a_j ||u - C_j||_{p_j}. \tag{5.53}$$

We showed that (5.53) can be cast in the standard form (5.4), i.e.

$$\begin{aligned} &\min_z a^T z \\ &\text{s.t. } z \in \mathcal{C} \end{aligned} \tag{5.54}$$

where $z = (u, w) \in \mathbb{R}^{N+M}$, $a = (0, \ldots, 0, a_1, \ldots, a_M)$ and

$$\mathcal{C} = \{(u, w) : ||u - C_j||_{p_j} \leq w_j, j = 1, \ldots, M\}.$$

Note that there are no linear equality constraints present in (5.54), i.e. G and g do not exist. Since p-norm terms in $\mathcal{C}$ can be expressed in terms of power cones, we arrive at the following formulation:

$$\begin{aligned} &\min_{z,v} a^T z \\ &\text{s.t. } (z,v) \in \mathcal{Q}, \\ &Ez + Bv = d, \end{aligned} \tag{5.55}$$

where

$$\mathcal{Q} = \big\{(u,w,v) : (v_{ij}, w_j, u_i - C_{i,j}) \in \mathcal{K}_{\alpha_j}, i = 1,\dots,N, j = 1,\dots,M\big\}.$$

The matrices E and B are given by

$$E = \begin{bmatrix} 0 & -I \end{bmatrix} \in \mathbb{R}^{M,\,(N+M)}$$

and

$$B = \texttt{blkdiag}(\mathbf{1}_N) \in \mathbb{R}^{M,\,N\,M},$$

where $\mathbf{1}_N$ is a row vector of ones of size N (compare (LOC_2) with $R = 0$). The right-hand side vector is $d = 0 \in \mathbb{R}^M$. In the notation of this chapter we have that $n_1 = N + M$ (the number of variables in (5.54)), $n_2 = N\,M$ (the number of artificial variables v), $m_1 = M$ (the number of equality constraints involving z and v) and $m_2 = 0$ (no equality constraints only involving z).

Before we come to the numerical results, let us first verify that indeed Assumption 4 is satisfied.

1. It is clear that $\mathcal{C}$ and $\mathcal{Q}$ are full-dimensional closed convex sets, since they contain interior points (take for example $u = 0$ and v, w with components that are sufficiently large).

2. $B = \texttt{blkdiag}(\mathbf{1}_N) \in \mathbb{R}^{M \times NM}$ has M linearly independent rows. G is not present in this case.

3. There is clearly a bijection between $\mathcal{C}$ and $\mathcal{Q} \bigcap \mathcal{L}$, where $\mathcal{L} = \{(z,v) : Ez + Bv = d\}$ (see e.g. Section 4.1.3).

4. $\mathcal{C}$ does not contain straight lines since it consists of proper (and therefore pointed) cones $\mathcal{P}_p^{(n)}$.

5. If we fix $\bar{z} = (\bar{u}, \bar{w}) \in \operatorname{int} \mathcal{C}$, then $\mathcal{Q} \bigcap \mathcal{L} \bigcap \{(z,v) : z = \bar{z}\}$ can be written as

$$\begin{aligned} & v_{ij}^{\alpha_j} (\bar{w}_j)^{1-\alpha_j} \geq |\bar{u}_i - C_{ij}|,\ i = 1,\dots,N, j = 1,\dots,M, \\ \Leftrightarrow\quad & v_{ij} \geq \underbrace{\left(\frac{|\bar{u}_i - C_{ij}|}{(\bar{w}_j)^{1-\alpha_j}}\right)^{1/\alpha_j}}_{\text{positive constant}},\ i = 1,\dots,N, j = 1,\dots,M, \end{aligned}$$

and

$$\sum_{i=1}^{N} v_{ij} = \bar{w}_j,\ j = 1,\dots,M.$$

That means $\mathcal{Q} \bigcap \mathcal{L} \bigcap \{(z, v) : z = \bar{z}\}$ can be considered as a direct product of the diagonal facets of translated simplices. Therefore it is bounded.

6. It is straightforward to write down a ν-self-concordant barrier F (with $\nu = 3NM$) for $\mathcal{Q}$, using the 3-self-concordant barrier for the power cone $\mathcal{K}_\alpha$.

As in Section 4.5.1 we consider instances of (5.53) with $a_j = 1$ (equal weights on the objective terms), C_j randomly distributed locations in $(0, 1)^N$ for $j = 1, \ldots, M$ and randomly distributed $p_j \in (1, 3), j = 1, \ldots, M$.

In Table 5.1 we illustrate the number of iterations for Algorithm 4 (`D-IPM`) and Algorithm 9 (`D-IPM (PM)`). Note that the iteration numbers for the dual path-following method are not exactly the same as in Table 4.1. This is due to the fact that we were running the code on new random instances and we did not average over 10 runs, as in Table 4.1. One can see that there is a slight improvement in terms of the number of iterations of the path-following method using partial minimization. For small instances the improvement is not significant, but for larger problems the number of iterations reduces by a factor of up to 25%. Note that this effect was not expected. As we have outlined in Section 5.3, the partial minimization technique can be viewed as a restriction of a ν-self-concordant barrier F to a sequence of subspaces $\mathcal{L}_{J_k}$. In accordance with Section 2.4.3 restrictions to subspaces preserve the self-concordance property with the *same* value of the self-concordance parameter ν. The numerical results, however, seem to suggest that these restrictions yield a slight improvement of the self-concordance parameter.

dimension	D-IPM	D-IPM (PM)
$N = 2, M = 10$	24	26 (5)
$N = 2, M = 100$	37	36 (14)
$N = 2, M = 1000$	48	42 (25)
$N = 2, M = 5000$	63	58 (46)
$N = 2, M = 10000$	58	55 (43)
$N = 10, M = 10$	40	38 (26)
$N = 10, M = 50$	46	40 (29)
$N = 10, M = 100$	56	51 (30)
$N = 10, M = 500$	68	58 (42)
$N = 10, M = 1000$	70	64 (70)
$N = 50, M = 50$	68	53 (72)
$N = 50, M = 100$	80	62 (61)
$N = 50, M = 200$	113	68 (69)
$N = 50, M = 400$	101	70 (78)

Table 5.1: LOC - Number of iterations for each solver. In brackets total number of partial minimization steps.

We have argued at the end of Section 2.3.3 that if some partial minimization steps have to be done in some of the outer iterations, then this might result in a higher cost of one iteration of Algorithm 9 as compared to Algorithm 4. On the other hand, this additional computational effort could result in better directions and/or longer steps on the outer level of Algorithm 9. This conjecture cannot be observed directly since both algorithms generate a different sequence of iterates, so comparing these directions (or their lengths) does not have a useful interpretation. However, as we have observed a slight reduction of the number of iterations when using partial minimization, it seem to indicate that the directions $(\Delta z, J\Delta z)$ on the outer level are indeed more favorable than the full Newton directions Δy.

In Theorem 2.3.7 we established an upper bound on the number of partial minimization (p.m.) steps. This bound can be rather high (see Remark 5.3.9). However, we observe in Table 5.1 that the average number of p.m. steps is small as compared to the above-mentioned upper bound (the total number of p.m. steps is comparable to the number of main iterations, i.e. in average there is only one p.m. step needed in each iteration). In the numerical experiments we have observed a number of p.m. steps of less than 10, typically around 3 at the beginning and always 0 at the end of each centering problem. That means at the end of each centering process of Algorithm 9 the computation of each main step $(\Delta z, \Delta v)$ is cheaper than one iteration of Algorithm 4 (see Lemma 5.3.13). Furthermore, if we desire a high accuracy in terms of the centering process (i.e. if we choose β_z small), then we observe a fast reduction in terms of the Newton decrement δ_z. Typically δ_z reduces by a factor or 10 once it is less than 1.

In Table 5.2 we report the total computation time used by both methods. We observe that for the dual method the computation time is slightly better when using partial minimization. This improvement is mainly due to the reduction of the number of iterations. However, the computation times are still significantly higher as compared to the three AMPL solvers which we have tested in Section 4.5.1 (see Table 4.2).

dimension	D-IPM	D-IPM (PM)
$N = 2, M = 10$	0.1	0.1
$N = 2, M = 100$	0.3	0.3
$N = 2, M = 1000$	2.9	2.6
$N = 2, M = 5000$	23.8	21.4
$N = 2, M = 10000$	56.2	53.6
$N = 10, M = 10$	0.2	0.2
$N = 10, M = 50$	0.7	0.6
$N = 10, M = 100$	1.5	1.4
$N = 10, M = 500$	8.7	7.8
$N = 10, M = 1000$	19.1	17.9
$N = 50, M = 50$	4.2	4.4
$N = 50, M = 100$	9.7	10.6
$N = 50, M = 200$	28.9	23.5
$N = 50, M = 400$	55.8	49.9

Table 5.2: LOC - CPU time (in seconds) used by each solver

In order to give an idea of where most of the time is spent in Algorithm 9, we list in Table 5.3 the computationally most expensive operations for one particular instance of size $N = 10$, $M = 1000$. It is surprising that a large part of the computation time is spent only on evaluating the barrier F. The barrier is evaluated in the two line search procedures (one on the outer level and one for the partial minimization subproblem). In the above test case the barrier was evaluated around 600 times by both line search methods.

Operation	Percentage
evaluate F	59.0
compute $\bar{v}$	22.6
compute Δz	5.6
compute J and L	5.1
build H and h	2.9
compute Δv	0.2

Table 5.3: LOC - List of operations where most of the time is spent, $N = 10$, $M = 1000$.

Also the fact that computing $\bar{v}$ is so expensive comes as a surprise, having in mind the results from Section 2.3.3. This can be explained by the fact that most of the computational effort for obtaining $\bar{v}$ is spent on computing the partial derivatives of F. The actual computation of the directions Δv is negligible. That means more than 80% of the computation time is spent only on evaluating F and its derivatives. These operations were not considered in Section 2.3.3. The computation of Δz (which we would expect to be the dominating component) is rather cheap (5.6%). This observation can be explained by the fact that the Hessian H has a special arrow-shape structure and a large diagonal block of size $M = 1000$. Such a particular case is not considered in Section 2.3.3.

5.4.2 GP problems

Let us recall the definition of a geometric program in posynomial form, i.e.

$$\begin{aligned} \min_{x>0} & \sum_{i=1}^{n_0} D_{i,0}^{(\mathrm{pos})}\, x^{\mathbf{K}_{i,0}^{(\mathrm{pos})}} \\ \text{s.t.} & \sum_{i=1}^{n_j} D_{i,j}^{(\mathrm{pos})}\, x^{\mathbf{K}_{i,j}^{(\mathrm{pos})}} \le 1, \;\; j = 1, \dots, M, \\ & e_j^{(\mathrm{mon})}\, x^{K_j^{(\mathrm{mon})}} = 1, \;\; j = 1, \dots, M_{\mathrm{mon}}. \end{aligned} \tag{GP}$$

where $x \in \mathbb{R}^N$ are positive variables, $D_{i,j}^{(\mathrm{pos})}$ and $e_j^{(\mathrm{mon})}$ positive coefficients and $\mathbf{K}_{i,j}^{(\mathrm{pos})}$ and $K_j^{(\mathrm{mon})}$ real exponents.

We have seen in Section 4.4.2 that (GP) can be cast in standard convex form

(5.4) by a change of variables. We obtain

$$\begin{aligned}\min_{z}\ & a^T z \\ \text{s.t. } & z \in \mathcal{C} \\ & Gz = g,\end{aligned} \tag{GP_1}$$

where $z = (u, w) \in \mathbb{R}^{N+1}$, $a = [0, \dots, 0, 1]$,

$$\mathcal{C} = \Bigg\{ (u, w) : \sum_{i=1}^{n_0} \exp\left(u^T \mathbf{K}_{i,0}^{(\mathrm{pos})} + C_{i,0}^{(\mathrm{pos})}\right) \leq w; \\ \sum_{i=1}^{n_j} \exp\left(u^T \mathbf{K}_{i,j}^{(\mathrm{pos})} + C_{i,j}^{(\mathrm{pos})}\right) \leq 1,\ j = 1, \dots, M \Bigg\},$$

$$G = \begin{bmatrix} K^{(\mathrm{mon})} \\ 0 \cdots 0 \end{bmatrix}^T \in \mathbb{R}^{M_{\mathrm{mon}} \times (N+1)} \qquad g = -\log\left(e^{(\mathrm{mon})}\right) \in \mathbb{R}^{M_{\mathrm{mon}}}.$$

As we demonstrated in Section 4.4.2, (GP_1) can be cast in the following decomposed form.

$$\begin{aligned}\min_{z,v}\ & a^T z \\ \text{s.t. } & (z, v) \in \mathcal{Q} \\ Ez + Bv &= d \\ Gz &= g,\end{aligned} \tag{GP_2}$$

where $z = (u, w)$ and

$$\mathcal{Q} = \left\{ (u, w, v) : \exp\left(u^T \mathbf{K}_{i,j}^{(\mathrm{pos})} + C_{i,j}^{(\mathrm{pos})}\right) \leq v_{i,j},\ \ i = 1, \dots, n_j, j = 0, \dots, M \right\}.$$

The linear constraints $Ez + Bv = d$ are given by

$$E = \begin{bmatrix} 0 & \cdots & 0 & -1 \\ & & & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{(M+1) \times (N+1)}, \quad B = \mathtt{blkdiag}(\mathbf{1}_{n_j}) \in \mathbb{R}^{(M+1) \times (\sum_{j=0}^{M} n_j)},$$

where $\mathbf{1}_{n_j}$ is a row vector of ones of size $n_j, j = 0, \dots, M$. The right-hand side vector d is defined as $d = [0, 1, \dots, 1]^T \in \mathbb{R}^{M+1}$ (compare (GP_2)).

In the notation of the chapter we have that $n_1 = N+1$ (the number of variables in (GP_1)), $n_2 = \sum_{j=0}^{M} n_j$ (the number of artificial variables v), $m_1 = M + 1$ (the number of equality constraints involving z and v) and $m_2 = M_{\mathrm{mon}}$ (the number of equality constraints only involving z).

Before we present the numerical results, let us first check again that Assumption 4 is satisfied.

1. Of course $\mathcal{C}$ is not always full-dimensional for all data $\mathbf{K}_{i,j}^{(\text{pos})}$ and $C_{i,j}^{(\text{pos})}$, but the GP generator `mkgp` that we have used to create instances, is generating random instances that do have strictly feasible solutions. Since $\mathcal{C}$ is full-dimensional then so is $\mathcal{Q}$.

2. $B = \texttt{blkdiag}(\mathbf{1}_{n_j}) \in \mathbb{R}^{(M+1)\times(\sum_{j=0}^{M} n_j)}$ has $M+1$ linearly independent rows. G has full row rank since the vectors $K_j^{(\text{mon})}, j = 1, \dots, M_{\text{mon}}$ are assumed to be linearly independent.

3. There is clearly a bijection between $\mathcal{C}$ and $\mathcal{Q} \bigcap \mathcal{L}$, where $\mathcal{L} = \{(z, v) : Ez + Bv = d\}$ (see arguments in Section 4.4.2).

4. To see that $\mathcal{C}$ does not contain straight lines, let us consider a nontrivial direction $(\Delta u, \Delta w)$. If $\Delta w \neq 0$ then it is always possible to find a step size γ such that $w + \gamma\Delta w < 0$, which is not possible. Let therefore $\Delta w = 0$ and assume $\Delta u \neq 0$ (otherwise $(\Delta u, \Delta w) = 0$). Since $\mathbf{K}_{\text{pos}}$ has full row rank, it cannot contain any 0-rows, which means there exists a column $\mathbf{K}_{i,j}^{(\text{pos})}$ such that $\Delta u^T \mathbf{K}_{i,j}^{(\text{pos})} \neq 0$. It is clear that we can find a step size γ such that

   $$(u + \gamma\Delta u)^T \mathbf{K}_{i,j}^{(\text{pos})} + C_{i,j} > \underbrace{\max\{\log(w), 0\}}_{\text{constant}},$$

   which violates one of the inequality constraints. That means $\mathcal{C}$ does not contain straight lines because there is no direction $(\Delta u, \Delta w) \neq 0$ that we can extend to ±∞ without leaving $\mathcal{C}$.

5. If we fix $\bar{z} = (\bar{u}, \bar{w}) \in \operatorname{int} \mathcal{C}$, then $\mathcal{Q} \bigcap \mathcal{L} \bigcap \{(z, v) : z = \bar{z}\}$ can be written as

   $$v_{ij} \geq \underbrace{\exp\left(\bar{u}^T \mathbf{K}_{i,j}^{(\text{pos})} + C_{i,j}^{(\text{pos})}\right)}_{\text{positive constant}}, \; i = 1, \dots, n_j, j = 0, \dots, M,$$

   and

   $$\sum_{i=1}^{n_j} v_{ij} = \begin{cases} w, & j = 0, \\ 1, & j = 1, \dots, M. \end{cases}$$

   That means $\mathcal{Q} \bigcap \mathcal{L} \bigcap \{(z, v) : z = \bar{z}\}$ can be thought of as a direct product of the diagonal facets of translated simplices. Therefore it is bounded.

6. It is straightforward to write down a ν-self-concordant barrier F (with $\nu = 3\sum_{j=0}^{M} n_j$) for $\mathcal{Q}$, using the 3-self-concordant barrier for the exponential cone $\mathcal{K}_{\text{exp}}$.

As in the previous section we investigate the effect of partial minimization on Algorithm 4 with respect to the number of iterations and the total computation time. In order to test our algorithms we consider a family of random GP's generated by `mkgp`, a Matlab function included in `gpcvx`[1] that has originally been written by Lieven Vandenberghe and later been modified by Kwangmoo Koh.

[1] See `http://www.stanford.edu/~boyd/ggplab/gpcvx.pdf`

Our tests have been made on random instances with $n_j = 5, j = 0, \ldots, M$, $M_{\text{mon}} = 5$ for different values of N and M. In Table 5.4 we display the number of iterations for both methods, that is Algorithm 4 (`D-IPM`) and Algorithm 9 (`D-IPM (PM)`). One can see that the use of partial minimization yields a significant reduction in terms of the number of iterations. Note that this effect is not guaranteed by the worst-case complexity result (Theorem 5.3.12). Surprisingly, for increasing value of M, the number of iterations remains constant ($N = 50$) or decreases ($N = 100$, $N = 200$).

dimension	D-IPM	D-IPM (PM)
$N = 50, M = 50$	92	55
$N = 50, M = 100$	237	78
$N = 50, M = 150$	158	68
$N = 100, M = 100$	133	116
$N = 100, M = 200$	172	77
$N = 100, M = 300$	127	72
$N = 100, M = 400$	98	61
$N = 100, M = 500$	246	64
$N = 200, M = 200$	348	169
$N = 200, M = 400$	292	88
$N = 200, M = 600$	105	69
$N = 200, M = 800$	211	78

Table 5.4: GP - Number of iterations used by each solver.

Table 5.5 displays the total computation time used by each method. We see that the computation time for the dual path-following method is lower by a factor of 3 when using partial minimization. This improvement is mainly due to the

dimension	D-IPM	D-IPM (PM)
$N = 50, M = 50$	1.19	0.70
$N = 50, M = 100$	3.79	1.19
$N = 50, M = 150$	3.18	1.25
$N = 100, M = 100$	3.17	2.44
$N = 100, M = 200$	4.97	1.95
$N = 100, M = 300$	4.46	2.20
$N = 100, M = 400$	4.03	2.04
$N = 100, M = 500$	13.15	2.57
$N = 200, M = 200$	13.61	6.35
$N = 200, M = 400$	15.01	4.24
$N = 200, M = 600$	6.49	3.42
$N = 200, M = 800$	14.37	4.55

Table 5.5: GP - CPU time (in seconds) used by each solver.

reduction in the number of iterations (see Table 5.4), but also to the lower cost per iteration as we can see in Table 5.6. If we compare the computation time to GPCVX (see Table 4.5), we observe that the use of partial minimization could close the gap to some extent. However, the dedicated GP solver is still superior with respect to the computation time because GPCVX does not have the additional computational burden of the inner iterations to approximate the partial minimizers.

dimension	D-IPM	D-IPM (PM)	improvement
$N = 50, M = 50$	1.57	1.46	7.4
$N = 50, M = 100$	1.87	1.62	13.5
$N = 50, M = 150$	2.18	1.89	13.4
$N = 100, M = 100$	2.75	2.18	20.6
$N = 100, M = 200$	3.20	2.57	19.7
$N = 100, M = 300$	3.96	3.02	23.6
$N = 100, M = 400$	4.60	3.40	26.1
$N = 100, M = 500$	4.81	3.79	21.3
$N = 200, M = 200$	4.90	3.75	23.3
$N = 200, M = 400$	5.39	4.31	20.0
$N = 200, M = 600$	6.63	5.05	23.8
$N = 200, M = 800$	7.65	6.12	20.0

Table 5.6: GP - CPU time per iteration (in 10^{-2} seconds) used by both methods, and improvement (in %).

Let us have a look again at where most of the time is spent in Algorithm 9. The most expensive operations are listed in Table 5.7, for one random (GP) instance of size $N = 200$, $M = 800$, $M_{\text{mon}} = 5$, $n_j = 5$ for $j = 0, \dots, M$. The cost of evaluating F is still rather high (38.1%), even though it is not as pronounced as in the example of location problems (compare Table 5.3). Note that in this concrete example F had to be evaluated around 750 times. The cost of computing $\bar{v}$ is higher than expected. But the computation time of this operation is again dominated by computations and manipulations of the derivatives of F. The actual

Operation	Percentage
evaluate F	38.1
compute $\bar{v}$	18.6
compute Δz	10.8
build H and h	8.7
compute J and L	5.6
compute Δv	0.3

Table 5.7: GP - List of operations where most of the time is spent, $N = 200$, $M = 800$, $M_{\text{mon}} = 5$, $n_j = 5$ for $j = 0, \dots, M$.

time for obtaining $\bar{v}$ is only 6.8% of the total computation time.

Let us conclude with a comparison of partial minimization applied to location problems (5.53) and geometric programming problems (GP). We observe that partial minimization is more beneficial for (GP): the total computation time is reduced by roughly a factor of 3, while for the location problems we have observed only a very small improvement (see Table 4.2 and Table 4.5). This reduction is in both cases due to the reduced total number of iterations (which is more pronounced in the case of (GP), see Table 5.4 as compared to Table 5.1) and the reduced cost per iteration. The cost per iteration is in both cases dominated by simply evaluating the barrier F and its derivatives (see Table 5.3 and Table 5.7). For (GP) these evaluations seem to be cheaper as compared to location problems.

5.5 Summary

Let us conclude with a summary on partial minimization.

1. **Complexity preserved:** We saw in Theorem 5.3.12 that the global complexity of Algorithm 9 is essentially the same as the complexity of the standard long-step path-following method (Algorithm 4), namely

   $$\mathcal{O}\left(\nu \log\left(\frac{\nu}{t_0\epsilon}\right)\right).$$

   Moreover, as we illustrated in Section 5.3.4 that the partial minimization framework can be embedded in the primal-dual predictor-corrector method (Algorithm 5), at least to some extent.

2. **Numerical results:** The numerical results of Section 5.4 indicate that the embedding of partial minimization into dual interior-point methods is indeed beneficial. For the two problem classes considered (location problems and geometric programs) the number of iterations reduces (compare Table 5.1 and Table 5.4). As a result, the total computation time decreases (compare Table 5.2 and Table 5.5) for the dual path-following method using partial minimization.

3. **Fast convergence:** Even though we cannot establish formally quadratic convergence of the centering problems (Theorem 5.3.7), we do observe fast convergence at the end of each of the centering problems for both problem classes. In particular, the numerical tests seem to indicate that if $\delta_z < 1$, then we can do full steps ($\alpha = 1$) and no centering steps are needed, i.e. $\delta_{v^+} \leq \beta_v$. Moreover, δ_z typically reduces by a factor of 10 once it is less than 1.

4. **Possible improvements:** We observe that a large amount of computation time is spent only on evaluating the barrier (40-60%) for the line search procedures. There are line searches implemented on two levels: (a) for the partial minimization subproblems, (b) for the dual centering problems. As a result, the barrier typically has to be evaluated a couple of hundred times. A possible improvement could be a cheaper way of evaluating the barrier or

approximating the barrier values.

As we have outlined at the beginning of the chapter, the framework of partial minimization is well-suited for parallelization, since we consider problems that are partially separable. It could be possible to improve the performance by separating the partial minimization subproblems. This would mean one could evaluate the barrier and its derivatives in parallel and also compute the Newton directions for each small subproblem simultaneously.

Another improvement could be the use of larger neighborhoods. Algorithm 7 is designed such that in *each* outer iteration we generate an approximate partial minimizer with accuracy $\delta_v \leq \beta_v < \frac{1}{2}$. On the other hand, it might not be necessary to enforce such a high accuracy, especially at the beginning. Therefore, it might prove beneficial to require only $\delta_v \leq \bar{\beta_v}$, where $\bar{\beta_v}$ is large.

5. **Scope:** We illustrate how partial minimization can be applied whenever we want to optimize over a convex set $\mathcal{C}$ that has to be lifted to some higher-dimensional set $\mathcal{Q}$ in order to have a self-concordant barrier at hand. It is clear that the proposed framework is particularly successful when there are many artificial variables v that had to be introduced in the modelling process and if they are sparsely present in the final model. In other words, we need that the subproblems (PM(z)) can be solved efficiently. This is the case for example when the Hessian of F with respect to the artificial variables v is diagonal, but also the matrix B is sparse so that $B(F^{''}_{vv})^{-1}B^T$ is cheaply invertible. This is for example the case when there is exactly one nonzero present in each column of B. Then $B(F^{''}_{vv})^{-1}B^T$ is diagonal to, if $F^{''}_{vv}$ is diagonal (see Section 2.3.3).

CHAPTER 6

Conclusions

We have demonstrated in this thesis that interior-point methods exhibit many very favorable theoretical and practical properties such as polynomial complexity (under some technical assumptions) and reliability of their implementations. However, we have also outlined that there are certain limitations to their usage for practical applications, such as the access to a self-concordant barrier for the feasible set. In view of these drawbacks we have formulated in Chapter 1 three goals, i.e. an algorithmic framework with the following three properties. First, the complexity for solving any particular instance from the chosen problem class should be polynomial in the problem size. Second, the chosen problem class should be sufficiently general. Third, the framework should exploit structure from the original problem in order to improve efficiency.

How we met our goals

We have proposed in Chapter 2 two interior-point methods (a dual and a primal-dual path-following method) for convex problems in conic form. We have shown that these methods exhibit a polynomial complexity provided that a self-concordant barrier for the dual (or primal) cone are at hand. In Chapter 3 we have proved self-concordance of a new barrier for the power cone. This result turns out to be fundamental, in that we have shown in Chapter 4 that the power cone is very versatile since many convex sets and functions are representable using that cone. This is why we have chosen our basic formulation to be a dual conic problem based on the power cone and a limit of the power cone (the so-called exponential cone). Furthermore, we have explicitly computed the dual cones of these two principal cones, with the aim to be able to use the nonsymmetric primal-dual method proposed in Chapter 2. The process of reformulating the original problem in the dual conic form based on the power cone is called *lifting.* Unfortunately, such a lifting has the negative side effect that typically many artificial variables have to be introduced which slows down the solution process due to a higher cost per iteration. This phenomenon was observed in numerical test for two problems classes

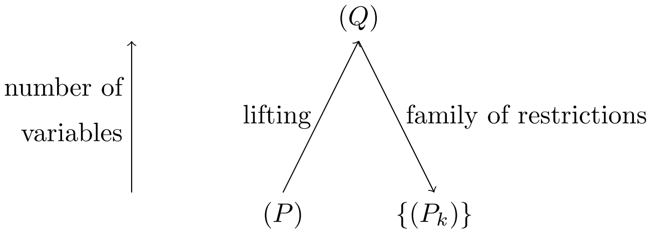


Figure 6.1: Lift and restrict

(generalized location problems and geometric programs). In order to overcome this negative effect of the conic modelling we have proposed in Chapter 5 a new framework, called partial minimization, which can be embedded in interior-point methods. We have proposed a two-level interior-point scheme where in each outer iteration we solve the partial minimization subproblem approximately. Using this approximate solution we define directions where the artificial variables are confined to affine subspaces that are approximately tangent to the surface of partial minimizers. Therefore, this technique effectively removes dependence on the artificial variables. We have shown that polynomial complexity is preserved. Finally, we have demonstrated that this new technique also works in practical implementations, where we have observed a reduction in the cost per iteration, and also in the total number of iterations.

Our strategy can be illustrated in the following way. Given an α-representable optimization problem (P), we denote its α-representation by (Q), which is the dual conic formulation based on the power cone and exponential cone presented in Chapter 4. This lifting is illustrated on the left-hand side in Figure 6.1. As we have argued above, the lifting is necessary in order to profit from the polynomial time algorithms. The right-hand side in Figure 6.1 illustrates the *technique* of partial minimization as a means to compute a solution for (Q) by *not* solving (Q) directly. Instead, we compute search directions for a *family* $\{(P_k)\}$ of lower-dimensional problems, where artificial variables are essentially removed.

Our thesis in the context of convex optimization

Optimization is an important field of applied mathematics with many applications in various domains, ranging from engineering to finance and operations research. In particular convex optimization is very popular because of highly efficient methods which are supported by strong theoretical results. The advantages of convex optimization problems can be classified into passive and active features. Passive features are for example the fact that local optimal solutions that are always also global solutions; or the fact that optimality conditions are also sufficient (and not only necessary). The active features are those that require some additional user expertise, such as the access to efficient methods with guaranteed polynomial complexity; and a rich duality theory that can be used for finding bounds on the optimal value. However, in order to profit from the favorable properties of convexity, it is necessary to ensure that the considered problem is indeed convex. Unfortunately, checking convexity is a highly non-trivial task. Once convexity is established, one profits automatically from its passive features. On the other hand, the active features do not come for free. In order to benefit from them we have to bring the optimization problem in a structured convex form, e.g. a form for which interior-point algorithms are designed; or a form for which a "good" dual problem can be identified.

In the context of interior-point methods structure means the access to self-concordant barriers for the feasible set (and possibly the epigraph of the objective function). In general it is a tedious work to find self-concordant barriers for given convex sets. Recently, we have observed some research activity in that direction, e.g. [47]. In this thesis we have chosen as the structured basic formulation a dual problem based on the power cone, for which we have proved self-concordance of a new barrier. On the other hand, the concept of self-concordance might not be the only paradigm. Indeed, self-concordance is a *global* property that might be too restrictive. For example, it could be sufficient to ensure so-called *local* self-concordance (see [57]) in a neighborhood around the central path, since this is the area where the iterates are confined. Analogously, other concepts might guarantee also polynomial complexity for certain problem classes.

The tasks of formulating and optimization problem, detecting to which problem class(es) it belongs and the transformation into the input format of the solver of choice can be facilitated with the help of modelling languages. These tools can be standalone software packes such as AMPL [19] and GAMS [7], or Matlab toolboxes like CVX [27] and Yalmip [39]. We want to point out here CVX, which is designed for so-called disciplined convex optimization problems (see [26]), i.e. convex problems that can be detected as such. The other three above-mentioned modelling languages are not necessarily restricted to convex optimization. In that sense the purpose of this thesis has some intersection with the concept of CVX, in that we build up a class of *well-behaved* convex optimization problems. The main difference is that CVX is currently restricted to symmetric conic optimization (as it only supports SeDuMi and SDPT3 as solvers). However, in principle CVX could be adapted to support also power cone representable sets (using the concepts presented in Chapter 4). Another difference is that our work is more biased towards the algorithmic side, while CVX entirely focuses on the modelling aspects of convex optimization.

The prerequisite of efficient interior-point methods for convex optimization problems is the reformulation of the original problem into a structured convex form for which interior-point algorithms are available. This extraction of the structure can be considered as active knowledge about the original problem, which in turn can be exploited to increase efficiency. There are several ways how structure can be exploited. For example sparsity in the Hessian can be used to reduce the storage memory and the cost for computing the search direction (see [22]). Similarly, separability (see [8]) can increase efficiency of the algorithms. In this work we have proposed to use the new technique of approximate partial minimization to compute the search directions in a lower-dimensional space at a reduced cost.

Future research

There are several potential future fields of research that can improve our results.

Conjectured self-concordant barriers

In Chapter 3 we formulated three conjectures for self-concordant barriers: an $(n+1)$-self-concordant barrier for the high-dimensional power cone $\mathcal{K}_\alpha^{(n)}$ and for a further a generalization of $\mathcal{K}_\alpha^{(n)}$, and (partial) scaling of the 3-self-concordant barrier for $\mathcal{K}_\alpha$ which results in a self-concordance parameter of $\nu \in [2,3]$ depending on the value of α. We obtained indication for these statements from numerical tests where we checked the self-concordance property for random points in the cones. However, it would be interesting to find an analytic proof of these conjectures.

In the same chapter we computed numerically the universal barrier for the p-cone. The evaluation of the barrier and its derivatives involves the computation of several integrals, which makes this barrier not practical for real applications. Moreover, we found in numerical tests that it should be possible to scale the barrier and obtain a self-concordance parameter of $\nu \in [2,3]$, depending on the value of p. This result is in line with the theory, since for the particular values of $p = 1, 2, \infty$ we know the optimal value of ν, which is also the value that we obtain in our tests. An improvement to this study could be an establishment of a formal proof that supports this observation of the self-concordance parameter.

Dual cones

In Chapter 4 we computed the dual cones of the power cone $\mathcal{K}_\alpha$ and of the exponential cone $\mathcal{K}_{\text{exp}}$. Moreover, we illustrated that $\mathcal{K}_{\text{exp}}$ is in fact the limit of a linear transformation $(\tilde{\mathcal{K}_\alpha})^*$ of the power cone $\mathcal{K}_\alpha$. It would be interesting to extend this study by showing whether a similar limit relation exists on the dual side, i.e. if $\mathcal{K}_{\text{exp}}^*$ is also the limit of $(\tilde{\mathcal{K}_\alpha})^*$.

Newton's method with approximate partial minimization

In Chapter 5 we presented a complexity result for a variant of Newton's method that uses approximate partial minimization. We could successfully show that this method guarantees a constant decrease of the objective function in each iteration (see Theorem 5.3.7). However, we could not establish the quadratically convergent phase of Newton's method close to an optimal solution, even though we do observe in our numerical tests a rapid convergence close to an optimal solution: once the Newton decrement for the outer level is sufficiently small it typically reduces by a factor of 10 in each following outer iteration. Moreover, the number of partial minimization steps is low, in particular close to the overall optimal solution. Finally, we observed a reduction in the number of outer iterations of the overall method. This observation was not covered by the theoretical results and it seems to indicate that partial minimization actually reduces the self-concordance parameter of the original barrier. The aim of a future work could be to theoretically support these observations by analytic proofs.

APPENDIX A

The conic solver

The numerical results in Chapter 4 and Chapter 5 have been achieved with a Matlab[1] implementation of Algorithm 4 and Algorithm 9. The solver can be freely downloaded[2] and installed as described therein.

We are going to give a brief documentation of the solver, its input and output format and a concrete example of a small convex optimization problem whose conic reformulation can be fed into the solver.

A.1 Problem class

The solver supports a primal-dual format that has been presented already in Section 2.5.3, i.e.

$$\begin{aligned} \min_{x,x_f}\ & c^T x + c_f^T x_f \\ Ax + A_f x_f &= b, \\ x &\in \mathcal{K}, \\ x_f &\text{ free}, \end{aligned} \qquad\qquad \begin{aligned} \max_{y,s}\ & b^T y \\ s + A^T y &= c, \\ A_f^T y &= c_f, \\ s &\in \mathcal{K}^*, \end{aligned}$$

where $x, c \in \mathbb{R}^n$, $x_f, c_f \in \mathbb{R}^{n_f}$, $y, b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m,n}$ and $A_f \in \mathbb{R}^{m,n_f}$ and $\mathcal{K}, \mathcal{K}^* \subset \mathbb{R}^n$ are dual to each other and $\mathcal{K}$ (respectively $\mathcal{K}^*$) is the direct product of the three elementary cones

- $\mathcal{K}_\alpha = \left\{ x \in \mathbb{R}_+^2 \times \mathbb{R} : x_1^\alpha \cdot x_2^{1-\alpha} \geq |x_3| \right\}$
- $\mathcal{K}_{\exp} = \mathrm{cl}\left(\left\{ x \in \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R}_{++} : \exp\left(\frac{x_1}{x_3}\right) \leq \frac{x_2}{x_3} \right\} \right)$

[1] Version 7.2.0 (R2006a), see `http://www.mathworks.com`

[2] `http://www.core.ucl.ac.be/~glineur/PowerSolver`

- $\mathbb{R}^{n_l}_+ = \{x : x_i \geq 0, i = 1, \ldots, n_l\}$.

The necessary requirement for the design of polynomial-time interior-point methods is the access to a self-concordant barrier for $\mathcal{K}$ and/or $\mathcal{K}^*$. In Chapter 3 we have presented a $(n+1)$-self-concordant barrier for the high-dimensional power cone $\mathcal{K}^{(n)}_\alpha$. For $n = 2$, we obtain the 3-self-concordant barrier

$$F_\alpha(x) = -\log\left(x_1^{2\alpha} x_2^{2-2\alpha} - x_3^2\right) - (1-\alpha)\log(x_1) - \alpha\log(x_2)$$

for the thee-dimensional power cone $\mathcal{K}_\alpha$. Further, it is well-known that $\mathcal{K}_{\exp}$ admits the 3-self-concordant barrier

$$F_{\exp}(x) = -\log\left(x_3 \cdot \log\left(x_2/x_3\right) - x_1\right) - \log(x_2) - \log(x_3).$$

We denote by n_α the number of power cones $\mathcal{K}_\alpha$ and by n_e the number of exponential cones $\mathcal{K}_{\exp}$. In that notation we have that the primal (respectively the dual) cone becomes

$$\underbrace{\mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_{n_\alpha}}}_{n_\alpha \text{ times}} \times \underbrace{\mathcal{K}_{\exp} \times \cdots \times \mathcal{K}_{\exp}}_{n_e \text{ times}} \times \mathbb{R}^{n_l}_+ \tag{A.1}$$

with the $(3n_\alpha + 3n_e + n_l)$-self-concordant barrier

$$F(x) = \sum_{i=1}^{n_\alpha} F_{\alpha_i}(x_\alpha) + \sum_{i=1}^{n_e} F_{\exp}(x_e) + F_l(x_l),$$

where $x = (x_\alpha, x_e, x_l)$, and $x_\alpha \in \mathbb{R}^{3n_\alpha}$, $x_e \in \mathbb{R}^{3n_e}$ and $x_\in \mathbb{R}^{n_l}_+$. F_l denotes the standard logarithmic barrier for the nonnegative orthant.

In Section 4.3 we have computed the dual cones of $\mathcal{K}_\alpha$ and $\mathcal{K}_{\exp}$. Therefore it is straightforward to write down the dual cone for (A.1) attached with a $(3n_\alpha + 3n_e + n_l)$-self-concordant barrier. Note, however, that (A.1) is not symmetric (unless there are no $\mathcal{K}_\alpha$ and $\mathcal{K}_{\exp}$ constraints present, in that case the conic pair boils down to a pair of linear programs). Moreover, the barriers for (A.1) and its dual are not conjugate.

Let us summarize the dual problem (using the above notation), as it will yield for further illustration. Let A and c be partitioned in the following way

$$\begin{aligned} \bar{A} &= [A_f, A] = [A_f, A_l, A_\alpha, A_e], \\ \bar{c} &= [c_f; c] = [c_f; c_l; c_\alpha; c_e], \end{aligned}$$

where A_l are the first n_l columns of A that correspond to the primal nonnegativity constraints. A_p are the further n_α columns of A that correspond to the primal power cone variables and A_e are the last n_e columns of A that correspond to the exponential cone constraints. Analogously, the primal objective vector c is partitioned into c_l, c_α and c_e. Then we get as the dual problem

$$\begin{aligned}
&\max b^T y\\
&\ s.t.\ c_f - A_f^T\, y \in \{0\} \subset \mathbb{R}^{n_f},\\
&\qquad c_l - A_l^T\, y \in \mathbb{R}_+^{n_l},\\
&\qquad c_\alpha - A_\alpha^T\, y \in \mathcal{K}_{\alpha_1} \times \cdots \times \mathcal{K}_{\alpha_{n_\alpha}},\\
&\qquad c_e - A_e^T\, y \in \mathcal{K}_{\exp} \times \cdots \times \mathcal{K}_{\exp}.
\end{aligned} \tag{A.2}$$

A.2 Initialization

Our solver is restricted to the proper cones that are the direct product of $\mathbb{R}_+^n$, $\mathcal{K}_\alpha$ and $\mathcal{K}_{\exp}$. We propose the following initialization that requires the introduction of only 3 additional variables and 3 additional linear inequality constraints. In the following we discuss the initialization for each of the three cones separately. The generalization to mixed problems is straightforward.

A.2.1 Initializing the linear constraints

In order to initialize the linear constraints

$$c_l - A_l^T\, y \in \mathbb{R}_+^{n_l}, \tag{A.3}$$

we propose to solve the following auxiliary problem

$$\begin{aligned}
(AUX_{\text{lin}}) \qquad &\min \theta\\
&\text{s.t. } c_{l,i} - A_{l,i}^T\, y + \theta \geq 0,\ i = 1, \ldots, n_l,\\
&\qquad A_f^T y = c_f.
\end{aligned}$$

It is clear that (AUX_{lin}) can be easily initialized by taking any y_0 that solves the linear system $A_f^T y = c_f$ and setting

$$\theta_0 = \max_{i=1,\ldots,n_l} c_{l,i} - A_{l,i}^T\, y_0 + 1.$$

As soon as we have found a feasible point $(\bar{y}, \bar{\theta})$ for (AUX_{lin}) such that $\bar{\theta} < 0$ we can stop and take $\bar{y}$ as a strictly feasible point for the linear constraints $c_l - A_l^T\, y \in \mathbb{R}_+^{n_l}$. On the other hand, if (A.3) admits a strictly feasible solution, then the optimal value of (AUX_{lin}) is negative.

A.2.2 Initializing the power cone constraints

For initializing the power cone constraints we propose the following auxiliary problem. For sake of simplicity we consider here the case of one single power cone constraint $c_\alpha - A_\alpha^T\, y \in \mathcal{K}_\alpha$, where $c_\alpha = [c_1, c_2, c_3] \in \mathbb{R}^3$ and $A_\alpha = [a_1, a_2, a_3]$ and $a_i \in \mathbb{R}^m$. Then the above power cone constraint becomes

$$(c_1 - a_1^T y)^\alpha \cdot (c_2 - a_2^T y)^{1-\alpha} \geq |c_3 - a_3^T y|. \tag{A.4}$$

Let us consider the auxiliary problem.

$$\begin{aligned} (AUX_\alpha) \qquad & \min \theta \\ & \text{s.t. } (c_1 - a_1^T y + \theta)^\alpha \cdot (c_2 - a_2^T y + \theta)^{1-\alpha} \geq |c_3 - a_3^T y|, \\ & \quad A_f^T y = c_f. \end{aligned}$$

Equivalently, the power cone constraint of (AUX_α) can be written as

$$c_\alpha - \tilde{A}_\alpha^T \tilde{y} \in \mathcal{K}_\alpha,$$

where $\tilde{A}_\alpha = \begin{bmatrix} & A_\alpha & \\ -1 & -1 & 0 \end{bmatrix} \in \mathbb{R}^{m+1,3}$ and $\tilde{y} = (y, \theta)$.

(AUX_α) can be initialized easily by taking any y_0 that solves the linear system $A_f^T y = c_f$ and setting $\theta_0 = \max\{\theta_1, \theta_2\}$, where

$$\begin{aligned} \theta_1 &= \max(-s_1, -s_2) + 1, \\ \theta_2 &= |s_3|^{1/(\alpha(1-\alpha))} - \min\{s_1, s_2\} + 1 \end{aligned}$$

and $s_i = c_i - a_i^T y_0$ for $i = 1, 2, 3$. Then we have

$$s_i + \theta_0 \geq s_i + \theta_1 \geq 1 > 0, \ i = 1, 2$$

and

$$(s_1 + \theta_0)^\alpha \cdot (s_2 + \theta_0)^{1-\alpha} \geq (\min\{s_1, s_2\} + \theta_0)^{\alpha(1-\alpha)} \geq (\min\{s_1, s_2\} + \theta_2)^{\alpha(1-\alpha)}$$

because of monotonicity of the power function. Using the definition of θ_2, we get

$$(s_1 + \theta_0)^\alpha \cdot (s_2 + \theta_0)^{1-\alpha} \geq (|s_3|^{1/(\alpha(1-\alpha))} + 1)^{\alpha(1-\alpha)} > |s_3|,$$

which means that the initial (y_0, θ_0) satisfies strictly the power cone constraint in (AUX_α). It remains to note that as soon as we have found a point $(\bar{y}, \bar{\theta})$ that is feasible for (AUX_α) with $\bar{\theta} < 0$, we can conclude that $\bar{y}$ is strictly feasible for (A.4).

Conversely, it is clear that if the optimal value of (AUX_α) is $\theta^* = 0$, then the power cone constraint in (AUX_α) must be tight at the optimum and y^* is feasible for (A.4), but not strictly feasible. Similarly, if $\theta^* > 0$, then there exists no feasible point y for (A.4).

A.2.3 Initializing the exponential cone constraints

For initializing the exponential cone constraints we propose the following auxiliary problem. Again, we only consider here the case of one single exponential cone constraint $c_{\exp} - A_{\exp}^T y \in \mathcal{K}_{\exp}$, where $c_{\exp} = [c_1, c_2, c_3] \in \mathbb{R}^3$ and $A_{\exp} = [a_1, a_2, a_3]$ and $a_i \in \mathbb{R}^m$. Then the above exponential cone constraint becomes

$$\exp\left(\frac{c_1 - a_1^T y}{c_3 - a_3^T y}\right) \leq \frac{c_2 - a_2^T y}{c_3 - a_3^T y}. \tag{A.5}$$

Let us consider the auxiliary problem.

$$
\begin{aligned}
(AUX_{\text{exp}}) \qquad & \min \theta \\
& \text{s.t. } \exp\left(\frac{c_1 - a_1^T y - \delta}{c_3 - a_3^T y + \tau}\right) \le \frac{c_2 - a_2^T y + \delta}{c_3 - a_3^T y + \tau}, \\
& \theta \ge \delta, \\
& \theta \ge |\tau|, \\
& A_f^T y = c_f.
\end{aligned}
$$

The exponential cone constraint of (AUX_{exp}) can be written as

$$c_{\text{exp}} - \tilde{A}_{\text{exp}}^T \tilde{y} \in \mathcal{K}_{\text{exp}},$$

where $\tilde{A}_{\text{exp}} = \begin{bmatrix} & A_\alpha & \\ 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \in \mathbb{R}^{m+3,3}$ and $\tilde{y} = (y, \theta, \delta, \tau)$.

We see that a strictly feasible starting point for (AUX_{exp}) can be found by taking y_0 as any solution to the linear system of equations $A_f^T y = c_f$. Denote $s_i = c_i - a_i^T y_0$ for $i = 1, 2, 3$. Then we can define

$$
\begin{aligned}
\delta_0 &= \max\{s_1, -s_2\} + 1, \\
\tau_0 &= -s_3 + 1.
\end{aligned}
$$

Then we see that $\tau_0 + s_3 = 1$, and on the other hand $s_2 + \delta_0 \ge 1$ and $s_1 - \delta_0 \le -1 < 0$. This implies

$$\exp\left(\frac{c_1 - a_1^T y_0 - \delta_0}{c_3 - a_3^T y_0 + \tau_0}\right) = \exp(s_1 - \delta_0) < \exp(0) = 1 \le s_2 + \delta_0 = \frac{c_2 - a_2^T y_0 + \delta_0}{c_3 - a_3^T y_0 + \tau_0}.$$

Finally, we can define $\theta_0 = \max\{\delta_0, |\tau_0|\} + 1$.

It is clear that if there exists a feasible point $\bar{y}$ for (A.5), then $(\bar{y}, 0, 0, 0))$ is feasible for (AUX_{exp}) with objective value 0. On the other hand, 0 is a trivial lower bound for the optimal objective value of (AUX_{exp}). Reversely, if we find a point $(\bar{y}, \bar{\theta}, \bar{\delta}, \bar{\tau})$ that is feasible for (AUX_{exp}) with $\bar{\theta} = 0$, then $\tau = 0$ and $\delta \le 0$, which implies

$$\exp\left(\frac{c_1 - a_1^T \bar{y}}{c_3 - a_3^T \bar{y}}\right) \le \exp\left(\frac{c_1 - a_1^T \bar{y} - \bar{\delta}}{c_3 - a_3^T \bar{y} + \bar{\tau}}\right) \le \frac{c_2 - a_2^T \bar{y} + \bar{\delta}}{c_3 - a_3^T \bar{y} + \bar{\tau}} \le \frac{c_2 - a_2^T \bar{y}}{c_3 - a_3^T \bar{y}},$$

i.e. $\bar{y}$ is feasible for (A.5). Moreover, if $\bar{\delta} < 0$ (and $\bar{\theta} = 0$), then the above inequality is strict which means that $\bar{y}$ is strictly feasible for (A.5).

On the other hand, if we find that $\theta^* > 0$, then we can conclude that (A.5) is not feasible.

A.3 Input and output format

The input and output format of our conic solver is similar to the one of SeDuMi[3] format. The input consists of parts, the data matrix A which is partitioned as

[3] http://sedumi.ie.lehigh.edu/

described above, the primal objective vector c which is also described as earlier, the dual objective vector b and a Matlab structure K which describes the structure of the primal or dual cone. The necessary fields in K are:

- $K.f$, the number of primal free variables,
- $K.l$, the number of nonnegativity constraints,
- $K.p$, a vector of size n_α containing the exponents α_i of the power cones,
- $K.e$, the number of exponential cones,
- $K.pd$, the flag determining whether the principal problem is the primal or dual formulation, values: '`p`' or '`d`'.

The optional fields of K are:

- $K.method$, the flag determining which method shall be used ('`1`': dual-path-following, '`2`': nonsymmetric predictor-corrector method, default value: '`1`'),
- $K.problem\_class$, a string determining the problem class ('`GP`': geometric programming, '`LOC`': generalized location problems, default value: '`default`'),
- $K.partial$, boolean variable determining whether or not to use partial minimization, only available if problem class is either '`GP`' or '`LOC`', default value: '`false`',
- $K.quiet$, boolean variable determining whether quiet or verbose output, default value: '`false`',
- $K.no\_output$, boolean variable determining whether no output at all, default value: '`false`',
- $K.szs$, vector containing additional problem information, for problem class '`GP`' its components is the number of monomials in each of the posynomials, for problem class '`LOC`' the first component of $K.szs$ is the dimension N, the second is the number of norm terms M.

The output of the solver contains the primal-dual solution (x, y) and a Matlab structure `info` containing additional output information. If the dual path-following method is used, then the the variable x contains only zeros (because the primal optimal solutions has not been computed). The structure `info` contains the following fields:

- $info.iter$, the number of main iterations
- $info.time$, the computation time (in seconds) for the main phase)
- $info.time\_init$, the computation time (in seconds) for the initialization phase),
- $info.time\_iter$, the time (in seconds) per iteration in the main phase
- $info.feas$: dual problem '`feasible`' or '`infeasible`',

- $info.bounded$: dual problem '`bounded`' or '`unbounded`',
- $info.status$: dual problem '`solved`' or '`not solved`'.

Optionally, a dual starting point y_0 can be provided. If y_0 is indeed strictly feasible, then the main phase starts at y_0. Otherwise a strictly feasible starting point is computed by solving an auxiliary model as proposed in Section A.2.

To summarize, the calling sequence for the conic solver based on the power cone is

```
[x,y,info] = powersolver(A,b,c,K,y0)
```

where `y0` is an optional dual (strictly feasible) starting point. The first input argument `A` is the *full* data matrix $\bar{A} = [A_f, A_l, A_\alpha, A_e]$, `b` is the dual objective vector and `c` is the full primal objective vector $\bar{c} = [c_f; c_l; c_\alpha; c_e]$.

The output contains the primal optimal solution `x` (which is only available if the primal-dual method is used, otherwise it is set to zero), the dual optimal solution `y` and an information structure `info` with fields as described above.

A.4 Yalmip interface

As we have mentioned above, in order to call our solver one needs to provide the data $\bar{A}$, b and $\bar{c}$ as well as a Matlab structure K defining the cone $\mathcal{K}^*$. Even for tiny problems this can be a tedious and error-prone task.

Therefore we have established with the help of Johan Löfberg ([39]) an interface to YALMIP[4]. YALMIP is a Matlab-based modelling language for defining and solving advanced optimization problems. The calling sequence for solving an optimization problem in YALIMP is given by `solvesdp(C,h)`, where C is a set of constraints and h is the (linear) objective function. By default, minimization is assumed. For example, given a matrix A and vectors b and c in suitable dimension

```
x = sdpvar(length(c),1);
C = [A*x<b];
h = c'*x;
solvesdp(C,h);
```

solves the linear problem

$$\begin{aligned} &\min c^T x \\ &\text{s.t. } A \cdot x \le b. \end{aligned}$$

In order to call our conic solver using YALMIP, one needs to have YALMIP installed and the root directory (and all of its subdirectories) of our solver have to be in the Matlab path.

The set C may contain linear constraints and power cone constraints. At the moment exponential cone constraints are not yet supported. Power cone constraints are defined in the following way.

[4] `http://control.ee.ethz.ch/~joloef/wiki/pmwiki.php?n=Main.HomePage`

```
sdpvar x y z
C = powercone(z,x,y,alpha)
```

where `alpha` is a scalar constant between 0 and 1. The above definition corresponds to the constraint $(x, y, z) \in \mathcal{K}_\alpha$, where α is the above-mentioned constant. Note the sequence of the input arguments (i.e. z, x, y instead of x, y, z).

The calling sequence for solving a conic problem involving linear and power cone constraints using our conic solver is

```
options = sdpsettings('solver','powersolver','savesolveroutput',1);
output = solvesdp(C,h,options)
```

The first line specifies the options (i.e. to use our conic solver, and to provide additional output information). The second line calls the main Yalmip routine to solve the optimization problem to minimize the objective `h` due to the constraints `C` with the options specified above.

Additionally, we have added support for p-cone constraints. Let x and t be `SDPVAR` objects and $p \geq 1$ a scalar constant. Then the constraint $||x||_p \leq t$ (or equivalently $(x, t) \in \mathcal{P}_p^{(n)}$) can be expressed by the command `p_cone(x,t,p)`.

A.5 A concrete example

Let us illustrate the usage of our conic solver on the problem class that we have encountered already in Section 4.5.3, i.e.

$$\begin{aligned} \min_{x,t} \ & d^T x + t \\ \text{s.t. } & (x,t) \in \mathcal{C}_p \\ & t \leq 1, \end{aligned} \tag{A.6}$$

where

$$\mathcal{C}_p = \left\{ (x,t) : \sum_{i=1}^N |x_i|^{p_i} \leq t^{p_0}, \ t \geq 0 \right\},$$

$d \in \mathbb{R}^N$ and $1 \leq p_0 \leq \min_{i=1,\dots,N} p_i$. We have seen in Section 4.1.4 how $\mathcal{C}_p$ can be decomposed in terms of power cones $\mathcal{K}_\alpha$. For $N = 2$ the conic reformulation of (A.6) becomes

$$\begin{aligned} - \max_{x,t,v,w} \ & - d^T x - t \\ \text{s.t. } & (v_i, 1, x_i) \in \mathcal{K}_{\alpha_i}, \ \alpha_i = \frac{p_0}{p_i}, \ i = 1, 2, \\ & (w_i, t, v_i) \in \mathcal{K}_{\alpha_0}, \ \alpha_0 = \frac{1}{p_0}, \ i = 1, 2, \\ & 1 - t \in \mathbb{R}_+, \\ & w_1 + w_2 = t. \end{aligned} \tag{A.7}$$

We see that we could remove the dependence on the variable t, but for the sake of transparency we will keep this extended formulation. There is one linear equality constraint present ($w_1 + w_2 = t$), one linear inequality constraint ($1 - t \in \mathbb{R}_+$) and four power cone constraints. This means we have $n_f = 1$, $n_l = 1$, $n_\alpha = 4$ and the number of dual variables is $m = 7$.

A.5.1 Direct call

In order to call our conic solver, we need to provide the data $\bar{A}$, b and $\bar{c}$ as well as a structure K defining the cone $\mathcal{K}^*$. We see immediately that $K.f = n_f = 1$, $K.l = n_l = 1$, $K.p = \left[\frac{p_0}{p_1}, \frac{p_0}{p_2}, \frac{1}{p_0}, \frac{1}{p_0}\right]$, $K.e = 0$ and $K.pd =$'d' because we are using a dual formulation.

Further, if we define the dual variables $y = (x, t, v, w)$, the dual objective vector becomes $b = (-d, -1, 0) \in \mathbb{R}^7$. The linear equality constraint $w_1 + w_2 = t$ only involve the variables w and t (and no constant term). Therefore we conclude $c_f = 0$ and $A_f = [0, 0, -1, 0, 0, 1, 1]^T$. The linear inequality constraint involves the constant term $c_l = 1$ and only the variably t, therefore we have $A_l = [0, 0, 1, 0, 0, 0, 0]^T$. The first power cone constraint $(v_1, 1, x_1) \in \mathcal{K}_{\alpha_1}$ involves only the variable v_1 (and no constant term) in the first component; the second component does not involve any of the variables, but the only the constant term 1; the last component involves only the variable x_1. Therefore, if we define $c_{\alpha,1} = [0, 1, 0]^T$ and

$$A_{\alpha,1} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

we have that $(v_1, 1, x_1) \in \mathcal{K}_{\alpha_1}$ is equivalent to $c_{\alpha,1} - A_{\alpha,1}^T y \in \mathcal{K}_{\alpha_1}$. In a similar fashion we can define $c_{\alpha,i}$ and $A_{\alpha,i}$, for $i = 1, 2, 3$, that correspond to the remaining three power cone constraints. The following Matlab code solves the conic reformulation (A.7) of the problem (A.6):

```
alpha = [p(end)/p(1);p(end)/p(2);1/p(end)*ones(2,1)];
K = struct('f',1,'l',1,'p',alpha,'e',0,'pd','d');
b = sparse((1:3),ones(3,1),[-d;-1],7,1);
c = sparse([2,4,7],[1,1,1],[1,1,1],14,1);
A_f = sparse([3,6,7],ones(3,1),[-1,1,1],7,1);
A_l = sparse(3,1,1,7,1);
A_p = sparse([1,2,3,3,4,4,5,5,6,7],[3,6,8,11,1,9,4,12,7,10],...
     -ones(10,1),7,12);
A = [A_f,A_l,A_p];

[x,y,s,info] = powersolver(A,b,c,K);
```

The first line defines the power cone exponents that correspond to the four power cones that are present in the conic formulation (A.7). The second line defines the Matlab structure that determines the cone $\mathcal{K}^*$. The following 7 lines define the data $\bar{A}$,b and $\bar{c}$ in sparse form, where b is the objective vector of (A.7), $\bar{c}$ contains three ones (one corresponding to c_f and two corresponding to c_α). Finally, $\bar{A}$ consists of three blocks: A_f, A_l and A_α, where A_α in turn consists of four sparse 7×3-blocks $A_{\alpha,1}, A_{\alpha,2}, A_{\alpha,3}, A_{\alpha,4}$

A.5.2 Indirect call using Yalmip

As we have mentioned above, our solver is partially supported by Yalmip, a Matlab interface for solving optimization problems with links to an extensive list of nonlinear solvers. This link to Yalmip significantly simplifies the modelling process.

Using the results from Section 4.5.3, we see that the conic reformulation of (A.7) using p-cones is

$$\begin{aligned}\min_{x,t,v}\ & d^T x + t \\ \text{s.t. } & (v_i, 1, x_i) \in \mathcal{K}_{\alpha_i},\ \alpha_i = \frac{p_0}{p_i},\ i = 1, 2, \\ & (v, t) \in \mathcal{K}_{p_0}, \\ & t \leq 1.\end{aligned} \tag{A.8}$$

The Matlab code for solving (A.8) is

```
x = sdpvar(n,1); t = sdpvar;
v = sdpvar(n,1);
C = [powercone(x(1),v(1),1,p(end)/p(1))];
C = C + [powercone(x(2),v(2),1,p(end)/p(2))];
C = C + [p_cone(v,t,p(end))];
C = C + [t<=1];
h = d'*x +t;

options = sdpsettings('solver','powersolver','savesolveroutput',1);
output = solvesdp(C,h,options);
```

We see that the above code snippets are very close to the actual mathematical formulation (A.8). Lines 1 and 2 are introducing the original variables x and t as well as the artificial variables v as `SDPVAR` objects. The following four lines define the four constraints (two power cone constraints, one p-cone constraint and one linear constraint), the 7th line defines the objective and the last two lines define the options and call of the solver.

Bibliography

[1] K.M. Anstreicher. On long step path following and sumt for linear and quadratic programming. *SIAM Journal on Optimization*, 6:33–46, 1996.

[2] A. Barvinok. *A Course in Convexity.* AMS, 2002.

[3] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization. Analysis, Algorithms, and Engineering Applications.* SIAM, Philadelphia, 2001.

[4] D.S. Bernstein. *Matrix mathematics.* Princeton University Press, 2005.

[5] S. Boyd, S. Kim, L. Vandenberghe, and A. Hassibi. A tutorial on geometric programming. *Optimization and Engineering*, 8(1):67–127, 2007.

[6] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[7] A. Brooke, D. Kendrick, A. Meeraus, and R. Raman. *GAMS: A User's guide.* The Scientific Press, South San Francisco, 1998.

[8] T.J. Carpenter, I.J. Lustig, J.M. Mulvey, and D.F. Shanno. Separable quadratic programming via a primal-dual interior point method and its use in a sequential procedure. *ORSA Journal on Computing*, 5(2):182–191.

[9] R. Chares and F. Glineur. An interior-point method for the location problem with mixed norms using a conic formulation. *Mathematical Methods of Operations Research*, 68(3):383–405, 2008.

[10] R. Chares and F. Glineur. New self-concordant barriers for the power cone. unpublished manuscript, 2009.

[11] R. Chares and F. Glineur. Variable reduction via approximate partial minimization. unpublished manuscript, 2009.

[12] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. E. Knuth. On the lambert w function. *Advances in Computational Mathematics*, 5:329 – 359, 1996.

[13] G.B. Dantzig. Programming in a linear structure. *Econometrica*, 17:73–74, 1949.

[14] G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, New Jersey, 1963.

[15] M. del Mar Hershenson, S.P. Boyd, and T.H. Lee. Optimal design of a cmos op-amp via geometric programming. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 20(1):1–21, 2001.

[16] R. Duffin, E. Peterson, and C. Zener. *Geometric programmingtheory and application*. Wiley, New York, 1967.

[17] A. Fiacco and G. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. Wiley, New York, 1968.

[18] A. Forsgren and P.E. Gill. Primal-dual interior methods for nonconvex nonlinear programming. *SIAM Journal of Optimization*, 8:1132–1152, 1998.

[19] R. Fourer. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, 2002.

[20] R.W. Freund, F. Jarre, and S. Schaible. On interior–point methods for fractional programs and their convex reformulation. Technical Report 94–17, Murray Hill, NJ 07974, USA, 1994.

[21] K.R. Frisch. Principles of linear programming - with particular reference to the double gradient form of the logarithmic potential method. Technical report, University Institute of Economics, Oslo, Memorandum on October 18, 1954.

[22] K. Fujisawa, M. Kojima, and K. Nakata. Exploiting sparsity in primal-dual interior-point methods for semidefinite programming. *Mathematical Programming*, 79:235–253, 1997.

[23] F. Glineur. *Topics in Convex Optimization: Interior-Point methods, Conic Duality and Approximations*. Ph.D. thesis, Faculté Polytechnique de Mons, Mons, Belgium, January 2001.

[24] F. Glineur and T. Terlaky. Conic formulation for l_p-norm optimization. *Journal of Optimization Theory and Applications*, 122(2):285–307, 2004.

[25] M.X. Goemans and F. Rendl. Semidefinite programming in combinatorial optimization. *Mathematical Programming*, 79:143–161, 1999.

[26] M.C. Grant. *Disciplined convex programming*. Ph.D. thesis, Department of Electrical Engineering, Stanford University, Stanford, USA, December 2004.

[27] M.C. Grant. *CVX Users' Guide, version 0.85*, 2005.

[28] O. Güler. Barrier functions in interior point methods. *Mathematics of Operations Research*, 21:860–885, 1996.

[29] P. Huard. Resolution of mathematical programming with nonlinear constraints by the method of centers. In *Nonlinear Programming (J. Abadie, ed.)*, page 207219. North Holland, Amsterdam, The Netherlands, 1967.

[30] F. Jarre. Interior-point methods via self-concordance or relative lipschitz condition. Habilitation thesis, Würzburg, Germany, March 1994.

[31] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[32] W. Karush. Minima of functions of several variables with inequalities as side constraints. Master's thesis, Dept. of Mathematics, Univ. of Chicago, Chicago, Illinois, USA, 1939.

[33] L.G. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[34] V. Klee and G.J. Minty. How good is the simplex algorithm? In *Inequalities*, pages 159–175. O. Shisha ed.,Academic Press, New York, 1972.

[35] M. Koecher. Positivitätsbereiche im $\mathbb{R}^n$. *American Journal of Mathematics*, 79:575–596, 1957.

[36] K. Koh, S. Kim, A. Mutapcic, and S. Boyd. Gpcvx - a matlab solver for geometric programs in convex form. Technical report, Stanford, CA 94305, USA, 2006.

[37] M.H. Koulaei and T. Terlaky. On the extension of a mehrotra-type algorithm for semidefinite optimization. *Optimization Online*, March 2007.

[38] H.W. Kuhn and A.W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492. University of California Press, Berkeley, 1951.

[39] J. Löfberg. Yalmip: A toolbox for modeling and optimization in matlab, 2004.

[40] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal of Optimization*, 2:575–601, 1992.

[41] S. Mizuno, M.J. Todd, and Y. Ye. On adaptive step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18:945–981, 1993.

[42] A. Nemirovski and M. Todd. Interior-point methods for optimization. *Acta Numerica*, pages 191–234, 2008.

[43] A. Nemirovski and D.B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley-Interscience, New York, 1983.

[44] Y. Nesterov. Interior-point methods: An old and new approach to nonlinear programming. *Mathematical Programming*, 79:285–297, 1997.

[45] Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. CORE Discussion Papers 1997044, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), June 1997.

[46] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course.* Kluwer Academic Publishers, 2003.

[47] Y. Nesterov. Constructing self-concordant barriers for convex cones. *CORE Discussion Paper*, 30, 2006.

[48] Y. Nesterov. Nonsymmetric potential reduction methods for general cones. *CORE Discussion Paper*, 34, 2006.

[49] Y. Nesterov. Towards nonsymmetric conic optimization. *CORE Discussion Paper*, 28, 2006.

[50] Y. Nesterov. Parabolic target space and primal-dual interior-point methods. *Discrete Applied Mathematics*, 156(11):2079–2100, 2008.

[51] Y. Nesterov. Primal-dual interior-point methods with asymmetric barriers. *CORE Discussion Paper*, 57, 2008.

[52] Y. Nesterov and A. Nemirovski. *Interior-Point Polynomial Algorithms in Convex Programming.* SIAM, Philadelphia, 1994.

[53] Y. Nesterov and M.J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22:1–42, 1997.

[54] Y. Nesterov and M.J. Todd. Primal-dual interior-point methods for self-scaled cones. *SIAM Journal on Optimization*, 8:324–364, 1998.

[55] J. Renegar. *A Mathematical View of Interior-Point Methods in Convex Optimization.* SIAM, Philadelphia, 2001.

[56] R.T. Rockafellar. *Convex analysis.* Princeton University Press, 1970.

[57] C. Roos and H. Mansouri. Full-newton step polynomial-time methods for linear optimization based on locally self-concordant barrier functions. Working paper, Department of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2006.

[58] M. Salahi, J. Peng, and T. Terlaky. On mehrotra-type predictor-corrector algorithms. Technical report, Advanced Optimization Lab., Department of Computing and Software, McMaster University, Hamilton, ON, Canada, 2005.

[59] N.Z. Shor. The use of operations of space dilatation in problems of minimization of convex functions. *Kibernetika*, (1):6–12, 1970.

[60] M. Slater. Lagrange multipliers revisited: a contribution to non-linear programming, 1950. Cowles Commission Discussion Paper, Math 403.

[61] J.F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653, 1999.

[62] M.J. Todd, K.C. Toh, and R.H. Tütüncü. On the nesterov-todd direction in semidefinite programming. *SIAM Journal on Optimization*, 8:769–796, 1998.

[63] V.A. Truong and L. Tunçel. Geometry of homogeneous convex cones, duality mapping, and optimal self-concordant barriers. *Mathematical Programming*, 100:295–316, 2004.

[64] L. Tunçel and A. Nemirovski. Self-concordant barriers for convex approximations of structured convex sets. Research Report CORR 2007-03, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada, February 2007.

[65] R.J. Vanderbei and D.F. Shanno. An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13(1-3):231–252, 1999.

[66] G. Xue and Y. Ye. An efficient algorithm for minimizing a sum of p-norms. *SIAM Journal on Optimization*, 10(2):551–579, 1998.

[67] Y. Ye. A class of projective transformations for linear programming. *SIAM Journal on Computing*, 19:457–466, 1990.