Convex Optimization via Domain-Driven Barriers and Primal-Dual Interior-Point Methods

by

Mehdi Karimi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2017

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner	**Lieven Vandenberghe** Professor, Electrical Engineering Department University of California, Los Angeles
Supervisor	**Levent Tunçel** Professor, Department of Combinatorics & Optimization
Internal Members	**William Cook** Professor, Department of Combinatorics & Optimization
	Stephen Vavasis Professor, Department of Combinatorics & Optimization
Internal-external Member	**Yaoliang Yu** Professor, Department of Computer Science

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

This thesis studies the theory and implementation of infeasible-start primal-dual interior-point methods for convex optimization problems. Convex optimization has applications in many fields of engineering and science such as data analysis, control theory, signal processing, relaxation and randomization, and robust optimization. In addition to strong and elegant theories, the potential for creating efficient and robust software has made convex optimization very popular. Primal-dual algorithms have yielded efficient solvers for convex optimization problems in conic form over symmetric cones (linear-programming (LP), second-order cone programming (SOCP), and semidefinite programing (SDP)). However, many other highly demanded convex optimization problems lack comparable solvers. To close this gap, we have introduced a general optimization setup, called *Domain-Driven*, that covers many interesting classes of optimization. Domain-Driven means our techniques are directly applied to the given "good" formulation without a forced reformulation in a conic form. Moreover, this approach also naturally handles the cone constraints and hence the conic form.

A problem is in the Domain-Driven setup if it can be formulated as minimizing a linear function over a convex set, where the convex set is equipped with an efficient self-concordant barrier with an easy-to-evaluate Legendre-Fenchel conjugate. We show how general this setup is by providing several interesting classes of examples. LP, SOCP, and SDP are covered by the Domain-Driven setup. More generally, consider all convex cones with the property that both the cone and its dual admit efficiently computable self-concordant barriers. Then, our Domain-Driven setup can handle any conic optimization problem formulated using direct sums of these cones and their duals. Then, we show how to construct interesting convex sets as the direct sum of the epigraphs of univariate convex functions. This construction, as a special case, contains problems such as geometric programming, p-norm optimization, and entropy programming, the solutions of which are in great demand in engineering and science. Another interesting class of convex sets that (optimization over it) is contained in the Domain-Driven setup is the generalized epigraph of a matrix norm. This, as a special case, allows us to minimize the nuclear norm over a linear subspace that has applications in machine learning and big data. Domain-Driven setup contains the combination of all the above problems; for example, we can have a problem with LP and SDP constraints, combined with ones defined by univariate convex functions or the epigraph of a matrix norm.

We review the literature on infeasible-start algorithms and discuss the pros and cons of different methods to show where our algorithms stand among them. This thesis contains a chapter about several properties for self-concordant functions. Since we are dealing with

general convex sets, many of these properties are used frequently in the design and analysis of our algorithms. We introduce a notion of duality gap for the Domain-Driven setup that reduces to the conventional duality gap if the problem is a conic optimization problem, and prove some general results. Then, to solve our problems, we construct infeasible-start primal-dual central paths. A critical part in achieving the current best iteration complexity bounds is designing algorithms that follow the path efficiently. The algorithms we design are predictor-corrector algorithms.

Determining the status of a general convex optimization problem (as being unbounded, infeasible, having optimal solutions, etc.) is much more complicated than that of LP. We classify the possible status (seven possibilities) for our problem as: solvable, strictly primal-dual feasible, strictly and strongly primal infeasible, strictly and strongly primal unbounded, and ill-conditioned. We discuss the certificates our algorithms return (heavily relying on duality) for each of these cases and analyze the number of iterations required to return such certificates. For infeasibility and unboundedness, we define a weak and a strict detector. We prove that our algorithms return these certificates (solve the problem) in polynomial time, with the current best theoretical complexity bounds. The complexity results are new for the infeasible-start models used. The different patterns that can be detected by our algorithms and the iteration complexity bounds for them are comparable to the current best results available for infeasible-start conic optimization, which to the best of our knowledge is the work of Nesterov-Todd-Ye (1999).

In the applications, computation, and software front, based on our algorithms, we created a Matlab-based code, called DDS, that solves a large class of problems including LP, SOCP, SDP, quadratically-constrained quadratic programming (QCQP), geometric programming, entropy programming, and more can be added. Even though the code is not finalized, this chapter shows a glimpse of possibilities. The generality of the code lets us solve problems that CVX (a modeling system for convex optimization) does not even recognize as convex. The DDS code accepts constraints representing the epigraph of a matrix norm, which, as we mentioned, covers minimizing the nuclear norm over a linear subspace. For acceptable classes of convex optimization problems, we explain the format of the input. We give the formula for computing the gradient and Hessian of the corresponding self-concordant barriers and their Legendre-Fenchel conjugates, and discuss the methods we use to compute them efficiently and robustly. We present several numerical results of applying the DDS code to our constructed examples and also problems from well-known libraries such as the DIMACS library of mixed semidefinite-quadratic-linear programs. We also discuss different numerical challenges and our approaches for removing them.

Acknowledgements

I would like to thank my family and friends for their support, specially my wife Mehrnoosh. I thank my advisor, Dr. Levent Tunçel, who gave me this opportunity to be his student, and his guidance and support made this work possible. I also thank my committee members, namely Dr. William Cook, Dr. Lieven Vandenberghe, Dr. Stephen Vavasis, and Dr. Yaoliang Yu, for their helpful feedback.

The material in this thesis is based upon research supported in part by NSERC Discovery Grants, Ontario Graduate Scholarships (OGS), Sinclair Graduate Scholarship, and U.S. Office of Naval Research under award numbers: N00014-12-1-0049 and N00014-15-1-2171. This financial support is gratefully acknowledged.

Dedication

This thesis is dedicated to the love of my life, Mehrnoosh.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Convex optimization in the view of this thesis

Convex optimization has been studied heavily not only for its very powerful and elegant theory, but also because of its widespread applications in many different fields of engineering and science. By convex optimization we mean minimizing a linear function of finitely many variables over a *convex set* in a finite dimensional vector space.

Definition 1.1.1. *For a general metric vector space* $\mathbb{E}$*, a set* $C \subseteq \mathbb{E}$ *is a* convex set *if for every* $x, y \in C$ *and every* $\lambda \in [0, 1]$*,* $\lambda x + (1 - \lambda)y \in C$*. A convex set* $K \subseteq \mathbb{E}$ *is a* convex cone *if for every* $x \in K$ *and every* $\lambda \geq 0$*,* $\lambda x \in K$*.*

Many problems arising in practice can naturally be formulated as a convex optimization problem. However, even if our set S is not convex, minimizing a linear function over S is equivalent to minimizing it over the *convex hull* of S, which by definition is the smallest convex set containing S (see Figure 1.1). This simple fact lets us use convex relaxations for many non-convex problems and enjoy the strong theory and numerical stability. Just to list a few applications of convex optimization:

1. **Data analysis:** big data, machine learning, compressed sensing (see [10, 22, 13, 2]).

2. **Engineering:** control theory, signal processing, circuit design (see [9, 4, 7, 1]).

3. **Relaxation and randomization:** provable bounds and robust heuristics for hard non-convex problems (see [75]).

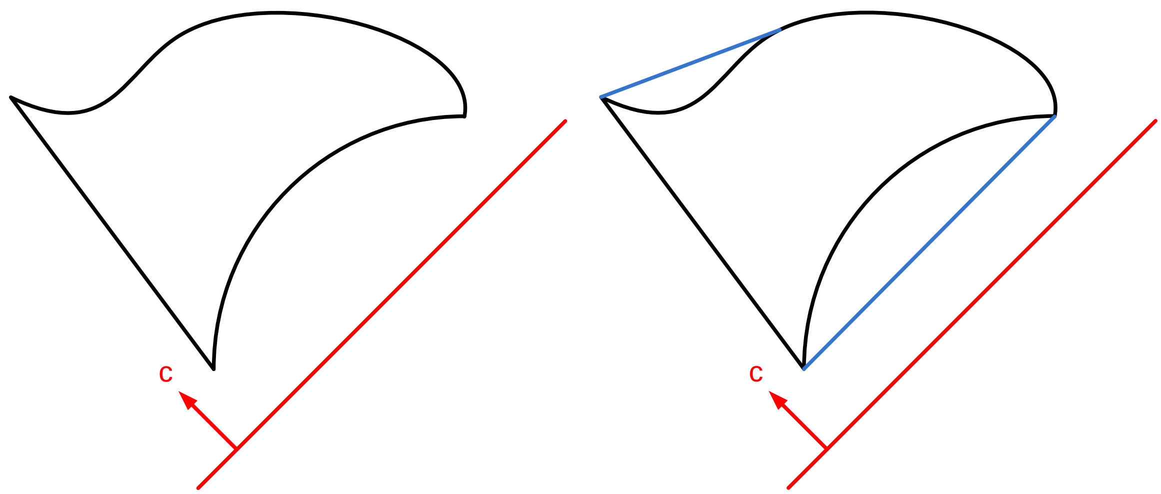

Figure 1.1: Optimizing a linear function over a set is equivalent to optimizing over its convex hull.

4. **Robust optimization** (see [5, 3]).

Convex optimization approaches became even more attractive when new algorithms such as modern *interior-point methods*, faster and more reliable numerical linear algebra software, and powerful computers made it possible to solve convex optimization problems efficiently. Generally speaking, interior-point methods follow a sequence of points in the relative interior of the feasible region to an optimal solution (see Figure 1.2). Modern theory of interior-point methods started with Karmarkar's revolutionary paper [24] in 1984. The research activity that followed led to faster theoretical algorithms [63, 47, 19], many interesting software projects [12, 69, 43, 73], applications [21, 61], and generalizations [52, 49, 74, 44]. Besides theoretical advantages, it was the appealing practical performance of interior-point methods that ended the 50-year unchallenged dominance of the Simplex method for linear programming (LP) and inspired a revolution in convex optimization.

The basic principles of classical interior-point methods for nonlinear optimization were developed in 1950's and 1960's, for example in [16]. However, extension of modern interior-point methods, with polynomial iteration complexity, from linear optimization to general convex optimization problems was started by Nesterov and Nemirovskii [52] in the late 1980's. Since then, many different approaches in that direction have been proposed, the majority of them for the problems presented in a conic formulation. A *conic optimization* problem is minimizing a linear function over the intersection of an affine subspace and a convex cone. One of the main strengths of conic optimization is its *duality* properties. For

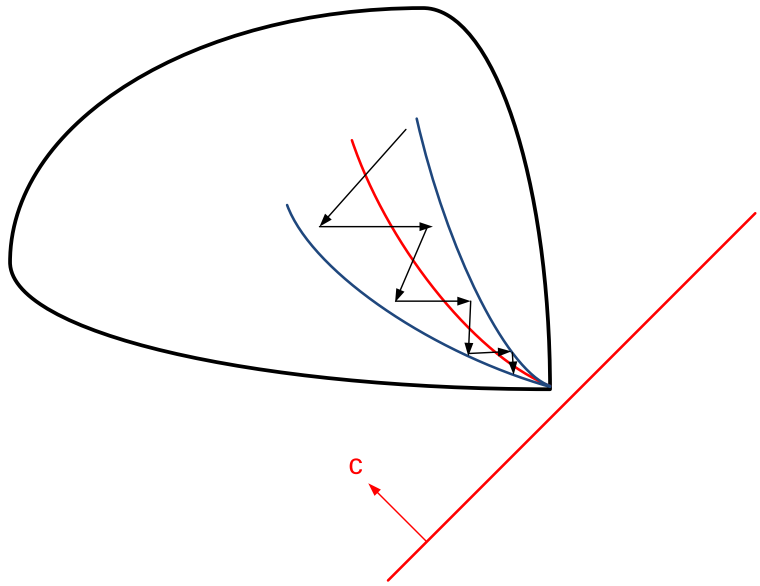

Figure 1.2: The scheme of a *path-following* interior-point algorithm that the sequence of points lie in a *neighborhood* of a *central path*.

every convex optimization problem, we assign a dual optimization problem whose feasible points give bounds on the objective value of the primal problem. The dual of a conic optimization problem is another conic optimization problem and we can define a nice *duality gap* of a pair of primal-dual feasible points that quantifies distance to optimality. This "symmetry" between the primal conic formulation and its dual lets us build a powerful and elegant theoretical foundation. An essential fact about the importance of conic optimization is that it is as general as convex optimization in the sense that every convex optimization problem can be written as a conic optimization problem. This can be done by adding an artificial variable to lift the feasible region into a higher dimensional space and writing it as the intersection of an affine subspace and a convex cone, as shown in Figure 1.3.

The focus on conic formulations is mostly because of its popularity in practice; linear programming (LP), second-order cone programming (SOCP), and semi-definite programming (SDP) are the most popular problems utilized in applications. It has been proven that there are many advantages in considering the primal and the dual problems at the same time and developing *primal-dual* interior-point methods; advantages such as defining efficient and easy-to-calculate proximity measures and potential functions that incorporate dual information, developing long-step algorithms, and better handling of the infeasible-

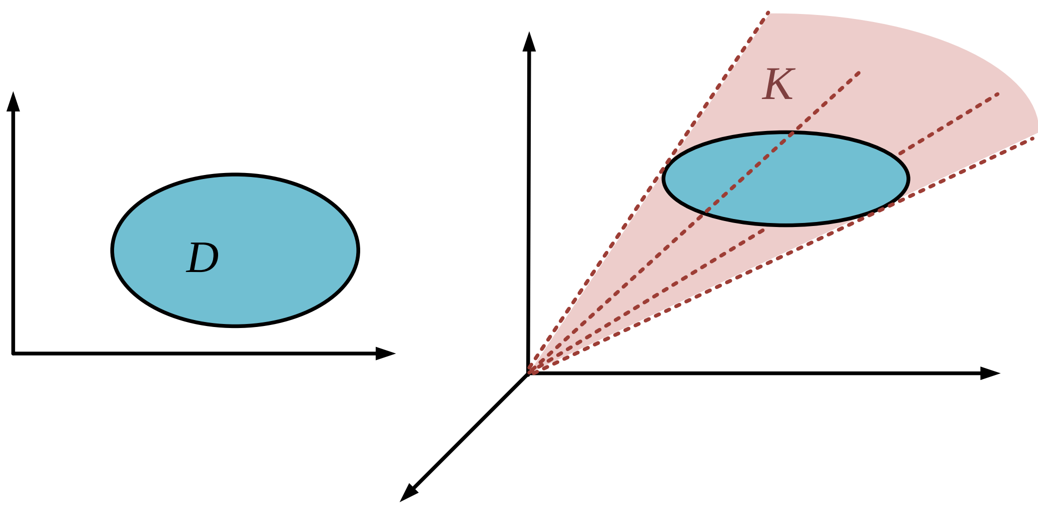

Figure 1.3: Lifting a convex set to write it as the intersection of an affine subspace and a convex cone.

start cases. The theory of Primal-Dual algorithms for conic formulations is strong and elegant, see for example [48, 56, 54, 55, 80, 40, 74, 44]. Considering that conic optimization is general, trying new methods seems unnecessary. Here, we highlight some arguments against this view:

- The application and software of conic optimization have not gone much beyond optimization over symmetric (self-scaled) cones; more specifically linear, second-order cone, and semidefinite programming. There have been efforts to enforce the desired properties of optimization over symmetric cones to general conic optimization [51, 68]; however, they have not been continued or put into practice.
- From a theoretical point of view, we do not need a conic formulation to achieve the current best theoretical bounds, see for example papers by Nemirovskii and Tunçel [46], and Nesterov and Nemirovskii [53]. In other words, to achieve the best theoretical bounds, enforcement into conic formulation is not needed.
- Enforcement of the conic framework is not costless. If there is no compelling numerical evidence, maybe working with the more natural formulation of the problem is logical.

In order to succeed, any new primal-dual method that does not use conic reformulation must not only be elegant in theory, but also open new doors in practice. In this thesis, we define an optimization setup, called *Domain-Driven*, which is more "general" than the conic setup. Our examples and discussion show that the Domain-Driven setup covers many interesting examples that arise in practice. We design primal-dual algorithms that not only achieve the best theoretical results available, but also lead to a code for solving many interesting convex optimization classes.

1.2 Organization of the thesis

In Chapter 2, we informally define a Domain-Driven setup and show many of its interesting applications. Our algorithms are infeasible-start, that is, we do not know beforehand a feasible starting point for our algorithms. Infeasible-start algorithms are much more challenging than feasible-start ones, both in theory and in applications. In Chapter 3, we review the literature on popular infeasible-start approaches and compare our methods to them. Self-concordant functions and their properties form the machinery we need to design and analyze our algorithms. Chapter 4 contains a list of the properties we frequently use, as well as the proofs for some theorems and lemmas specifically needed for this thesis.

After seeing the definition of self-concordant functions and barriers, we formally define the Domain-Driven setup in Chapter 5 and take another look at the examples of Chapter 2. Then, we define the notion of duality gap and prove some results about it. Designing our infeasible-start primal-dual central paths also comes in this chapter. Chapter 6 contains the expression for our predictor-corrector algorithms and the main tool of the proof that the proposed algorithms achieve the current best iteration complexity bounds. In Chapter 7, we define the statuses a given problem may have, and the corresponding certificates our algorithms return for each possible status. Relying on the main complexity result of Chapter 6, we show the number of iterations our algorithms take to return such certificates.

One of our final goals has been creating a code that solves the large number of optimization classes in the Domain-Driven setup. We have started to create such a code that currently solves all the examples we discuss in Chapter 2. The algorithm being used in the code has not been finalized; however, the code represents the potential for creating a general and efficient code for convex optimization. In Chapter 8, we discuss our code, called DDS. We show how to input each class of problems, and the difficulties and our techniques for calculating the gradient and Hessian of the self-concordant barrier for each class. Chapter 9 is the conclusion of the thesis.

1.3 Some notations

We represent two general Euclidean vector spaces by $\mathbb{E}$ and $\mathbb{Y}$. We show an inner product on a general vector space by $\langle \cdot, \cdot \rangle$. We show the dual space of $\mathbb{E}$ by $\mathbb{E}^*$ and a n-dimensional Euclidean vector space over real numbers by $\mathbb{R}^n$. When working with $\mathbb{R}^n$, for simplicity, we frequently use $y^\top x$ instead of $\langle y, x \rangle$. $\mathbb{R}_+$ and $\mathbb{R}_{++}$ represent nonnegative and positive real numbers, respectively. For a self-adjoint positive definite linear transformation $B : \mathbb{E} \to \mathbb{E}^*$, we define a conjugate pair of Euclidean norms as:

$$\begin{aligned} \|x\|_B &:= [\langle Bx, x \rangle]^{1/2}, \\ \|s\|_B^* &:= \max\{\langle s, y \rangle : \ \|y\|_B \le 1\} = \|s\|_{B^{-1}} = \left[\langle s, B^{-1}s \rangle\right]^{1/2}. \end{aligned} \tag{1.1}$$

Note that (1.1) immediately gives us a general Cauchy-Schwarz (CS) inequality that we use several times in the thesis:

$$\langle s, x \rangle \le \|x\|_B \|s\|_B^*, \quad \forall x \in \mathbb{E}, \forall s \in \mathbb{E}^*. \tag{1.2}$$

We denote the set of $m \times m$ symmetric matrices by $\mathbb{S}^m$, and the set of positive semidefinite and positive definite matrices by $\mathbb{S}_+^m$ and $\mathbb{S}_{++}^m$, respectively. The generalized inequality $X \succeq Y$ ($X \succ Y$) for $X, Y \in \mathbb{S}^m$ means that $X - Y \in \mathbb{S}_+^m$ ($X - Y \in \mathbb{S}_{++}^m$).

Consider two sets $D_1 \in \mathbb{E}_1$ and $D_2 \in \mathbb{E}_2$; we define the *direct sum* of them as

$$D_1 \oplus D_2 := \left\{(z^1, z^2) : z^1 \in D_1, \ z^2 \in D_2\right\}. \tag{1.3}$$

Throughout the thesis, RHS and LHS stand for right-hand-side and left-hand-side, respectively.

Chapter 2

Applications of Domain-Driven setup

In Chapter 1, we discussed the importance of convex optimization. The approach to solve a convex optimization problem depends on how the convex feasible region is presented or given as input data. In the Domain-Driven setup, the underlying convex set is treated as the domain of a *convex function*. Let us first define a convex function.

Definition 2.0.1. *Let $C \subseteq \mathbb{E}$ be a convex set. The function $f : C \to \mathbb{R}$ is called a* convex function *if for every two points $x, y \in C$ and every $\lambda \in (0,1)$, we have*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y).$$

The function is called strictly convex *if for every distinct pair $x, y \in C$ we have strict inequality in the above definition.*

Every convex set C is the domain of a convex function (for example the function that is constant on C and $+\infty$ otherwise). In the Domain-Driven setup, we assume that our convex set is the domain of a special convex function. We give a formal definition in Chapter 5, but let us show a less formal definition here. A convex optimization problem is said to be in the Domain-Driven setup if it can be written as

$$\inf_x \{\langle c, x\rangle : Ax \in D\}, \tag{2.1}$$

where $x \mapsto Ax : \mathbb{E} \to \mathbb{Y}$ is a linear embedding ($\ker A = \{0\}$), $c \in \mathbb{E}^*$ is given, and $D \subset \mathbb{Y}$ is presented as the closure of the domain of a convex function $\Phi(\cdot)$ that is a *self-concordant (s.c.) barrier* [52]. We define s.c. barriers in Chapter 4.

Every convex set is the domain of a s.c. barrier [52]. However, a s.c. barrier that can be efficiently evaluated is not necessarily available for a general convex set. One structure we assume for the Domain-Driven setup is that $\Phi(\cdot)$ can be evaluated efficiently. For many interesting convex sets (each of which allows us to handle classes of convex optimization problems), we know how to construct an efficient s.c. barrier. Specifically, the feasible region of many classes of problems that arise in practice is the direct sum of small dimensional convex sets with known s.c. barriers. Let us elaborate more by looking at linear programming.

Consider the 1-dimensional set $\{z \in \mathbb{R} : z \geq \beta\}$ for $\beta \in \mathbb{R}$. It is well-known that $-\ln(z - \beta)$ is a s.c. barrier for this set (actually, the properties of this function motivated the definition of s.c. barriers). Using this simple function, we can construct a s.c. barrier for a polyhedron and solve LP; for $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$, a s.c. barrier for $\{x \in \mathbb{R}^n : Ax \leq b\}$ is $-\sum_{j=1}^m \ln(b_j - a_j^\top x)$, where a_j is the jth row of A. Knowing a s.c. barrier for a 1-dimensional convex set may not seem very useful. However, when we direct sum an arbitrarily large number of these sets, we are able to solve basically every LP problem.

When we add just one degree of freedom to go to $\mathbb{R}^2$, we can construct many interesting convex sets rather than just intervals. Constructing an efficient s.c. barrier is not hard for a structured 2-dimensional set. In this Chapter, we show several examples that are constructed by using univariate convex functions. In the same fashion we explained for LP, each of these 2-dimensional convex sets and its corresponding s.c. barrier can solve an interesting class of optimization problems. The power of this method is accentuated when we consider the possibility of direct summing convex sets of different types. Each set constraint is a block and we can bind an arbitrary number of different blocks by direct sum to construct a problem in the Domain-Driven setup. In this chapter, we show many set constraints/functions as the building blocks of a problem in the Domain-Driven setup. We start by showing that the Domain-Driven setup covers the popular optimization over *symmetric cones*. We express the s.c. barriers corresponding with the convex sets in this chapter, but the formal definitions come in Chapter 4. Many of these functions can be found in Nesterov and Nemirovski's book [52].

For the Domain-Driven setup, we also assume a dual structure that we discuss later in Chapter 5, after defining the *Legendre-Fenchel (LF) conjugate* of a convex function in Chapter 4.

2.1 LP, SOCP, and SDP

Linear programming (LP), second-order cone programming (SOCP), and semidefinite programming (SDP) are the most popular convex optimization problems in the literature and in applications. These optimization problems are associated with polyhedral cones, second-order cones, and positive semidefinite cones, respectively. These cones are called *symmetric* or *self-scaled* [15] and possess some nice properties that make LP, SOCP, and SDP very elegant in theory and practice [15, 70].

Optimization over symmetric cones is a special case of our Domain-Driven setup. Table 2.1 shows the constraints that specify D and the s.c. barrier associated with the convex set defined by the constraint. Here we give an example to understand the connection of

Table 2.1: LP, SOCP, and SDP constraints and the corresponding s.c.b.'s

	Constraint	s.c. barrier Φ	Domain of Φ
LP	$z \le \beta, \quad z, \beta \in \mathbb{R},$	$-\ln(\beta - z)$	$z < \beta$
SOCP	$\|z\| \le t, \quad z \in \mathbb{R}^m, \quad t \in \mathbb{R},$	$-\ln(t^2 - z^\top z)$	$\|z\| < t$
SDP	$Z \preceq B, \quad Z, B \in \mathbb{S}^m$	$-\ln(\det(B - Z))$	$Z \prec B$

formulation (2.1) and Table 2.1. Assume that our feasible region is

$$\left\{ x \in \mathbb{R}^n : a_1^\top x \le \beta_1, \ \|Fx + f\| \le a_2^\top x + \beta_2, \ A_0 + \sum_{i=1}^n A_i x_i \preceq B \right\},$$

for given $a_1, a_2 \in \mathbb{R}^n$, $\beta_1, \beta_2 \in \mathbb{R}$, $F \in \mathbb{R}^{\ell \times n}$, $f \in \mathbb{R}^\ell$, and $A_0, \cdots, A_n, B \in \mathbb{S}^m$. Then, we can define our set D and its s.c. barrier Φ as

$$\begin{aligned} &D := \{(z, h, u, Z) \in \mathbb{R} \oplus \mathbb{R}^\ell \oplus \mathbb{R} \oplus \mathbb{S}^m : z \le \beta_1, \ \|h + f\| \le u + \beta_2, \ Z + A_0 \preceq B\}, \\ &\Phi(z, h, u, Z) := -\ln(\beta_1 - z) - \ln((u + \beta_2) - ((h + f)(h + f)^\top) - \ln(\det(B - (Z + A_0))), \end{aligned}$$

and our linear transformation as

$$Ax = \left(a_1^\top x, Fx, a_2^\top x, \sum_{i=1}^n A_i x_i \right).$$

This example shows how easily we can put a problem with LP, SOCP, and SDP constraints into a Domain-Driven setup. This is basically the method we use to combine all the problems we discuss in this chapter. Note that for implementation, we should represent our linear transformation as one matrix and our variables as one vector. We discuss the implementation part in detail in Chapter 8.

2.2 Direct sum of 2-dimensional sets (geometric programming, entropy programming, and more)

The 2-dimensional sets we are interested in are epigraphs of univariate convex functions. For a convex function $f : C \to \mathbb{R}$, the epigraph is defined as

$$\text{epi}(f) := \{(z, \gamma) \in C \oplus \mathbb{R} : f(z) \leq \gamma\}. \tag{2.2}$$

Figure 2.1 shows the epigraph of a univariate function that is a 2-dimensional convex set. Table 2.2 shows 2-dimensional sets defined by the epigraph of some famous convex

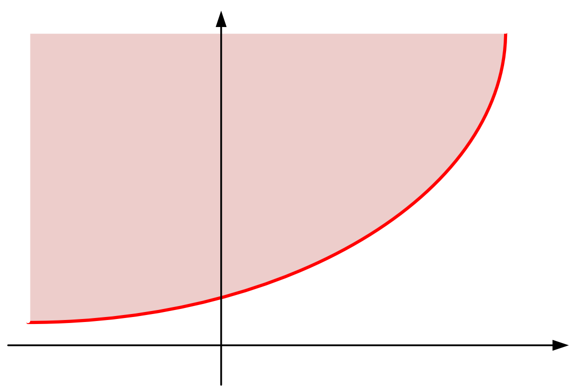

Figure 2.1: Epigraph of a convex function is a convex set.

functions, and a s.c barrier associated to each set. In the same fashion that a s.c. barrier for a 1-dimensional interval lets us solve LP problems, each of these 2-dimensional convex sets describes its own class of optimization models. Every inequality of the form

$$\sum_{i=1}^{\ell} \alpha_i f_i(a_i^\top x + \beta_i) + g^\top x + \gamma \leq 0, \quad a_i, g \in \mathbb{R}^n, \quad \beta_i, \gamma \in \mathbb{R}, \quad i \in \{1, \ldots, \ell\}, \tag{2.3}$$

where $\alpha_i \geq 0$ and $f_i(x)$, $i \in \{1, \ldots, \ell\}$, can be any function from Table 2.2, can be handled in the Domain-Driven setup (we show this in Chapter 5). By using this simple structure, we can model many interesting optimization problems. Geometric programming (GP) [8] and entropy programming (EP) [14] with many applications in engineering are constructed with constraints of the form (2.3) when $f_i(z) = e^z$ for $i \in \{1, \cdots, \ell\}$ and $f_i(z) = z\ln(z)$ for $i \in \{1, \cdots, \ell\}$, respectively. The other functions with p powers let us solve optimization problems related to p-norm minimization.

Table 2.2: Some 2-dimensional convex sets and their s.c. barriers.

	set (z,t)	s.c. barrier $\Phi(z,t)$
1	$-\ln(z) \le t,\ z > 0$	$-\ln(t + \ln(z)) - \ln(z)$
2	$e^z \le t$	$-\ln(\ln(t) - z) - \ln(t)$
3	$z\ln(z) \le t,\ z > 0$	$-\ln(t - z\ln(z)) - \ln(z)$
4	$\lvert z\rvert^p \le t,\ p \ge 1$	$-\ln(t^{\frac{2}{p}} - z^2) - 2\ln(t)$
5	$-z^p \le t,\ z > 0,\ 0 \le p \le 1$	$-\ln(z^p + t) - \ln(z)$
6	$z^p \le t,\ z > 0,\ p \le -1$	$-\ln(z - t^{\frac{1}{p}}) - \ln(t)$
7	$\frac{1}{z} \le t,\ z > 0$	$-\ln(zt - 1)$

Let us write a GP constraint explicitly

$$\sum_{i=1}^{\ell} \alpha_i e^{a_i^\top x + \beta_i} + g^\top x + \gamma \le 0. \tag{2.4}$$

Note that this is the most general form of GP. In [8], a GP constraint is written as

$$\sum_{i=1}^{\ell} \alpha_i y_1^{a_{1i}} \cdots y_n^{a_{ni}} \le 1, \tag{2.5}$$

for positive y_i's that if we define $y_i := e^{x_i}$, $i \in \{1, \ldots, n\}$, then (2.5) becomes a special case of (2.4). Let us give another simple example in two variables:

$$2e^{-x_1+3x_2} + (x_1 - x_2)\ln(x_1 - x_2) + 4x_2 \le 0. \tag{2.6}$$

This example can be easily put into the above setup for $f_1(z) = e^z$ and $f_2(z) = z\ln(z)$. CVX [20], a famous interface for convex optimization, uses approximation to solve these problems. More on this and other examples come in Chapter 8.

2.3 Generalized epigraph of a matrix norm, minimizing nuclear norm

Assume that we have constraints of the form

$$Z - UU^\top \succeq 0, \quad \text{where } \; Z = Z_0 + \sum_{i=1}^{\ell} x_i Z_i, \;\; U = U_0 + \sum_{i=1}^{\ell} x_i U_i. \tag{2.7}$$

Z_i, $i \in \{0, \dots, \ell\}$, are $m \times m$ symmetric matrices, and U_i, $i \in \{0, \dots, \ell\}$, are $m \times n$ matrices. This problem can be modeled as an SDP using the Schur complement theorem as:

$$Z - UU^\top \succeq 0, \quad \Leftrightarrow \quad \bar{Z} := \begin{pmatrix} Z & U \\ U^\top & I_{n\times n} \end{pmatrix} \succeq 0. \tag{2.8}$$

However, the set $\{(Z, U) : Z - UU^\top \succeq 0\}$ accepts the following s.c. barrier:

$$\Phi(Z, U) := -\ln(\det(Z - UU^\top)). \tag{2.9}$$

A parameter is assigned to every s.c. barrier (see Chapter 4), which is directly related to the iteration complexity bounds. The parameter of the s.c. barrier in (2.9) is m. In the cases that $m \ll n$, $Z - UU^\top$ is much smaller than $\bar{Z}$ in (2.8). Therefore, the parameter of the s.c. barrier for (2.9) is also much smaller than n, the parameter of the s.c. barrier if we use the SDP reformulation. This can make a huge difference both in theory and applications, even though $\bar{Z}$ in (2.8) is sparse.

As a special but very important application for constraints of the form (2.7) is minimizing the *nuclear norm*. The nuclear norm of a matrix Z is defined as $\|Z\|_* = \mathrm{Tr}\left((ZZ^\top)^{1/2}\right)$. The dual norm of $\|\cdot\|_*$ is the 2-norm $\|\cdot\|$ of a matrix. It can be shown that the following optimization problems are a primal-dual pair [62].

$$\begin{array}{llll} (P) & \min_X & \|X\|_* \\ & s.t. & AX = b. \end{array} \qquad \begin{array}{llll} (D) & \max_z & \langle b, z\rangle \\ & s.t. & \|A^* z\| \le 1, \end{array} \tag{2.10}$$

where A is a linear operator on matrices and A^* is its conjugate. Problem (P) in (2.10) has become very popular recently as a relaxation of minimizing $\mathrm{rank}(X)$, an optimization problem with applications in big data, machine learning, matrix sparsification, and compressed sensing. The dual problem (D) in (2.10) is a special case of (2.7) where $Z = I$ and $U = A^* z$. As we develop primal-dual techniques, it can be shown that solving (D) immediately gives us the solution for (P).

2.4 Generality of Domain-Driven setup

One strength of the Domain-Driven setup is its generality and versatility. Basically, every convex set with a s.c. barrier can be added to the setup (and be implemented), while the core of the algorithm stays the same. As we explained above, knowing an efficient s.c. barrier for a small dimensional set and using direct sum provides a machinery for solving a class of optimization problems. Therefore, we have an expanding list for types of constraints, where a problem in the Domain-Driven setup can have an arbitrary number of constraints of each type. Even problems that have a conic reformulation can instead be dealt with in their natural form. Another example, in addition to the epigraph of a matrix norm we mentioned above, is a quadratic constraint of the form

$$x^\top Bx + b^\top x + \beta \leq 0, \tag{2.11}$$

where B is a positive semidefinite matrix. This constraint can be reformulated as an SOCP, but we show in Chapter 8 how to use an appropriate s.c. barrier to put it in the Domain-Driven setup and implement it without a conic reformulation.

Chapter 3

Infeasible-start algorithms

Iterative algorithms for solving optimization problems start the process from a *starting point*. No matter where the starting point lies, the final answer must be (approximately) feasible if the feasible region is not empty. Having a starting point that is feasible is desirable for most of the algorithms; however, a feasible point may not be known beforehand. Finding a feasible point is also not “easier” than solving the optimization problem. In many theoretical setups, if we have an oracle to find a feasible point, we can design an efficient algorithm for the optimization problem. In the literature, algorithms are divided into *feasible-start* and *infeasible-start* and there is a clear-cut boundary between them especially when we are considering iteration complexity bounds. A feasible-start algorithm assumes that the given problem is feasible and a feasible point with required properties is given. On the other hand, an infeasible-start algorithm must distinguish the possible statuses for a given problem and decide whether it is solvable, infeasible, or unbounded. In this chapter, we review the popular approaches for handling infeasibility and the pros and cons of them. At the end, we discuss where our approach stands and how strong it is among other methods.

3.1 Linear programming

When we have a feasible-start algorithm, an obvious approach for handling infeasibility is using a two-phase method. The purpose of phase-1 is finding a feasible point for the original problem. Then, phase-2 is just applying the feasible-start algorithm to the original problem. For phase-1, we need a formulation of an auxiliary problem with an obvious feasible starting point. Solving this auxiliary problem by using the feasible-start algorithm either confirms that the original problem is infeasible or returns a feasible point for it. In terms of theoretical complexity bound, with an appropriate reformulation in phase-1, we trivially have an extra factor of 2 compared to the feasible-start version. However, this approach is not desirable in practice and researchers are interested in approaches that gain feasibility and optimality at the same time.

Infeasible-start algorithms have been extensively studied for LP. We review some of the important methods that have been extended to general convex optimization problems. A popular approach is the big-M approach. Let us explain an elementary version for the Simplex method. Assume that we want to solve the LP problem $\min\{c^\top x : Ax = b, x \geq 0\}$, where $A \in \mathbb{R}^{m\times n}$. To start from a feasible basis, we add m artificial variables and construct the following auxiliary LP:

$$\begin{array}{lll} \min & c^\top x + Me^\top x' & \\ \text{s.t.} & (A \;\; I)\begin{bmatrix} x \\ x' \end{bmatrix} & = \;\; b \\ & \quad x, x' & \geq \;\; 0, \end{array} \tag{3.1}$$

where e is the vector of all ones and M is a large enough number. For a hand-wavy argument for why this approach works, assume that the initial LP has an optimal solution x^*. Note that $x' \geq 0$ for all the feasible points of (3.1) and so $e^\top x' \geq 0$. Let t be the smallest nonzero value of $e^\top x'$ over all the extreme points of the feasible region in (3.1). If we take $M > 2|c^\top x^*|/t$, then, for every optimal solution of (3.1) we must have $x' = 0$. The main issue with this approach is that it is not clear beforehand how large M should be to get the desired result, and very large values for M accentuate numerical issues.

We change our focus to *interior-point* algorithms that this thesis is mostly about. Let us rewrite the above LP problem and its dual

$$\begin{array}{lll} \text{(P)} & \min & \{c^\top x : Ax = b, x \geq 0\}, \\ \text{(D)} & \max & \{b^\top y : A^\top y + s = c, s \geq 0\}. \end{array} \tag{3.2}$$

By writing the optimality conditions, a pair (x, s) is primal-dual optimal if they satisfy the following system for $\mu = 0$:

$$\begin{array}{rcll} Ax & = & b, & x \geq 0, \\ A^\top y + s & = & c, & s \geq 0, \\ Xs & = & \mu e, & \end{array} \tag{3.3}$$

where X is a diagonal matrix with the elements of x on the diagonal. Let us define

$$\mathcal{F}_+ \;\; := \;\; \{(x,s): \; Ax = b, \; A^\top y + s = c, \; x > 0, s > 0, y \in \mathbb{R}^m\}. \tag{3.4}$$

If $\mathcal{F}_+$ is not empty, then system (3.3) has a unique solution for every $\mu > 0$ and we denote the solution set as the standard *primal-dual central path*. In the standard feasible-start primal-dual algorithms, we "follow" this path for values of μ that tend to zero for a certain number of iterations. At every iteration, we need a search direction that is calculated by applying a Newton-like method to the equations in system (3.3) with an appropriate value of μ and using the current point as the starting point. When the data (A, b, c) are rational, let L be the size of the given data in the LP (the number of bits required to store the given data). Then, achieving the $O(\sqrt{n}L)$ number of iterations [1] for solving the LP is straightforward for feasible-start algorithms and numerous of them have been studied in the literature. However, getting this bound for infeasible-start algorithms is a different story.

The general idea in infeasible-start algorithms is adding some artificial variables to transform the problem into an auxiliary optimization problem with an obvious feasible point. Then, this feasible point is used to apply a feasible-start algorithm to the modified problem. At the end, based on the values of artificial variables, we determine the status of the problem. The transformation of the initial problem can be done in different ways. The big-M approach has been used in interior-point methods [27, 41, 29, 28, 35, 36] to achieve $O(\sqrt{n}L)$ number of iterations. For arbitrary $x^0 > 0$, $z^0 > 0$, and y^0 initial points, we construct the following primal-dual pair of LPs:

$$\begin{array}{llcl} \min & c^\top x + K_c x_{n+1} & & \\ \text{s.t.} & Ax + (b - Ax^0)x_{n+1} & = & b \\ & (A^\top y^0 + z^0 - c)^\top x + x_{n+2} & = & K_b \\ & x, x_{n+1}, x_{n+2} & \geq & 0. \end{array} \tag{3.5}$$

$$\begin{array}{llcl} \max & b^\top y + K_b y_{m+1} & & \\ \text{s.t.} & A^\top y + (A^\top y^0 + z^0 - c)y_{m+1} + z & = & c \\ & (b - Ax^0)^\top y + z_{n+1} & = & K_c \\ & y_{m+1} + z_{n+2} & = & 0 \\ & z, z_{n+1}, z_{n+2} & \geq & 0. \end{array} \tag{3.6}$$

As can be easily verified, by setting $\bar{x}_{n+1} := 1$ and $\bar{y}_{m+1} := -1$, x_{n+2} and z_{n+1} can be chosen based on K_c and K_b to get feasible points for the modified LPs. For large enough

[1] This bound is the current best that has been achieved by the conventional s.c. barrier for a polyhedral cone we gave in Chapter 2. By using the *universal barrier* given in [52], we get a theoretical limit $O(\sqrt{\text{rank}(A)}L)$ iteration complexity bound. This bound is claimed to be achieved, up to polylogarithmic factors, by Lee and Sidford [30].

values for K_c and K_b (order of 2^{kL}, where L is the size of the LP and k is a small natural number), a primal dual optimal pair can be interpreted by the values of artificial variables to figure out the status of the initial LPs. For example, the authors in [42] showed the following: Let x^* and (y^*, z^*) be optimal solutions for (3.5) and (3.6), respectively. Then

- If $x^*_{n+2} z^*_{n+1} = 0$,
  - If $x^*_{n+2} = 0$ and $z^*_{n+1} = 0$, then restricting x^* and (y^*, z^*) to the initial variables gives primal and dual optimal solutions for the initial problems.
  - If $x^*_{n+2} \neq 0$, then the initial primal problem is infeasible.
  - If $z^*_{n+1} \neq 0$, then the initial primal problem is unbounded.
- If $x^*_{n+2} z^*_{n+1} \neq 0$, then the initial primal problem is either infeasible or unbounded. We need to solve another auxiliary problem to distinguish them.

Even though an $O(\sqrt{n}L)$ iteration complexity bound can be achieved, the big-M approach has implementation issues:

- It is not clear beforehand how large must K_c and K_b be. One way to overcome this issue is enlarging these numbers dynamically during the algorithm [29]; however, we loose the best complexity bound.
- Putting large values in data tends to make the problem ill-conditioned. We ultimately want to run these algorithms on finite precision machines and, as we have experienced thoroughly in our numerical experiments, numerical issues are very critical. Some types of ill-conditioning arise naturally in interior-point methods and the algorithms that avoid unnecessarily large or small values of data are more justified.

There is a method we call *modifying the RHS* that works well in practice (used in a once popular code OB1 [34]), but it has not achieved the $O(\sqrt{n}L)$ iteration complexity bound. The idea is that we start from an "arbitrary" point and, at every iteration, calculate the search direction as the Newton direction for (3.3), i.e., at iteration k, the search direction is calculated by solving

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} d_x \\ d_y \\ d_s \end{bmatrix} = - \begin{bmatrix} Ax^k - b \\ A^\top y^k + s^k - c \\ X^k s^k - \mu e \end{bmatrix}. \tag{3.7}$$

Authors in [26] proved a global convergence result for a version of these algorithms. Zhang [81] proved an $O(n^2L)$ iteration complexity bound for this method, and for some variations the bound was further improved to $O(nL)$, for example by Mizuno [38].

There are types of interior-point algorithms called *potential reduction*. In these methods, a potential function is defined with some connection to the duality gap, and the purpose of the algorithm is to reduce this potential function efficiently. Mizuno-Kojima-Todd designed an infeasible-start potential reduction algorithm [39] for LP. Their purely potential reduction algorithm achieves $O(n^{2.5}L)$ iteration complexity bound and the bound can be improved to $O(nL)$ by adding some centering steps. Seifi and Tunçel [67] designed another infeasible-start potential reduction algorithm with iteration complexity bound $O(n^2L)$.

The most elegant modification of the initial LP is the *homogeneous self-dual embedding* introduced by Ye, Todd, and Mizuno [80]. If we choose $y^{(0)} := 0$, $x^{(0)} := e$, and $s^{(0)} := e$, the homogeneous self-dual embedding is defined as

$$
\begin{array}{lrrrrl}
\min & & & & (n+1)\theta & \\
\text{s.t.} & & Ax & -bt & +\bar{b}\theta & = 0 \\
& -A^\top y & & +ct & -\bar{c}\theta & \geq 0 \\
& b^\top y & -c^\top x & & +\bar{z}\theta & \geq 0 \\
& -\bar{b}^\top y & +\bar{c}^\top x & -\bar{z}t & & = -(n+1) \\
& x \geq 0, & t \geq 0, & & &
\end{array}
$$

where $\bar{b} := b - Ae$, $\quad \bar{c} := c - e$, and $\bar{z} := c^\top e + 1$. The optimal value of this LP is zero and there is an optimal solution $(y^*, x^*, t^*, \theta^* = 0, s^*, \kappa^*)$, such that:

$$
\begin{pmatrix} x^* + s^* \\ t^* + \kappa^* \end{pmatrix} > 0,
$$

which we call a *strictly self-complementary solution*. Now we have the following theorem:

Theorem 3.1.1. [80] *Let $(y^*, x^*, t^*, \theta^* = 0, s^*, \kappa^*)$ be a strictly-self-complementary solution for* (HLP). *Then:*

- (P) *has an optimal solution if and only if $t^* > 0$. In this case, (x^*/t^*) is an optimal solution for* (P) *and $(y^*/t^*, s^*/t^*)$ is an optimal solution for* (D)*;*

- *if $t^* = 0$, then $\kappa^* > 0$, which implies that $c^\top x^* - b^\top y^* < 0$, i.e., at least one of $c^\top x^*$ and $-b^\top y^*$ is strictly less than 0. If $c^\top x^* < 0$ then* (D) *is infeasible; if $-b^\top y^* < 0$ then* (P) *is infeasible; and if both $c^\top x^* < 0$ and $-b^\top y^* < 0$ then both* (P) *and* (D) *are infeasible.*

This theorem shows that if we use a feasible-start algorithm that returns a strictly self-complementary solution, we can immediately solve both of the problems (P) and (D). By "solving" we mean deciding on the feasibility patterns that can happen to the primal and dual problems. The homogeneous self-dual embedding gives a nice and elegant approach to solve LPs and it also works very well in practice. Note that the above issues with the big-M approach no longer exist. Different versions of the homogeneous self-dual embedding have been designed and implemented, see for example [78].

We mentioned that good solvers for conic optimization have been restricted to optimization over symmetric cones. There are methods (such as using Jordan algebra) to unify LP, SOCP, and SDP into a general setup, and because of similarity of SOCP and SDP with LP, the extension of many infeasible-start algorithms for LP to this general setup is intuitive (see for example [71]). Some of the good solvers for SDP such as SeDuMi [69], MOSEK [43], and SDPT3 [72, 73] use an extension of the homogeneous self-dual embedding to deal with infeasibility. We aim to solve a general convex optimization problem and in the next section we review some methods for conic optimization.

3.2 General convex optimization

We can interpret all the above methods for LP as lifting the feasible region to a higher dimensional space such that the original feasible region is a face of a convex polyhedron with a known interior point. The same idea can be used for a general convex optimization problem if we replace the polyhedron with a general convex set. The path from feasible-start to infeasible-start algorithms is more demanding for general convex optimization. Many types of *ill-conditioning* (including those that cannot exist for linear programming) are possible for a general convex optimization problem such as the case where both primal and dual are feasible while the duality gap is not zero. Ill-conditioned problems are unstable and a small perturbation makes them well behaved. However, a good approach (and code) must still strive to determine the status of the problem as rigorously as possible. For a starting discussion on the possible feasibility patterns see [52]–Section 4.2.2. or [32, 33].

Let us consider the case of conic optimization problems. Consider a primal-dual conic optimization setup

$$\begin{array}{lll} \text{(P)} & \inf & \{\langle c, z\rangle : Az = b, z \in K\}, \\ \text{(D)} & \inf & \{\langle b, y\rangle : s := c + A^* y \in K^*\}, \end{array} \tag{3.8}$$

where K^* is the dual cone of K we define later in Chapter 4. Nesterov in [48] defined a convex set as the solution of the following system

$$\begin{array}{rcl} Az & = & Az^0 + \tau b, \\ s & = & s^0 + \tau c + A^\top y, \\ \langle c, z\rangle + \langle b, y\rangle & = & \langle c, z^0\rangle, \\ & & z \in K, s \in K^*, \tau \in \mathbb{R}_+. \end{array} \tag{3.9}$$

His approach for finding an approximate solution for (3.8) was to find a recession direction for the convex set defined in (3.9). Note that $\langle c, z\rangle + \langle b, y\rangle$ is the conic duality gap. Assume that we have a point satisfying (3.9) with a large $\tau > 0$. Then, $(z/\tau, s/\tau)$ approximately satisfies all the optimality conditions, and if τ tends to infinity, they converge to primal-dual optimal solutions. Nesterov [48] found a recession direction for (3.9) by minimizing a *self-concordant barrier* over this set. We define and analyze self-concordant barriers and functions in Chapter 4. Nesterov's approach in [48] uses two parameters in addition to the primal and dual variables; τ that is defined in (3.9) and μ is used to parameterize the primal-dual central path.

In view of the homogeneous self-dual embedding for LP, Nesterov, Todd, and Ye [56] generalized the approach in [48]. Their work, as far as we know, is the strongest and most comprehensive result for infeasible-start interior-point methods for conic optimization. They added another variable κ and for arbitrary starting points $z^0 \in \text{int}K$, $s^0 \in \text{int}K^*$, $y^0 \in \mathbb{R}^m$, and $\tau^0, \kappa^0 > 0$, defined the following convex set (we use matrix form to highlight the self-dual structure)

$$\begin{bmatrix} 0 & A & -b \\ -A^\top & 0 & c \\ b^\top & -c^\top & 0 \end{bmatrix} \begin{bmatrix} y \\ z \\ \tau \end{bmatrix} - \begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} Az^0 - \tau^0 b \\ -A^\top y^0 - s^0 + \tau^0 c \\ -\langle c, z^0\rangle + \langle b, y^0\rangle - \kappa^0 \end{bmatrix}$$
$$z \in K, s \in K^*, \tau, \kappa \in \mathbb{R}_+. \tag{3.10}$$

The authors solved the optimization problem again by finding a recession direction of this convex set by minimizing a self-concordant barrier over it. Note that analyzing the outcomes of the algorithms is much more complicated in general convex optimization compared to LP. Several possible statuses were defined in [48] and [56], and the ability of the algorithms in detecting each of them was analyzed. The feasibility patterns defined in [56] are: solvable, strictly primal-dual feasible, strictly and strongly primal infeasible, strictly and strongly dual infeasible, and ill-conditioned.

We finish this chapter by reviewing another interesting result [53] that is close to our approach in terms of formulating the given problem. Under some mild conditions, we can use a big-M approach to reformulate our infeasible-start problem in a feasible-start one written as

$$\min\{c^\top x + M x_n : x \in G\}, \tag{3.11}$$

where G is a convex set with a known interior point and we need x_n to be zero to get feasibility for the initial problem. Typically, we can define a feasible central path for the initial problem, parametrized by μ, as the solution set of $-\nabla F(x) = \mu c$ for an appropriate function F. This path becomes $-\nabla F(x) = \mu(c + Mf)$ for (3.11), where f is a vector that satisfies $f^\top x = x_n$. What if we replace μM with another parameter $\bar{\mu}$ and look at a surface defined by $-\nabla F(x) = \mu c + \bar{\mu} f$? Nesterov and Nemirovski [53] took this idea to introduce multi-parameter surfaces of analytic centers and use them to design surface-following interior-point algorithms. The idea and analysis are elegant and they proved the current best complexity bound attainable for conic optimization. However, the whole approach is complicated and it seems hard to implement in practice.

3.3 Properties of our approach

Our infeasible start approach does not require any additional large auxiliary constants and can start basically at arbitrary points. From one perspective, this method lies in the "modifying the RHS" category that we explained for LP. However, if we look at the conic reformulation of a problem in the Domain-Driven setup (see Appendix A), our central paths and algorithms have close connections with those in [48, 56] for general conic optimization, and our main objective in this research project has been to do at least as well as the current best approaches for the conic setup. The statuses we define and analyze for a problem in the Domain-Driven setup are similar to the ones in [56] (the most comprehensive we know) for general conic optimization, and we achieve the current best iteration complexity bounds, which are new for the "modifying the RHS" setup even for LP. Similar to [48], we add an auxiliary variable τ and we add μ to parameterize the primal-dual central path. We also have a simple algorithm in which we only add τ and it acts as the parameter of the central path. We have not been able to prove the best theoretical results for this simpler algorithm; however, it has some nice properties such as long-step property and is easier to analyze and understand. We explain this simpler algorithm briefly in Appendix B.

Chapter 4

Convex optimization and self-concordant functions

In this chapter, we present some of the fundamental results about a special class of convex functions that we use frequently in this thesis. Especially we express and sometimes prove several results and properties about self-concordant functions and barriers. These properties form the machinery for our discussions in later chapters. In comparison to many papers in this context, we use a wider range of properties in our development of the Domain-Driven setup. The reason is that we design primal-dual methods for general convex optimization problems. Primal-dual interior-point methods have been mostly studied for conic optimization. Convex cones attain special s.c. barriers with stronger properties, as we explain later in this chapter. These stronger properties include a primal-dual "symmetry", and a stronger calculus that makes the design and analysis of the algorithms easier. We give up some of these properties to work with the "natural" domain of a given problem.

4.1 Convex optimization

We defined a convex set in Chapter 1. Let us see more definitions and results. The definitions and proofs in this section that we state without reference can be found in many books on convex analysis and optimization, such as, Hiriart-Urruty and Lemaréchal's book [23], Schneider's book [66], Rockafellar's book [64], Rockafellar and Wets' book [65], and Boyd and Vandenberghe's book [9].

As we mentioned before, for our purposes in this thesis, convex optimization is the problem of minimizing a linear function over a convex set. Given a convex set D, a linear functional $\langle c, \cdot \rangle$ on $\mathbb{E}$, and a linear mapping $A : \mathbb{E} \to \mathbb{Y}$, a *convex optimization problem* is a problem that can be formulated as (2.1). The problems that arise in practice are usually given in another form that uses convex functions. Let $f_i : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$, $i \in \{0, 1, \cdots, k\}$, be convex functions and $\mathcal{L} \subseteq \mathbb{E}$ be an affine subspace. Then the problem

$$\begin{array}{ll} \min & f_0(x) \\ s.t. & x \in \mathcal{L} \\ & f_i(x) \leq 0, \quad i \in \{1, \cdots, k\}, \end{array} \tag{4.1}$$

is a convex optimization problem. It can be shown that the above two definitions are equivalent.

Definition 4.1.1. *Let $f : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ be convex and consider a point $\bar{x} \in \mathrm{dom} f$, where the domain* dom$f$ *is the set of x such that $f(x) < +\infty$. The vector $y \in \mathbb{E}^*$ that satisfies*

$$f(x) \geq f(\bar{x}) + \langle y, x - \bar{x} \rangle, \quad \forall x \in \mathbb{E}, \tag{4.2}$$

is called a subgradient *of f at $\bar{x}$. The set of all subgradients of f at $\bar{x}$ is called* subdifferential *and is denoted by $\partial f(\bar{x})$.*

To every convex function, we assign another "dual" convex function as follows.

Definition 4.1.2. *Let $f : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ be convex. The* Legendre-Fenchel conjugate *of f is defined as*

$$f_*(y) := \sup_x \{\langle y, x \rangle - f(x)\}. \tag{4.3}$$

f_* is always a convex function and its domain is all the points that (4.3) has a bounded solution. For a proper convex function, we have $(f_*)_* = f$ if and only if the epigraph of f is closed (f is a closed convex function), see for example [23]. We use the following inequality frequently in this thesis.

Theorem 4.1.1. (Fenchel-Young inequality) *Let $f : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ be a convex function and f_* be its Legendre-Fenchel conjugate. For every point x in the domain of f and every y in the domain of f_*, we have*

$$f(x) + f_*(y) \geq \langle y, x \rangle. \tag{4.4}$$

Equality holds if and only if $y \in \partial f(x)$.

Assume that $f(x)$ is differentiable and the optimal value of (4.3) for $\bar{y}$ is attained at $\bar{x}$, then we must have $\bar{y} = f'(\bar{x})$. By Theorem 4.1.1, if both f and f_* are twice differentiable, for every point x in the domain of f we have

$$x = f_*'(f'(x)) \quad \Rightarrow \quad f_*''(f'(x)) = [f''(x)]^{-1}. \tag{4.5}$$

The Legendre-Fenchel conjugate is important to develop the concept of *duality* in convex optimization. Duality theory provides us with bounds and certificates in convex optimization. The following proposition is one version of *weak duality* when we only have linear equality constraints.

Proposition 4.1.1. ([6]-Corollary 3.3.11) *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a convex function, f_* be its Legendre-Fenchel conjugate, $A : \mathbb{R}^n \to \mathbb{R}^m$ be a linear mapping, and b be any element in $\mathbb{R}^n$. Then,*

$$\inf_{x \in \mathbb{R}^n} \{f(x) : Ax = b\} \geq \sup_{y \in \mathbb{R}^m} \{\langle y, b\rangle - f_*(A^* y)\}. \tag{4.6}$$

Moreover, if $b \in \operatorname{int}\{Ax : f(x) < +\infty\}$, then equality holds and the supremum is attained.

We defined a primal-dual conic optimization setup in (3.8) without giving some definitions, here they are

Definition 4.1.3. *Let $K \subseteq \mathbb{E}$ be a closed convex cone. K is said to be* pointed *if there is no $v \in K \setminus \{0\}$ such that $v \in K$ and $-v \in K$.*

Definition 4.1.4. *Let $C \subseteq \mathbb{R}^n$ be a convex set. The* polar *of C is defined by*

$$C^\circ := \{h \in \mathbb{R}^n : \forall x \in C, \langle h, x\rangle \leq 1\}.$$

If $C = K$ is a convex cone, its dual cone *is defined by*

$$K^* := \{h \in \mathbb{R}^n : \forall x \in K, \langle h, x\rangle \geq 0\}.$$

Theorem 4.1.2. *For every set C, C° is convex. If C is a closed convex set that contains 0, then $C = (C^\circ)^\circ$.*
For every cone K, K^* is a convex cone. K is a closed convex cone iff $K = (K^*)^*$.

4.2 Self-concordant functions

In this section, we define different classes of self-concordant (s.c.) functions and summarize (sometimes prove) their fundamental properties we frequently use in this thesis. First, we start from the strongest one that is logarithmically homogeneous (LH) s.c. barrier defined for convex cones. Motivation for the definition comes from $f(z) = -\ln(z)$ defined on $\mathbb{R}_{++}$. The derivatives of this function have a nice pattern:

$$f'(z) = -\frac{1}{z}, \quad f''(z) = \frac{1}{z^2}, \quad f'''(z) = -\frac{2}{z^3},$$

and we have $|f'''(z)| = 2(f''(z))^{3/2}$ for all $z \in \mathbb{R}_{++}$. It also behaves nicely by positive scaling of the argument:

$$f(tz) = f(z) - \ln(t), \quad \forall t > 0.$$

These are the properties that motivate the definition of a LH s.c. barrier:

Definition 4.2.1. *Let K be a pointed closed convex cone in $\mathbb{E}$. A function $f : \text{int}K \to \mathbb{R}$ that is $\mathcal{C}^3$ is called a ϑ-*logarithmically homogeneous (LH) s.c. barrier *if*

(i) *$f(x_i) \to +\infty$ for every sequence $\{x_i\} \subset \text{int}K$ that converges to a point on the boundary of K.*

(ii) *(Self-concordance) For every $x \in \text{int}K$, we have*

$$|f'''(x)[h,h,h]| \leq 2(f''(x)[h,h])^{3/2}, \quad \forall (h \in \mathbb{E}). \tag{4.7}$$

(iii) *(ϑ-logarithmic homogeneity) There exists $\vartheta \geq 1$ such that for every $x \in \text{int}K$, we have*

$$f(tx) = f(x) - \vartheta \ln(t), \quad \forall (t > 0). \tag{4.8}$$

By our above discussion, $f(z) = -\ln(z)$ is a 1-LH s.c. barrier for $\mathbb{R}_+$. Another example is $f : \mathbb{R}^k \oplus \mathbb{R} \to \mathbb{R}$ that is defined as $f(z,t) = -\ln(t^2 - z^\top z)$, and is a 2-LH s.c. barrier for the second-order cone. Self-concordance, the most critical property, is similar to a Lipschitz continuity constraint between the third and second derivative. Intuitively, this property implies that the Hessian of the function is changing smoothly; a useful property in the analysis of optimization algorithms. A convex function is defined to be *self-concordant* if it has properties (i) and (ii) in Definition 4.2.1 (we define self-concordance in a slightly more general way).

If we take the derivative from both sides of (4.8) with respect to x or t, we can get different properties such as [52] (for every $x \in \text{int}K$):

$$
\begin{aligned}
f'(tx) = \frac{1}{t} f'(x), \qquad & f''(tx) = \frac{1}{t^2} f''(x), \quad \forall t \in \mathbb{R}_{++}, \\
f''(x)x = -f'(x), \qquad & \langle f'(x), x \rangle = -\vartheta, \\
\langle f''(x)x, x \rangle \;\; = \;\; & \langle f'(x), [f''(x)]^{-1} f'(x) \rangle = \vartheta. \qquad (4.9)
\end{aligned}
$$

Let us focus on the last equation $\langle f'(x), [f''(x)]^{-1} f'(x) \rangle = \|[f''(x)]^{-1} f'(x)\|^2_{f''(x)} = \vartheta$. The term in the norm is the conventional Newton direction. This equality implies that the *local norm* of the Newton direction is bounded by (here is equal to) a constant number at *every* point in the domain. This property is very useful for having efficient Newton-type algorithms. A s.c. function is a *ϑ-s.c. barrier* if it also has a property similar to this local norm bound on the Newton step for a $\vartheta \geq 1$. (4.9) shows that we have this property for free by logarithmic homogeneity.

A very strong property that confirms part of the symmetry we mentioned above for conic optimization is about the LF conjugate of a LH s.c. barrier. [52]-Theorem 2.4.4 proves that if $f(x)$ is a ϑ-LH s.c. barrier, $f_*(-s)$ is also a ϑ-LH s.c. barrier for $\text{int}K^*$, where K^* is the dual cone of K. This property does not hold for a general s.c. barrier. Let us mention some of the relations between f and f_* [52]: For all $x \in \text{int}K$ and $s \in \text{int}K^*$:

$$
\begin{aligned}
-f'(x) \in \text{int}K^*, \qquad & f'_*(-s) \in \text{int}K, \\
f_*(f'(x)) = -\vartheta - f(x), \qquad & f(f'_*(-s)) = -\vartheta - f_*(-s), \\
f'_*(f'(x)) = x, \qquad & f'(f'_*(-s)) = -s, \\
f''(f'_*(s)) = [f''_*(-s)]^{-1}, \qquad & f''_*(f'(x)) = [f''(x)]^{-1}.
\end{aligned}
$$

These are part of the strong primal-dual properties for LH s.c. barriers. These properties lead to a very strong and elegant machinery in the design and analysis of interior-point algorithms for convex optimization problems in conic form. We lose many of these properties when we work with the given domain of the problem and avoid enforcing into a conic reformulation.

In this section, we summarize the properties of self-concordant (s.c.) functions and s.c. barriers. We also have a subsection for Legendre-Fenchel conjugate of s.c. barriers. The LF conjugate of a s.c. barrier is a s.c. function, but not necessarily a s.c. barrier, which makes our analysis even harder. However, the LF conjugate of a s.c. barrier has some special properties that we use several times in this thesis; we highlight them in a separate subsection.

We define the following function that is frequently used in the context of self-concordant functions.

$$\rho(t) := t - \ln(1+t) = \frac{t^2}{2} - \frac{t^3}{3} + \frac{t^4}{4} + \cdots . \tag{4.10}$$

We also need, in some sense, the inverse of this function

$$\sigma(s) := \max\{t : \rho(t) \leq s\},\ s \geq 0. \tag{4.11}$$

By [46]-Lemma 2.1, we have

$$\sigma(s) \leq \sqrt{2s} + s, \quad \forall s \geq 0. \tag{4.12}$$

4.2.1 Self-concordant (s.c.) functions

A convex function $f : \mathbb{E} \to \mathbb{R} \cup \{+\infty\}$ is called a-s.c. function if its domain Q is open, f is $\mathcal{C}^3$ on Q and

(i) $f(x_i) \to +\infty$ for every sequence $\{x_i\} \subset Q$ that converges to a point on the boundary of Q.

(ii) There exists a positive real constant a such that

$$|f'''(x)[h,h,h]| \leq 2a^{-1/2}(f''(x)[h,h])^{3/2} = 2a^{-1/2}\|h\|^3_{f''(x)}, \quad \forall (x \in Q, h \in \mathbb{E}), \tag{4.13}$$

where $f^k(x)[h_1, \ldots, h_k]$ henceforth is the value of the kth differential of f along directions $h_1, \ldots, h_k \in \mathbb{E}$.

We say that f is *non-degenerate* if its Hessian $f''(x)$ is positive definite at a point (and then it can be proved to be positive definite at all points) in Q.

For a a-s.c. function f and any point x in its domain, we define an important concept of the *Newton decrement* of f at x as

$$\lambda(f, x) := a^{-1/2} \max\{f'(x)[h] : h \in \mathbb{E}, f''(x)[h,h] \leq 1\}. \tag{4.14}$$

When f is non-degenerate, it can be shown that we have

$$\lambda(f, x) = a^{-1/2}\|f'(x)\|^*_{f''(x)}. \tag{4.15}$$

In the following, we list some of the important properties of s.c. functions and s.c. barriers. Properties are labeled with SC for future reference:

SC-1 **(Stability under intersections, direct sums, and affine maps)** [52]-Proposition 2.1.1:

(a) Let f_i, $i \in \{1, \dots, m\}$, be an a_i-s.c. function on $\mathbb{E}$ with domains Q_i. Then, for real coefficients $\gamma_i \geq 1$, if $Q := \cap_{i=1}^m Q_i$ is not empty, $f := \sum_{i=1}^m \gamma_i f_i$ is an a-s.c. function with domain Q, where $a := \min\{\gamma_i a_i : i \in \{1, \dots, m\}\}$.

(b) Let f_i, $i \in \{1, \dots, m\}$, be an a-s.c. function on $\mathbb{E}_i$ with domains Q_i. Then, the function $f(x^1, \dots, x^m) := \sum_{i=1}^m f_i(x^i)$, defined on $Q_1 \oplus \cdots \oplus Q_m$, is an a-s.c. function.

(c) Let f be a s.c. function with domain Q and $x = Ay + b$ be an affine mapping with image intersecting Q, then $f(Ay + b)$ is also a s.c. function on $\{y : Ay + b \in Q\}$.

From now on, we assume that f is a s.c. function with domain Q.

SC-2 **(Behaviour in Dikin ellipsoid and some basic inequalities):**

(a) For every point $x \in Q$, we define the *Dikin ellipsoid* centered at x as

$$W_1(x) := \{y \in \mathbb{E} : \frac{1}{\sqrt{a}} \|y - x\|_{f''(x)} \leq 1\}.$$

Then we have $W_1(x) \subset Q$ and for every point $y \in W_1(x)$ we can estimate the Hessian of f at y in term of the Hessian of f at x as

$$(1 - r)^2 f''(x) \preceq f''(y) \preceq \frac{1}{(1 - r)^2} f''(x), \tag{4.16}$$

where $r := \frac{1}{\sqrt{a}} \|y - x\|_{f''(x)}$. For a proof see [52]-Theorem 2.1.1.

(b) For every point $x, y \in Q$ and for $r := \frac{1}{\sqrt{a}} \|y - x\|_{f''(x)}$, we have

$$\begin{array}{rcl} f(y) & \geq & f(x) + \langle f'(x), y - x \rangle + \rho(r), \\ f(y) & \leq & f(x) + \langle f'(x), y - x \rangle + \rho(-r), \quad \text{if} \quad r < 1, \end{array} \tag{4.17}$$

where $\rho(\cdot)$ is defined in (4.10). For a proof of this for $a = 1$, please see Lecture notes [45], a simplified version of [52]. The proof for general a is similar.

(c) Let $r := \frac{1}{\sqrt{a}} \|y - x\|_{f''(x)}$, then, for every point $x, y \in Q$, we have (for $a = 1$ see [50])

$$\langle f'(x) - f'(y), y - x \rangle \geq \frac{r^2}{1 + r}. \tag{4.18}$$

If $r < 1$, then

$$\langle f'(x) - f'(y), y - x \rangle \leq \frac{r^2}{1 - r}. \tag{4.19}$$

SC-3 **(Newton iterate):** For every point x, we define the *Newton direction* as

$$\text{Newton}(x) := \text{argmin}_h \left\{ f(x) + f'(x)[h] + \frac{1}{2} f''(x)[h,h] \right\}.$$

Then, we define the *damped Newton iterate* of x as

$$x^+ = x + \frac{1}{1+\lambda(f,x)} \text{Newton}(x). \tag{4.20}$$

We have the following properties for a damped Newton step

$$\begin{array}{ll} \text{(a)} & x^+ \in Q, \\ \text{(b)} & f(x^+) \leq f(x) - a\rho(\lambda(f,x)), \\ \text{(c)} & \lambda(f,x^+) \leq 2\lambda^2(f,x). \end{array} \tag{4.21}$$

For parts (a) and (b), see [52]-Proposition 2.2.2. For part (c), plug in $s = \frac{1}{1+\lambda}$ in [52]-Theorem 2.2.1.

SC-4 **(Existence of minimizer):** f attains its minimizer on Q if and only if f is bounded below on Q, and if and only if there exists $x \in Q$ such that $\lambda(f,x) < 1$. For an arbitrary minimizer x_f, we have

$$\lambda(f,x) < 1 \;\Rightarrow\; f(x) - f(x_f) \leq \rho(-\lambda(f,x)). \tag{4.22}$$

If f is non-degenerate, the minimizer is unique. The proof of $a=1$ is in [45], and the proof for general a is similar.

SC-6 **(Legendre-Fenchel conjugate of a s.c. function):** We defined the LF conjugate of a convex function f in (4.3). Let Q_* be the domain of f_*; the set of all points for which the right hand side of (4.3) is finite. We mentioned that Q_* is convex and f_* is a convex function on Q_*. It is shown in [52]- Section 2.4 that $Q_* = f'(Q)$, f_* is a non-degenerate s.c. function and the Legendre-Fenchel conjugate of f_* is exactly f.

The following lemma is very useful when our feasible region is made up from direct sum of building blocks.

Lemma 4.2.1. *Let $D = \oplus_{i=1}^n D_i$ be the direct sum of D_i's and let f_i be a self-concordant function for $D_i \subseteq \mathbb{E}_i$, $i \in \{1,\dots,n\}$ with the Legendre-Fenchel conjugate f_{i*}. Then $f(x^1,\dots,x^n) := \sum_{i=1}^n f_i(x^i)$ is a self-concordant function for D, and we have*

$$f_*(s^1,\dots,s^n) = \sum_{i=1}^n f_{i*}(s^i).$$

The following lemma is also very useful:

Lemma 4.2.2. *Let f be a 1-s.c. function and x and y in its domain such that $r := \|x-y\|_{f''(x)} < 1$. Then*

$$\frac{r}{1+r} \leq \|f'(x) - f'(y)\|^*_{f''(x)} \leq \frac{r}{1-r}. \tag{4.23}$$

Proof. Let us define $q := y - x$. Starting with the fundamental theorem of calculus, we have:

$$\begin{aligned}
\|f'(x) - f'(y)\|^*_{f''(x)} &= \left\| \int_0^1 f''(x+tq)q dt \right\|^*_{f''(x)} \\
&\leq \int_0^1 \|f''(x+tq)q\|^*_{f''(x)} dt \\
&\leq \int_0^1 \frac{1}{1 - \|tq\|_{f''(x)}} \|f''(x+tq)q\|^*_{f''(x+tq)} dt, \quad \text{by (4.16)} \\
&= \int_0^1 \frac{1}{1 - \|tq\|_{f''(x)}} \|q\|_{f''(x+tq)} dt \\
&= \left(\int_0^1 \frac{1}{(1-tr)^2} dt \right) r = \frac{r}{1-r}.
\end{aligned}$$

The other direction is an immediate consequence of (4.18) and Cauchy-Schwarz inequality. □

4.2.2 Self-concordant (s.c.) barriers

For a $\vartheta \geq 1$, we say that a 1-s.c. function is a ϑ-s.c. barrier for $\mathrm{cl}(Q)$ if we have

$$|f'(x)[h]| \leq \sqrt{\vartheta} \|h\|_{f''(x)}, \quad \forall (x \in Q, h \in \mathbb{E}). \tag{4.24}$$

In view of definition (4.14), a non-degenerate s.c. function f is a ϑ-s.c. barrier if and only if

$$\lambda(f, x) = \|f'(x)\|_{[f''(x)]^{-1}} \leq \sqrt{\vartheta}, \quad \forall x \in Q. \tag{4.25}$$

SCB-1 **(Stability under intersections, direct sums, and affine maps)** [52]-Proposition 2.3.1:

(a) Assume that for each $i \in \{1,\dots,m\}$, f_i is a ϑ_i-s.c. barrier on $\mathbb{E}$ with domains Q_i, and consider real coefficients $\gamma_i \geq 1$. If $Q := \cap_{i=1}^m Q_i$ is not empty, then $f := \sum_{i=1}^m \gamma_i f_i$ a $(\sum_{i=1}^m \gamma_i \vartheta_i)$-s.c. barrier on Q.

(b) Let f_i, $i \in \{1,\dots,m\}$, be a ϑ_i-s.c. barrier on $\mathbb{E}_i$ with domains Q_i. Then, the function $f(x^1,\dots,x^m) := \sum_{i=1}^m f_i(x^i)$, defined on $Q := Q_1 \oplus \cdots \oplus Q_m$, is a $(\sum_{i=1}^m \vartheta_i)$-s.c. barrier on Q.

(c) Let f be a ϑ-s.c. barrier with domain Q and $x = Ay + b$ be an affine mapping with image intersecting Q, then $f(Ay+b)$ is also a ϑ-s.c. barrier on $\{y : Ay + b \in Q\}$.

SCB-2 **(Basic properties of s.c. barrier's):** Let us define the *Minkowski function* $\pi_y : \mathbb{E} \to \mathbb{R}_+ \cup \{+\infty\}$ of G at point $y \in G$ as

$$\pi_y(x) := \inf\left\{\frac{1}{\mu} : \mu \geq 0, y + \mu(x-y) \in G\right\}. \tag{4.26}$$

Figure 4.1 shows how the function works for two points x and y in G.

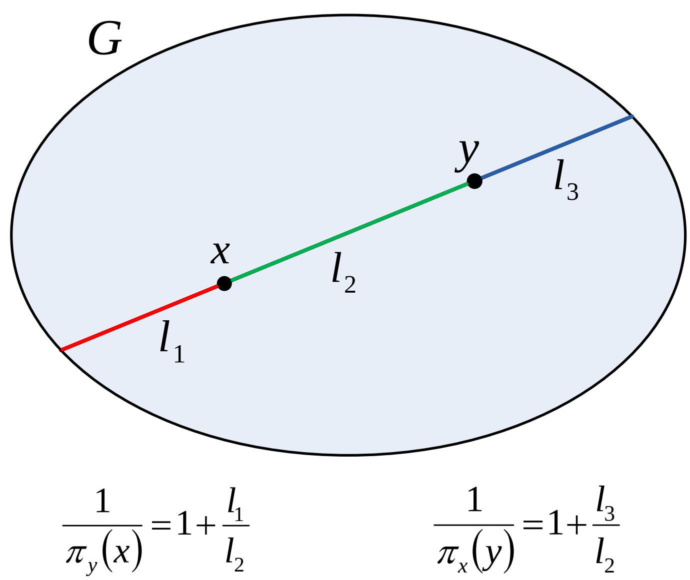

Figure 4.1: How the Minkowski function works for two points x and y in G.

Let f be a ϑ-s.c.b., then the following inequalities hold for every pair $x, y \in Q$ (see [52]-Proposition 2.3.2 and [45]-Chapter 3):

$$f'(x)[y-x] \leq \vartheta; \tag{4.27}$$

where, as before, $f'(x)[h]$ is the first order differential of f taken at x along the direction h.

$$f''(x) \preceq \left(\frac{\vartheta + 2\sqrt{\vartheta}}{1-\pi_y(x)}\right)^2 f''(y); \tag{4.28}$$

$$\langle y - x, f'(x)\rangle \geq 0 \;\Rightarrow\; \|y-x\|_{f''(x)} \leq \vartheta + 2\sqrt{\vartheta}; \tag{4.29}$$

$$f(x) \leq f(y) + \vartheta \ln\left[\frac{1}{1-\pi_y(x)}\right]; \tag{4.30}$$

$$f(x) \geq f(y) + \langle f'(y), x-y\rangle + \ln\left[\frac{1}{1-\pi_y(x)}\right] - \pi_y(x). \tag{4.31}$$

SCB-3 **(Non-degeneracy, attaining minimizer):** f is non-degenerate if and only if Q does not contain lines. f is bounded below if and only if Q is bounded. Then, f is non-degenerate and attains its unique minimizer x_f on Q, and we have

$$\left\{y : \|y-x_f\|_{f''(x_f)} < 1\right\} \subseteq Q \subseteq \left\{y : \|y-x_f\|_{f''(x_f)} < \vartheta + 2\sqrt{\vartheta}\right\}. \tag{4.32}$$

Let us prove the following lemma that is crucial in this thesis.

Lemma 4.2.3. *Let $\Phi(\cdot)$ be a ϑ-s.c. barrier with domain $\text{int}D \subset \mathbb{E}$, and $\xi > 1$. Then, the function $\Phi\left(\frac{z}{\tau}\right) - \xi\vartheta\ln(\tau)$ with domain $\{(z,\tau) : \tau > 0, \frac{z}{\tau} \in \text{int}D\}$ is a $\bar{\xi}$-s.c. function for an absolute constant $\bar{\xi}$ depending on ξ. Moreover, its LF conjugate and also the summation of $\Phi\left(\frac{z}{\tau}\right) - \xi\vartheta\ln(\tau)$ with its LF conjugate are also $\bar{\xi}$-s.c. functions.*

Proof. Consider the function $\Phi(\frac{z}{\tau}) - \xi\vartheta\ln(\tau)$. First we show that the function is convex. Let us define

$$g(\alpha) := \Phi\left(\frac{z+\alpha d_z}{\tau + \alpha d_\tau}\right) - \xi\vartheta\ln(\tau + \alpha d_\tau).$$

Then, we have

$$g''(0) \;=\; \frac{1}{\tau^2}\left[\langle d_z - \frac{d_\tau}{\tau}z, \Phi''\left(\frac{z}{\tau}\right)\left(d_z - \frac{d_\tau}{\tau}z\right)\rangle + 2d_\tau\langle\Phi'\left(\frac{z}{\tau}\right), d_z - \frac{d_\tau}{\tau}z\rangle + \xi\vartheta d_\tau^2\right].$$

Note that because Φ is a ϑ-s.c.b., by definition (4.24), we have

$$\left|\langle \Phi'\left(\frac{z}{\tau}\right), d_z - \frac{d_\tau}{\tau} z\rangle\right| \leq \sqrt{\vartheta}\left\|d_z - \frac{d_\tau}{\tau} z\right\|_{\Phi''(z/\tau)}.$$

Substituting this and doing some simple algebra we get

$$g''(0) \geq \frac{1}{\tau^2}\left[\left\|d_z - \frac{d_\tau}{\tau} z\right\|_{\Phi''} - d_\tau\sqrt{\vartheta}\right]^2 + (\xi - 1)\frac{d_\tau^2}{\tau^2}\vartheta. \tag{4.33}$$

(4.33) shows that $\Phi(\frac{z}{\tau}) - \xi\vartheta\ln(\tau)$ is strictly convex for every $\xi > 1$.

To prove that it is a s.c. function, we show that there exists an absolute constant $\bar{\xi}$ depending on ξ such that

$$|g'''(0)| \leq 2\bar{\xi}^{-1/2}(g''(0))^{3/2}.$$

For simplicity, let us define $h := \frac{1}{\tau}\left(d_z - \frac{d_\tau}{\tau} z\right)$. First, note that from (4.33) we have

$$\begin{aligned} \left|\frac{d_\tau}{\tau}\sqrt{\vartheta}\right| &\leq \frac{\sqrt{g''(0)}}{\sqrt{\xi - 1}}, \\ \|h\|_{\Phi''} &\leq \sqrt{g''(0)} + \left|\frac{d_\tau}{\tau}\sqrt{\vartheta}\right| \leq \underbrace{\left(1 + \frac{1}{\sqrt{\xi - 1}}\right)}_{=:\gamma}\sqrt{g''(0)}. \end{aligned} \tag{4.34}$$

By expanding the expression for $g'''(0)$, we have

$$g'''(0) = \Phi'''[h, h, h] + 3\Phi''[h, h]\left(\frac{d_\tau}{\tau}\right) + 2\Phi'[h]\left(\frac{d_\tau}{\tau}\right)^2 + 2\xi\vartheta\left(\frac{d_\tau}{\tau}\right)^3. \tag{4.35}$$

Because Φ is a 1-s.c. function, by definition in (4.13), we have $|\Phi'''[h, h, h]| \leq 2(\Phi''[h, h])^{3/2} = 2(\|h\|_{\Phi''})^3$, and because Φ is a ϑ-s.c barrier, by definition (4.24), we have $|\Phi'[h]| \leq \sqrt{\vartheta}\|h\|_{\Phi''}$. Substituting these in (4.35), using the inequalities in (4.34) and the fact that $\vartheta \geq 1$, we have:

$$g'''(0) \leq \left(2\gamma^3 + \frac{3\gamma^2}{\sqrt{\xi - 1}} + \frac{2\gamma}{\xi - 1} + \frac{2\xi}{(\xi - 1)^{3/2}}\right)(g''(0))^{3/2}, \tag{4.36}$$

where γ is defined in (4.34).

For the second part of the lemma for the conjugate function, see the proof of Theorem 2.4.1 in [52]. $\square$

4.2.3 Legendre-Fenchel conjugate of s.c. barriers

If f is a ϑ-s.c.b., then f_* is a s.c. function, but it is **not** necessarily a s.c. barrier. As we will see later, this fact can restrict us to a great extent. Q_* is either the entire $\mathbb{E}^*$ if Q is bounded, or the open cone

$$\operatorname{rec}_*(Q) := \{s \in \mathbb{E}^* : \langle s, h \rangle < 0, \forall h \in \operatorname{rec}(Q)\}, \tag{4.37}$$

where $\operatorname{rec}(Q)$ is the *recession cone* of Q defined as

$$\operatorname{rec}(Q) := \{h \in \mathbb{E} : x + th \in Q, \quad \forall x \in Q, \ \forall t \geq 0\}. \tag{4.38}$$

Losing s.c. barrier properties makes the analysis more challenging. However, f_* has some useful properties beyond those of a regular s.c. function, such as the following theorem that we use several times in this thesis.

Theorem 4.2.1 (Theorem 2.4.2 of [52])**.** *Assume that f is a ϑ-s.c. barrier on D and let f_* be the Legendre-Fenchel conjugate of f with domain D_*. Then,*

1. *for every point $y \in D_*$, we have*

$$f_*''(y)[y, y] \leq \vartheta; \tag{4.39}$$

2. *the support function of D*

$$\mathcal{S}(y) := \sup\{\langle y, z \rangle : z \in D\},$$

satisfies the inequality

$$\mathcal{S}(y) - \frac{\vartheta}{k} \leq \langle f_*'(ky), y \rangle \leq \mathcal{S}(y), \quad y \in D_*, \quad \forall k > 0. \tag{4.40}$$

Let us prove the following lemma by using Theorem 4.2.1:

Lemma 4.2.4. *Assume that Φ is a ϑ-s.c. barrier on $\operatorname{int} D$ and let Φ_* be the Legendre-Fenchel conjugate of Φ with domain $\operatorname{int} D_*$. For every $y \in \operatorname{int} D_*$ and every $\xi > 1$, the function $f(\gamma) := -\xi\vartheta \ln(\gamma) - \Phi_*(\gamma y)$ is a $\frac{(\xi-1)^3}{(\xi+1)^2}$-s.c. function.*

Proof. We have

$$\begin{aligned} f''(\gamma) = \frac{\xi\vartheta}{\gamma^2} - \Phi_*''(\gamma y)[y,y] &= \frac{\xi\vartheta}{\gamma^2} - \frac{1}{\gamma^2}\Phi_*''(\gamma y)[\gamma y,\gamma y] \\ &\geq \frac{\xi\vartheta}{\gamma^2} - \frac{\vartheta}{\gamma^2} \quad \text{using (4.39)} \\ &> 0, \quad \text{(because } \xi > 1\text{).} \end{aligned} \tag{4.41}$$

Hence, $f(\gamma)$ is a convex function. For the third derivative, we have

$$f'''(\gamma) = \frac{-2\xi\vartheta}{\gamma^3} - \Phi_*'''(\gamma y)[y,y,y] = \frac{-2\xi\vartheta}{\gamma^3} - \frac{1}{\gamma^3}\Phi_*'''(\gamma y)[\gamma y,\gamma y,\gamma y]. \tag{4.42}$$

Hence,

$$\begin{aligned} |f'''(\gamma)| &\leq \left|\frac{-2\xi\vartheta}{\gamma^3}\right| + \left|\frac{1}{\gamma^3}\Phi_*'''(\gamma y)[\gamma y,\gamma y,\gamma y]\right| \\ &\leq \frac{2\xi\vartheta}{\gamma^3} + \frac{2}{\gamma^3}\left[\Phi_*''(\gamma y)[\gamma y,\gamma y]\right]^{3/2}, \quad \Phi_* \text{ is 1-s.c.} \\ &\leq 2(\xi+1)\frac{\vartheta^{3/2}}{\gamma^3}, \quad \text{using (4.39) and } \vartheta \geq 1, \\ &\leq \frac{2(\xi+1)}{(\xi-1)^{3/2}}(f''(\gamma))^{3/2}, \quad \text{using (4.41).} \end{aligned} \tag{4.43}$$

□

Chapter 5

Domain-Driven setup and central path

We gave an informal definition of Domain-Driven setup in Chapter 2, which is a convex optimization problem of the form (2.1). We mentioned that D is presented as the domain of a s.c. barrier that we defined in Chapter 4. In this chapter, we are ready to give the formal definition. The definition of the Domain-Driven setup in Chapter 2 was for general vector spaces $\mathbb{E}$ and $\mathbb{Y}$. For simplicity, we give the formal definition here for $\mathbb{E} = \mathbb{R}^n$ and $\mathbb{Y} = \mathbb{R}^m$. This also works better for the implementation that we prefer to have one variable vector in $\mathbb{R}^n$. Therefore, we have to find a proper basis for our vector space and also consider practical purposes. For example, the set of symmetric matrices $\mathbb{S}^n$ is isomorphic to $\mathbb{R}^{n(n+1)/2}$, but for implementation purposes we represent a matrix in $\mathbb{S}^n$ by a vector in $\mathbb{R}^{n^2}$; more on this in Chapter 8. Let us define the Domain-Driven setup.

Definition 5.0.1. *An optimization problem is said to be in* Domain-Driven *setup if it can be written as*

$$\inf_x \{\langle c, x\rangle : Ax \in D\}, \tag{5.1}$$

where $x \mapsto Ax : \mathbb{R}^n \to \mathbb{R}^m$ is a linear embedding ($\ker A = \{0\}$)*, $c \in \mathbb{R}^n$ is given, and $D \subset \mathbb{R}^m$ is a convex set given as the closure of the domain of a ϑ-s.c. barrier $\Phi(\cdot)$. Moreover, the Legendre-Fenchel conjugate of $\Phi(\cdot)$, denoted by $\Phi_*(\cdot)$, can also be calculated efficiently.*

The domain of $\Phi_*(\cdot)$ is the interior of a cone D_* defined as (see (4.37)):

$$D_* = \{y : \langle y, h\rangle \le 0, \quad \forall h \in \mathrm{rec}(D)\}, \tag{5.2}$$

where rec(D) is the recession cone of D defined in (4.38).

Remark 5.0.1. *Note that $\Phi_*(\cdot)$ is a 1-s.c. function, but is not necessarily a s.c. barrier. Losing s.c. barrier properties makes the analysis more challenging. However, $\Phi_*(\cdot)$ has some useful properties beyond those of a regular s.c. function, such as Theorem 4.2.1. As we discussed in Chapter 4, this is not an issue for the conic optimization setup. There, both primal and dual cones are equipped with logarithmically homogenous (LH) s.c. barriers, since the LF conjugate of a LH s.c. barrier is also a LH s.c. barrier.*

Remark 5.0.2. *Nemirovskii and Tunçel [46] designed several feasible-start algorithms for problems in the Domain-Driven setup. They called their approach* cone-free. *Their path-following and potential-reduction algorithms achieve the current best theoretical complexity bound for the case of strict primal and dual feasibility.*

Remark 5.0.3. *The assumption that the LF conjugate of $\Phi(\cdot)$ can be evaluated efficiently can be restricting. Even when the s.c. barrier is available, an efficient way to evaluate its LF conjugate may not be available. The good news is that for all the examples we gave in Chapter 2 and we revisit in this chapter, we show an explicit formula for the LF conjugate or an efficient numerical way to calculate it with high accuracy. There are also numerical methods to evaluate the LF conjugate of a general convex function. Lucet et al. have been doing research on this topic for over a decade [31] . They have created the Computational Convex Analysis numerical library that computes the LF conjugate for a convex function, as well as other operators arising in convex analysis.*

Every convex set "attains" a self-concordant barrier, and as we discussed in Chapter 2, even though a computationally efficient s.c. barrier is not necessarily available for a given convex set, for many interesting convex sets we know how to construct an efficient s.c. barrier. We presented several examples in Chapter 2.

Section 5 of Nesterov and Nemirovski's book [52] is how to construct s.c. barriers, where they present barrier calculus and s.c. barriers for different convex sets. There are many papers that focus on creating s.c. barriers for specific convex sets. An interesting convex set, for example, is the p-cone defined as the epigraph of p-norm

$$K_p := \{x \in \mathbb{R}^n, t \in \mathbb{R} : \ \|x\|_p \le t\}.$$

Constructing s.c. barriers and optimization over K_p have been considered in the literature [11, 79, 51]. However, no simple s.c. barrier with "low" parameter is known for this cone for a general $p \in [1, +\infty)$. Note that for $p = 2$, this cone becomes the second-order cone and we have a 2-s.c. barrier for it.

As another example of constructing s.c. barriers, let f be a symmetric s.c. barrier function and define $F : \mathbb{S}^n \to \mathbb{R}$ ($\mathbb{S}^n$ is the space of $n \times n$ real symmetric matrices) as $F := f \circ \lambda$, where λ is the eigenvalue mapping $\lambda(X) = (\lambda_1(X), \dots, \lambda_n(X))^\top$, where $\lambda_1(X) \geq \cdots \geq \lambda_n(X)$ are the eigenvalues of $X \in \mathbb{S}^n$. Then, Tunçel conjectured that f is a s.c. barrier if and only if F is a s.c. barrier. This conjecture was proved for the special cases that f is separable and also for the case of $n = 2$ [59, 60].

In this chapter, we review examples from Chapter 2 more rigorously and present the corresponding LF conjugates. In Chapter 8, we show how these functions/set constraints can be implemented in software. As we also explained in Chapter 2, what we list here are types of convex sets, and the list can be expanded. We can construct a problem in the Domain-Driven setup by binding an arbitrary number of sets from each type with direct sum. For example, we show that LP, SOCP, or SDP problems of arbitrary size are covered in the Domain-Driven setup. This implies that the Domain-Driven setup contains optimization over symmetric cones with a finite but arbitrary number of constraints from each type. To see this, consider the problem

$$\begin{array}{rc} \min & \langle c, x \rangle \\ s.t. & A_i x \in D_i, \quad i \in \{1, \dots, \ell\}. \end{array} \tag{5.3}$$

Assume that $\Phi_1, \dots, \Phi_\ell$ are s.c. barriers for the sets $D_1, \dots, D_\ell$. Then, property SCB-1-(b) in Subsection 4.2.2 shows that $\Phi := \Phi_1 + \cdots + \Phi_\ell$ is a s.c. barrier for D, where $D := D_1 \oplus \cdots \oplus D_\ell$. Also Lemma 4.2.1 shows that $\Phi_* := \Phi_{1*} + \cdots + \Phi_{\ell *}$ is the LF conjugate of Φ and we have $D_* := D_{1*} \oplus \cdots \oplus D_{\ell *}$.

Remark 5.0.4. *Domain-Driven setup is compatible with the "inequality" form of optimization over symmetric cones. When we talk about implementation of LP, SCOP, and SDP in Chapter 8, we consider them in the inequality form, which can be seen as the dual of the equality form that is acceptable in many popular codes such as SeDuMi and SDPT3. This is an advantage for many problems that are given in the inequality form.*

In this chapter, we give the LF conjugate for the s.c. barriers given in Chapter 2. Many times we need to scale or translate the argument of the s.c. barrier and the following elementary lemma shows how these simple transformations change the LF conjugate.

Lemma 5.0.1. *[6] Assume that $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function and f_* is its Legendre-Fenchel conjugate. Then, translation and scaling of the argument change f_* as in Table 5.1.*

Table 5.1: Transformed LF conjugates.

Transformation	Update in the LF conjugate
$g(x) := f(x+b), \quad x \in \mathbb{R}^n$	$g_*(y) = -\langle y, b\rangle + f_*(y)$
$g(x) := f(\gamma x), \quad \gamma \in \mathbb{R} \setminus \{0\}$	$g_*(y) = f_*(y/\gamma)$

5.1 Restudying the examples from Chapter 2

5.1.1 LP, SOCP, and SDP

LF conjugates for the functions in Table 2.1 are very popular and can be found in most papers on interior-point methods for symmetric cones. Table 5.1.1 is the same table with an extra column for the LF conjugate functions.

Table 5.2: LP, SOCP, and SDP constraints, the corresponding s.c. barriers, and their LF conjugates.

	Constraint	Φ	Φ_*
LP	$z \leq \beta, \quad z, \beta \in \mathbb{R},$	$-\ln(\beta - z)$	$-1 + y\beta - \ln(y)$
SOCP	$\|z\| \leq t, \quad z \in \mathbb{R}^m, \quad t \in \mathbb{R},$	$-\ln(t^2 - z^\top z)$	$-2 + \ln(4) - \ln(\eta^2 - w^\top w)$
SDP	$Z \preceq B, \quad Z, B \in \mathbb{S}^m$	$-\ln(\det(B - Z))$	$-m + \langle Y, B\rangle - \ln(\det(Y))$

Note how the barriers and their LF conjugates are the same ignoring some affine terms, which is related to the fact that nonnegative orthant, second-order cones, and positive-semidefinite cones are symmetric cones.

5.1.2 Direct sum of 2-dimensional sets

We discussed that all the constraints of the form (2.3), where $\alpha_i \geq 0$ and $f_i(x)$, $i \in \{1, \dots, \ell\}$, can be any function from Table 2.2, is in the Domain-Driven setup. This can be shown by the following relation that can easily be verified.

$$\begin{aligned} & \left\{ x : \sum_{i=1}^{\ell} \alpha_i f_i(a_i^\top x + \beta_i) + g^\top x + \gamma \le 0 \right\} \\ = \ & \left\{ x : \exists u \in \mathbb{R}^\ell \text{ such that } \sum_{i=1}^{\ell} \alpha_i u_i + g^\top x + \gamma \le 0, \quad f_i(a_i^\top x + \beta_i) \le u_i, \ \forall i \right\}. \end{aligned} \tag{5.4}$$

The first constraint in the RHS is a linear constraint, which we showed is in the Domain-Driven setup, and constraints $f_i(a_i^\top x + \beta_i) \le u_i$ are also in the Domain-Driven setup by using Table 2.2. In this thesis, we have considered the implementation of the first five rows of Table 2.2, i.e., we accept constraints of the form (2.3), where $f_i(x)$'s are chosen from the first 5 rows of Table 2.2. For the first three rows, the corresponding LF conjugates are shown in Table 5.1.2.

Table 5.3: LF conjugates for the first three s.c. barriers in Table 2.2.

	$\Phi(z,t)$	$\Phi_*(y,\eta)$
1	$-\ln(t+\ln(z)) - \ln(z)$	$-1 + (-\eta+1)\left[-1 + \ln \frac{-(-\eta+1)}{y}\right] - \ln(-\eta)$
2	$-\ln(\ln(t) - z) - \ln(t)$	$-1 + (y+1)\left[-1 + \ln \frac{-(y+1)}{\eta}\right] - \ln(y)$
3	$-\ln(t - z\ln(z)) - \ln(z)$	$-\ln(-\eta) + \theta\left(1 + \frac{y}{\eta} - \ln(-\eta)\right) - \frac{y}{\eta} + \frac{1}{\theta\left(1+\frac{y}{\eta} - \ln(-\eta)\right)} - 3$

Finding the LF conjugates for the first two functions can be handled with easy calculus. In the third row, $\theta(r)$, defined in [46], is the unique solution of

$$\frac{1}{\theta} - \ln(\theta) = r.$$

It is easy to check by implicit differentiation that

$$\theta'(r) = -\frac{\theta^2(r)}{\theta(r)+1}, \quad \theta''(r) = \frac{\theta^2(r) + 2\theta(r)}{[\theta(r)+1]^2}\theta'(r).$$

We can calculate $\theta(r)$ with accuracy 10^{-15} in few steps with the following Newton iterations:

$$\theta_k = \frac{\theta_{k-1}^2}{\theta_{k-1}+1}\left[1 + \frac{2}{\theta_{k-1}} - \ln(\theta_{k-1}) - r\right], \quad \theta_0 = \begin{cases} \exp(-r), & r \le 1 \\ \frac{1}{r - \ln(r - \ln(r))}, & r > 1 \end{cases}.$$

The LF conjugates of rows 4 and 5 need more work that we will do in Chapter 8. We discussed in Chapter 2 that geometric programming is a special case when we use only the second row.

5.1.3 Generalized epigraph of a matrix norm

For $Z \in \mathbb{R}^{m\times m}$ and $U \in \mathbb{R}^{m\times n}$, we gave an m-s.c. barrier for the set $\{(Z,U) : Z-UU^\top \succeq 0\}$ in (2.9). Note that the parameter of the barrier for the SDP reformulation is $m+n$ that is much larger in the $m \ll n$ scenario. The LF conjugate of the s.c. barrier is derived in [46] as:

$$\Phi_*(Y,V) = -m - \frac{1}{4}\mathrm{Tr}(V^\top Y^{-1} V) - \ln(\det(-Y)), \tag{5.5}$$

where $Y \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{m\times n}$. We have a discussion in Chapter 8 on efficiently calculating the gradient and Hessian for $\Phi(Z,U)$ and $\Phi_*(Y,V)$.

5.2 Duality gap for Domain-Driven setup

An elegant notion of duality gap that can be easily calculated is a strength in conic optimization. Duality gap is a good measure of distance to optimality for a pair of primal and dual feasible points. In this section, we show a nice notion of duality for the Domain-Driven setup and define a duality gap. The definition and properties of our duality gap rely very much on the fact that Φ_* is the LF conjugate of a s.c. barrier, which gives it more properties compared to a general s.c. function; specifically Theorem 4.2.1. Recalling the definition of support function of D, that is defined on $\mathrm{int}(D_*)$ as

$$\mathcal{S}(y) := \sup\{\langle y, z\rangle : z \in D\}, \tag{5.6}$$

we define the duality gap as:

Definition 5.2.1. *For every point $x \in \mathbb{R}^n$ such that $Ax \in D$ and every point $y \in D_*$ such that $A^\top y = -c$,* the duality gap *is defined as:*

$$\langle c, x\rangle + \mathcal{S}(y). \tag{5.7}$$

Remark 5.2.1. *Duality gap must be easily computable and support function is not generally easy to calculate. However, Theorem 4.2.1 shows that we can estimate the support function with a desired accuracy using the fact that Φ_* is the LF conjugate of a s.c. barrier.*

The following lemma shows that duality gap is well-defined and zero duality gap is a guarantee for optimality:

Lemma 5.2.1. *For every point $x \in \mathbb{R}^n$ such that $Ax \in D$ and every point $y \in D_*$ such that $A^\top y = -c$, we have*

$$\langle c, x\rangle + \mathcal{S}(y) \geq 0. \tag{5.8}$$

Moreover, if the equality holds above for a pair $(\hat{x}, \hat{y})$ with $A\hat{x} \in D$ and $\hat{y} \in D_*$, $A^\top \hat{y} = -c$, then $\hat{x}$ is an optimal solution of (5.1).

Proof. Let x and y be as above. Then,

$$\langle c, x\rangle \underbrace{=}_{A^\top y=-c} -\langle A^\top y, x\rangle = -\langle y, Ax\rangle \underbrace{\geq}_{Ax\in D,\ y\in D_*} -S(y).$$

Thus, $\langle c, x\rangle + \mathcal{S}(y) \geq 0$, as desired. If equality holds for $(\hat{x}, \hat{y})$, then for every $Ax \in D$, we have

$$\langle c, \hat{x}\rangle \underbrace{=}_{\text{(5.8) holds with equality}} -\mathcal{S}(\hat{y}) \underbrace{\leq}_{(5.6)} -\langle \hat{y}, Ax\rangle = \langle -A^\top \hat{y}, x\rangle \underbrace{=}_{A^\top \hat{y}=-c} \langle c, x\rangle.$$

Therefore, $\hat{x}$ is an optimal solution for (5.1). □

Remark 5.2.2. *As shown in the proof of Lemma 5.2.1, if x and y satisfy the properties in Definition 5.2.1, then*

$$\langle c, x\rangle + \mathcal{S}(y) = \mathcal{S}(y) - \langle y, Ax\rangle. \tag{5.9}$$

Corollary 5.2.1. *Assume that there exist a sequence $\{z^k\} \in \operatorname{int} D$ such that $z^k \to A\hat{x} \in D$, and a sequence $\{y^k\} \in \operatorname{int} D_*$ such that $y^k \to \hat{y} \in D_*$ and $A^\top \hat{y} = -c$. If*

$$\lim_k \left(\langle c, x^k\rangle + \langle y^k, \Phi_*'(ky^k)\rangle\right) = 0,$$

then $\hat{x}$ is an optimal solution of (5.1).

Proof. We use Theorem 4.2.1 to approximate the support function and then the result holds by using Lemma 5.2.1. □

5.3 Primal-dual infeasible-start central path

Our algorithms are *infeasible-start*, which means we do not require a feasible point from the user to start the algorithm. To introduce our infeasible-start central path, we bring a feasible start central path, called cone-free in [46], which is defined by the set of solutions to:

$$\begin{array}{ll} (a) & Ax \in \text{int}D, \\ (b) & A^\top y = -\tau c, \quad y \in \text{int}D_*, \\ (c) & y = \Phi'(Ax), \end{array} \tag{5.10}$$

with $\tau > 0$ being the parameter of the path. Assume that there exists $\hat{x}$ such that $A\hat{x} \in \text{int}(D)$ and $\hat{y} \in \text{int}(D_*)$ such that $A^\top \hat{y} = -c$, and let $(x(\tau), y(\tau))$ be the solution of (5.10) for $\tau > 0$. We can also define $x(\tau)$ as

$$x(\tau) := \text{argmin}_x \left\{\tau\langle c, x\rangle + \Phi(Ax)\right\}. \tag{5.11}$$

This is a typical way for solving problem (5.1) by removing the hard constraint $Ax \in D$ and adding a penalty to the objective function. The system (5.10) can be seen as the optimality conditions for problem (5.11), for a fixed value of τ. Then, it is proved in [46] that $x(\tau)$ tends to a solution of (5.1) when $\tau \to +\infty$. By using Theorem 4.2.1, we can also show this by proving that $(x(\tau), y(\tau)/\tau)$ tends to satisfy the properties of Lemma 5.2.1 with zero duality gap when $\tau \to +\infty$.

Let us see how to modify (5.10) for an infeasible-start algorithm. What we have to start the algorithm with is a point $z^0 \in \text{int}D$ and $y^0 := \Phi'(z^0) \in \text{int}D_*$. We modify the primal and dual feasibility parts of (5.10) as follows:

$$\begin{array}{lll} (a) & Ax + \frac{1}{\tau}z^0 \in \text{int}D, & \tau > 0, \\ (b) & A^\top y = A^\top y^0 - (\tau - 1)c, & y \in \text{int}D_*, \end{array} \tag{5.12}$$

where $\tau^0 = 1$, $x^0 = 0$, is feasible for this system, and when $\tau \to +\infty$, we get a pair of primal-dual feasible points in limit. Let us give a name to the set of points that satisfy (5.12):

$$Q_{DD} := \left\{(x, \tau, y) : Ax + \frac{1}{\tau}z^0 \in \text{int}D,\ \tau > 0,\ \ A^\top y - A^\top y^0 = -(\tau - 1)c,\ y \in \text{int}D_*\right\}. \tag{5.13}$$

Remark 5.3.1. *A natural extension for* (5.10) *would be*

$$\begin{array}{lll} (a) & Ax + \frac{1}{\tau}z^0 \in \text{int}D, & \tau > 0, \\ (b) & A^\top y = A^\top y^0 - (\tau - 1)c, & y \in \text{int}D_*, \\ (c) & y = \Phi'\left(Ax + \frac{1}{\tau}z^0\right). & \end{array} \tag{5.14}$$

As we will briefly see in Appendix B, this system defines a primal-dual central path and we can design an algorithm based on it. Even though this central path is simpler compared to the one we will see later in this chapter, so far we could not achieve the best theoretical iteration complexity bounds using it.

Our goal is to design infeasible-start primal-dual algorithms as robust as the best ones for the conic setup, which as far as we know, are the homogenous self-dual embedding type algorithms proposed in [56]. Consider the primal-dual conic setup in (3.8). The duality gap for (3.8) is

$$\langle c, z\rangle + \langle y, b\rangle = \langle s, z\rangle. \tag{5.15}$$

The duality gap for the modified problem in [56] has the following two crucial properties when the parameter of the path tends to $+\infty$: (1) it tends to zero if the problem is solvable, (2) it tends to $+\infty$ if primal or dual is infeasible. We will show how to enforce such a property for the Domain-Driven setup. We are interested in the duality gap between x and $\hat{y} = \frac{y}{\tau}$, where $(x, \tau, y) \in Q_{DD}$. Note that we have two expressions for the Domain-Driven duality gap in (5.9) that we can associate with the conic duality gap in (5.15).

The natural extension of the nonlinear equation (5.10)-(c) to the infeasible-start case would be $y = \Phi'(Ax + \frac{1}{\tau}z^0)$. Assuming this equality, by using Theorem 4.2.1 for $k = \frac{1}{\tau}$ and $\hat{y}$, we obtain

$$\begin{aligned} &\mathcal{S}(\hat{y}) - \frac{\vartheta}{\tau} \le \langle \hat{y}, \Phi'_*(y)\rangle = \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle \le \mathcal{S}(\hat{y}), \\ \Rightarrow \quad & 0 \le \mathcal{S}(\hat{y}) - \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle \le \frac{\vartheta}{\tau}, \end{aligned} \tag{5.16}$$

where we used the fact that $\Phi'_*(\Phi'(z)) = z$ by the properties of Legendre-Fenchel conjugate in (4.5). Note that, by Remark 5.2.2, $\mathcal{S}(\hat{y}) - \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle$ is the duality gap for the modified problem. (5.16) implies that if $\tau \to +\infty$, the duality gap tends to zero. However, (5.12) does not have a solution for every $\tau > 0$ if at least one of $Ax \in D$ or $A^\top y = -c,\ y \in D_*$, is infeasible, and (5.16) shows that the duality gap stays bounded in this case. To resolve this problem, we extend (5.10)-(c) to $y = \frac{\mu}{\tau}\Phi'(Ax + \frac{1}{\tau}z^0)$, where μ plays the role of the parameter for the central path. Then, (5.16) changes to

$$0 \le \mathcal{S}(\hat{y}) - \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle \le \frac{\mu\vartheta}{\tau^2}. \tag{5.17}$$

If we add $\langle c, x\rangle + \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle$ to both sides, we get

$$\langle c, x\rangle + \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle \le \mathcal{S}(\hat{y}) + \langle c, x\rangle \le \frac{\mu\vartheta}{\tau^2} + \langle c, x\rangle + \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle. \tag{5.18}$$

We want to increase μ to $+\infty$ such that

- if τ increases with the same rate as μ, the duality gap tends to zero,
- if τ is bounded, the duality gap tends to $-\infty$.

Enforcing μ to satisfy $-\frac{\xi\vartheta\mu}{\tau^2} = \langle c, x\rangle + \langle \hat{y}, Ax + \frac{1}{\tau}z^0\rangle$ for an absolute constant $\xi > 1$ reduces (5.18) to

$$-\frac{\xi\vartheta\mu}{\tau^2} \leq \mathcal{S}(\hat{y}) + \langle c, x\rangle \leq -\frac{(\xi-1)\vartheta\mu}{\tau^2}, \tag{5.19}$$

which satisfies our goals. Now we are ready to define our central path with parameter μ. Let us fix $\xi > 1$ and define:

$$z^0 := \text{any vector in } \operatorname{int}(D), \quad y^0 := \Phi'(z^0), \quad y_\tau^0 := -\langle y^0, z^0\rangle - \xi\vartheta. \tag{5.20}$$

The following theorem defines our central path.

Theorem 5.3.1. *Consider the convex set $D \subset \mathbb{R}^m$ equipped with a ϑ-s.c. barrier Φ and let Φ_* be its Legendre-Fenchel conjugate with domain* $\operatorname{int}D_*$. *Then, for any set of starting points defined in* (5.20), *the system*

$$\begin{aligned}
&(a) \quad Ax + \frac{1}{\tau}z^0 \in \operatorname{int}D, \quad \tau > 0,\\
&(b) \quad A^\top y - A^\top y^0 = -(\tau-1)c, \quad y \in \operatorname{int}D_*,\\
&(c) \quad y = \frac{\mu}{\tau}\Phi'\left(Ax + \frac{1}{\tau}z^0\right),\\
&(d) \quad \langle c, x\rangle + \frac{1}{\tau}\langle y, Ax + \frac{1}{\tau}z^0\rangle = -\frac{\vartheta\xi\mu}{\tau^2} + \frac{-y_\tau^0}{\tau},
\end{aligned} \tag{5.21}$$

has a unique solution $(x(\mu), \tau(\mu), y(\mu))$ for every $\mu > 0$.

We denote the solution set of (5.21) for $\mu > 0$ by the *domain-driven primal-dual central path*. y_τ^0 in (5.21)-(d) is to make this equation hold for the initial points and $\mu^0 = 1$. In view of the definition of the central path, for all the points $(x, \tau, y) \in Q_{DD}$, we define

$$\begin{aligned}
\mu(x,\tau,y) &:= \frac{\tau}{\xi\vartheta}[-y_\tau^0 - \tau\langle c, x\rangle - \langle y, Ax + \frac{1}{\tau}z^0\rangle],\\
&= -\frac{1}{\xi\vartheta}\left[\langle y, z^0\rangle + \tau(y_\tau^0 + \langle y, Ax\rangle) + \tau^2\langle c, x\rangle\right]\\
&= -\frac{1}{\xi\vartheta}\left[\langle y, z^0\rangle + \tau(y_\tau^0 + \langle c, x\rangle + \langle y^0, Ax\rangle)\right], \quad \text{using (5.21)-(b)}.
\end{aligned} \tag{5.22}$$

Note that the formula in the second line is a quadratic in terms of τ. However, when we use dual feasibility, we get the third formula that is linear in τ. In other words, dual feasibility removes one of the roots. Assume that both the primal and dual are strictly feasible and we choose $z^0 = 0$ and y^0 such that $A^\top y^0 = -c$. Then, the last equation of (5.22) reduces to $\mu = \tau$ and (5.21) reduces to the cone-free setup in (5.10). We can roughly describe how the last definition of μ in (5.22) works: assume that τ is finite and we can prove that $\langle y, z^0\rangle + \langle y^0, Ax\rangle$ is bounded, then when μ tends to $+\infty$, either $\langle c, x\rangle$ tends to $-\infty$, which makes (5.1) unbounded, or $\langle y, z^0\rangle$ tends to $-\infty$, which we will see translates to (5.1) being infeasible.

Proof of Theorem 5.3.1. Consider the function $\Phi(\frac{z}{\tau}) - \xi\vartheta \ln(\tau)$ that we proved in Lemma 4.2.3 is a s.c. function. The Legendre-Fenchel conjugate of this function, as a function of (y, y_τ), is calculated from the following formula:

$$\max_{\gamma>0} \left[\Phi_*(\gamma y) + y_\tau \gamma + \xi\vartheta \ln \gamma\right]. \tag{5.23}$$

The gradient of $\Phi(\frac{z}{\tau}) - \xi\vartheta \ln(\tau)$ is

$$\begin{bmatrix} \frac{1}{\tau}\Phi'(\frac{z}{\tau}) \\ -\frac{1}{\tau^2}\langle \Phi'(\frac{z}{\tau}), z\rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix}. \tag{5.24}$$

By substituting (5.21)-(c) in (5.21)-(d) and reordering the terms, we can show that for every $\mu > 0$, the solution set of (5.21) corresponds to the solution set of the following system

$$\begin{aligned} \begin{bmatrix} y \\ y_\tau \end{bmatrix} &= \mu \begin{bmatrix} \frac{1}{\tau}\Phi'(\frac{z}{\tau}) \\ -\frac{1}{\tau^2}\langle \Phi'(\frac{z}{\tau}), z\rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix}, \\ z &= \tau A x + z^0, \\ A^\top y &= A^\top y^0 - (\tau - 1)c, \\ y_\tau &= y_\tau^0 + \tau\langle c, x\rangle. \end{aligned} \tag{5.25}$$

Consider the following function:

$$\begin{aligned} &\Phi\left(\frac{z}{\tau}\right) - \xi\vartheta\ln(\tau) + \max_{\gamma>0}\left[\Phi_*(\gamma y) + y_\tau\gamma + \xi\vartheta\ln\gamma\right] - \frac{1}{\mu}\left(\langle y, z\rangle + \tau y_\tau\right) \qquad (5.26) \\ &\geq \Phi\left(\frac{z}{\tau}\right) - \xi\vartheta\ln\tau + \Phi_*(\gamma y) + y_\tau\gamma + \xi\vartheta\ln\gamma - \frac{1}{\mu}\left(\langle y, z\rangle + \tau y_\tau\right), \qquad \forall \gamma > 0, \end{aligned}$$

where the last inequality trivially holds because of the max function. Let us substitute $\gamma := \frac{\tau}{\mu}$, then by using the Fenchel-Young inequality

$$\Phi\left(\frac{z}{\tau}\right) + \Phi_*\left(\frac{\tau y}{\mu}\right) \geq \langle \frac{\tau y}{\mu}, \frac{z}{\tau} \rangle = \frac{1}{\mu}\langle y, z \rangle,$$

we can continue (5.26) as

$$\begin{aligned} &\geq \frac{1}{\mu}\langle y, z\rangle + \frac{1}{\mu}\tau y_\tau - \xi\vartheta \ln \mu - \frac{1}{\mu}\left(\langle y, z\rangle + \tau y_\tau\right) \\ &= -\xi\vartheta \ln \mu. \end{aligned} \tag{5.27}$$

Hence, the function is bounded from below for every μ.

Further, we can write ($\bar{c}$ is any vector such that $A^\top \bar{c} = c$)

$$\begin{aligned} \langle y, z\rangle + \tau y_\tau &= \langle y, \tau A x + z^0\rangle + \tau(y_\tau^0 + \tau\langle c, x\rangle) \\ &= \langle A^\top y^0 - (\tau - 1)c, \tau x\rangle + \langle y, z^0\rangle + \tau y_\tau^0 + \tau^2\langle c, x\rangle, \\ &= \langle y^0 + \bar{c}, z - z^0\rangle + \tau y_\tau^0 + \langle y, z^0\rangle, \end{aligned} \tag{5.28}$$

which is linear in (z, τ, y, y_τ). For a fixed μ, consider the optimization problem

$$\begin{array}{ll} \min & \Phi\left(\frac{z}{\tau}\right) - \xi\vartheta\ln(\tau) + \max_{\gamma>0}\left[\Phi_*(\gamma y) + y_\tau\gamma + \xi\vartheta\ln\gamma\right] - \frac{1}{\mu}\left(\langle y, z\rangle + \tau y_\tau\right) \\ s.t. & Fz = Fz^0, \\ & A^\top y = A^\top y^0 - (\tau - 1)c, \\ & y_\tau = y_\tau^0 + \langle \bar{c}, z - z^0\rangle. \end{array} \tag{5.29}$$

By the above discussion, (5.29) is minimizing a s.c. function that is bounded from below and so attains its unique minimizer $(\bar{z}, \bar{\tau}, \bar{y}, \bar{y}_\tau)$ by property SC-4 of s.c. functions in Subsection 4.2.1. Using $F\bar{z} = Fz^0$, there exists a unique $\bar{x}$ such that $\bar{z} = \bar{\tau}A\bar{x} + z^0$. We can easily verify that $(\bar{x}, \bar{\tau}, \bar{y})$ is the unique solution of (5.21) for the fixed μ as it satisfies all the optimality conditions. □

The algorithm we design in the next section is a *predictor-corrector path-following* algorithm. We define a measure of proximity to the central path and specify a small $\mathcal{N}_1$ and a big $\mathcal{N}_2$ neighborhood. Then, at every iteration, we

- start from a point in $\mathcal{N}_1$ and take a predictor step to increase μ as much as possible to μ^+ while staying in $\mathcal{N}_2$.

- perform a small number of corrector steps to return to a point in $\mathcal{N}_1$ with parameter μ^+.

Let us finish this section by showing that if we are close to the central path, a strong inequality holds for the duality gap:

Lemma 5.3.1. *Let $(x, \tau, y) \in Q_{DD}$ be a point close to the central path, specifically assume that there exists a constant $\kappa > 0$ such that*

$$\left\| Ax + \frac{1}{\tau} z^0 - \Phi_*' \left(\frac{\tau}{\mu} y \right) \right\|_{[\Phi_*''(\frac{\tau}{\mu} y)]^{-1}} \leq \kappa. \tag{5.30}$$

Then, for the duality gap between x and $\hat{y} := \frac{y}{\tau}$ we have

$$\frac{-y_\tau^0}{\tau} - \frac{\xi \mu \vartheta + \mu \kappa \sqrt{\vartheta}}{\tau^2} \leq \langle c, x \rangle + \mathcal{S}(\hat{y}) \leq \frac{-y_\tau^0}{\tau} - \frac{(\xi - 1)\mu \vartheta - \mu \kappa \sqrt{\vartheta}}{\tau^2}. \tag{5.31}$$

Proof. By applying Theorem 4.2.1 to $k := \frac{\tau^2}{\mu}$ and $\hat{y}$ we get

$$\langle \hat{y}, \Phi_*' \left(\frac{\tau y}{\mu} \right) \rangle \leq \mathcal{S}(\hat{y}) \leq \langle \hat{y}, \Phi_*' \left(\frac{\tau y}{\mu} \right) \rangle + \frac{\mu \vartheta}{\tau^2}. \tag{5.32}$$

Note that by adding and subtracting a term we have

$$\langle \hat{y}, \Phi_*' \left(\frac{\tau y}{\mu} \right) \rangle = \langle \hat{y}, \Phi_*' \left(\frac{\tau y}{\mu} \right) - \left(Ax + \frac{1}{\tau} z^0 \right) \rangle + \langle \hat{y}, Ax + \frac{1}{\tau} z^0 \rangle. \tag{5.33}$$

Now by using the fact that $\|\frac{\tau y}{\mu}\|_{\Phi_*''(\frac{\tau}{\mu} y)} \leq \sqrt{\vartheta}$ (property (4.39)), the assumption of the lemma, and using Cauchy-Schwarz inequality we get

$$-\frac{\mu \kappa}{\tau^2} \sqrt{\vartheta} \leq \langle \hat{y}, \Phi_*' \left(\frac{\tau y}{\mu} \right) - \left(Ax + \frac{1}{\tau} z^0 \right) \rangle \leq \frac{\mu \kappa}{\tau^2} \sqrt{\vartheta}. \tag{5.34}$$

By substituting (5.34) in (5.33) and the result in (5.32) we get

$$\langle \hat{y}, Ax + \frac{1}{\tau} z^0 \rangle - \frac{\mu \kappa}{\tau^2} \sqrt{\vartheta} \leq \mathcal{S}(\hat{y}) \leq \langle \hat{y}, Ax + \frac{1}{\tau} z^0 \rangle + \frac{\mu \kappa}{\tau^2} \sqrt{\vartheta} + \frac{\mu \vartheta}{\tau^2}. \tag{5.35}$$

The result of the lemma follows if we substitute for $\langle \hat{y}, Ax + \frac{1}{\tau} z^0 \rangle$ from the definition of μ in (5.22). $\square$

5.4 Informal outcomes of following the path

In Chapter 6 we present an algorithm and analyze it to show how efficiently we can follow the path to $\mu \to +\infty$, and in Chapter 7 we rigorously discuss how to interpret the outcome of the algorithm. In this section, we give an informal discussion to see what we get when $\mu \to +\infty$. Let us for simplicity assume that our point is on the central path. At each iteration of the algorithm, we have (x, τ, y), which uniquely defines μ, and

(i) $Ax + \frac{1}{\tau} z^0 \in \text{int} D$, $A^\top \hat{y} = \frac{A^\top y^0 + c}{\tau} - c$, and the duality gap satisfies (Lemma 5.3.1):

$$\frac{-y^0_\tau}{\tau} - \frac{\xi \mu \vartheta}{\tau^2} \leq \langle c, x \rangle + \mathcal{S}(\hat{y}) \leq \frac{-y^0_\tau}{\tau} - \frac{(\xi - 1)\mu\vartheta}{\tau^2}.$$

(ii) Using this inequality, at least one of the following happens:

(1) For x we have

$$\langle c, x \rangle \leq \frac{-y^0_\tau}{\tau} - \frac{(\xi - 1)\mu\vartheta}{2\tau^2}.$$

(2) For y we have

$$\mathcal{S}(\hat{y}) \leq -\frac{(\xi - 1)\mu\vartheta}{2\tau^2}.$$

As can be seen, parameter $\frac{\mu\vartheta}{\tau^2}$ plays a crucial role in our interpretation. Table 5.4 summarizes how we interpret the outcome of the algorithm, with $\epsilon > 0$ being a small parameter and $L > 0$ being a large parameter set by the user.

Table 5.4: How to interpret the outcome of the algorithm with given ϵ and L.

$\frac{\mu\vartheta}{\tau^2} < \epsilon$	$\epsilon \leq \frac{\mu\vartheta}{\tau^2} \leq L$	$L < \frac{\mu\vartheta}{\tau^2}$
Solvable	Ill-conditioned: for example, both primal and dual (approximately) feasible with nonzero duality gap	Infeasible or unbounded

Here is how we interpret the above data:

- If $\frac{\mu\vartheta}{\tau^2}$ tends to zero when μ becomes large (which implies a large τ and approximately zero duality gap), we get an ϵ-approximate solution for an ϵ-perturbation of the problem. We will see that this happens when our problem is solvable.

- If τ is finite, (1) is a certificate of unboundedness and (2) is a certificate of infeasibility for an ϵ-perturbation of the problem. This means that we have either a certificate of infeasibility or a certificate of unboundedness for an approximate problem. We categorize two types of infeasibility (strongly or strictly) and two types of unboundedness (strongly or strictly) conditions and analyze how our algorithms can distinguish them. We show that in the strict cases, we can get certificates for the exact problem.

- If τ gets relatively large in the sense that $\frac{\mu\vartheta}{\tau^2}$ is not close to zero and not large enough to put us in the above two categories, then both primal and dual are approximately feasible, but there is a duality gap. Note that at least one of (ii)-(1) or (ii)-(2) still happens in any case.

Chapter 6

Algorithms and complexity analysis

In the previous chapter, we defined our infeasible-start primal-dual central path, parameterized with μ, and discussed the scheme of a predictor-corrector algorithm. In this chapter, we design a predictor-corrector path-following primal-dual algorithm that efficiently follows the path to $\mu = +\infty$. As we mentioned in Chapter 3, our algorithms achieve the current best iteration complexity bounds, which are new for the "modifying the RHS" type formulations. There are different challenges in following our path efficiently, which are mostly related to the fact that τ is attached to x and y in a nonlinear way in our setup (5.21).

6.1 Algorithms

To define neighborhoods of the central path, we need a notion of proximity. For a point $(x, \tau, y) \in Q_{DD}$, defined in (5.13), we define a proximity measure as

$$\begin{aligned} \Omega(x,\tau,y) &:= \Phi\left(Ax + \frac{1}{\tau}z^0\right) + \Phi_*\left(\frac{\tau y}{\mu}\right) - \frac{\tau}{\mu}\langle y, Ax + \frac{1}{\tau}z^0\rangle, \\ \mu &= \mu(x,\tau,y), \quad \text{as defined in (5.22)}. \end{aligned} \tag{6.1}$$

Throughout Chapters 6 and 7, we may drop the arguments of Φ and Φ_* (and also their gradients and Hessians) for simplicity, i.e., $\Phi := \Phi\left(Ax + \frac{1}{\tau}z^0\right)$ and $\Phi_* := \Phi_*\left(\frac{\tau}{\mu}y\right)$.

Remark 6.1.1. *We mentioned in Remark 5.0.2 that Nemirovskii and Tunçel [46] designed feasible-start algorithms for the Domain-Driven setup. The proximity measure they use in their paper is*

$$\Phi(Ax) + \Phi_*(y) - \langle y, Ax\rangle. \tag{6.2}$$

Even though this proximity measure and (6.1) *have similar structures (indeed for $z^0 = 0$ and $\tau = \mu$, we recover* (6.2)*), τ and μ bring nonlinearity into the arguments of Φ and Φ_* in* (6.1)*.*

Theorem 6.1.1. *For every $(x, \tau, y) \in Q_{DD}$ we have $\Omega(x, \tau, y) \geq 0$. Moreover, $\Omega(x, \tau, y) = 0$ with $\mu = \mu(x, \tau, y)$ iff (x, τ, y) is on the central path for parameter $\mu(x, \tau, y)$.*

Proof. Both parts of the theorem are implied by Fenchel-Young inequality (Theorem 4.1.1) and the definition of the central path. □

Now we can express the predictor-corrector algorithm. Note that we choose different step sizes for x and for (τ, y), i.e., for the search direction (d_x, d_τ, d_y), the updates are

$$x^+ := x + \alpha_1 d_x, \quad \tau^+ := \tau + \alpha_2 d_\tau, \quad y^+ := y + \alpha_2 d_y. \tag{6.3}$$

Framework for Predictor-Corrector Algorithms

Input: $A \in \mathbb{R}^{m \times n}$, $c \in \mathbb{R}^n$, neighborhood parameters $\delta_1, \delta_2 \in (0, 1)$ such that $\delta_1 < \delta_2$, desired tolerance $\epsilon \in (0, 1)$. Access to gradient and Hessian oracles for a ϑ-s.c. barrier $\Phi(\cdot)$ such that $\vartheta \geq 1$ and $\text{dom}\Phi = \text{int}D$, and its LF conjugate $\Phi_*(\cdot)$. $z^0 \in \text{int}D$.

1. Set $k := 0$, $y^0 := \Phi'(z^0)$, $x^0 := 0$, and $\tau^0 := 1$.

   Repeat until the stopping criteria are met:

2. If $\Omega(x^k, \tau^k, y^k) > \delta_1$, calculate the corrector search direction (d_x, d_τ, d_y), choose $\alpha_1 > 0$ and $\alpha_2 > 0$, and apply the update in (6.3) to get $(x^{k+1}, \tau^{k+1}, y^{k+1})$, such that $\Omega(x^{k+1}, \tau^{k+1}, y^{k+1})$ is smaller than $\Omega(x^k, \tau^k, y^k)$ by a "large enough" amount, while $\mu(x^{k+1}, \tau^{k+1}, y^{k+1}) = \mu(x^k, \tau^k, y^k)$.

3. If $\Omega(x^k, \tau^k, y^k) \leq \delta_1$, calculate the predictor search direction (d_x, d_τ, d_y), choose $\alpha_1 > 0$ and $\alpha_2 > 0$, and apply the update in (6.3) to get $(x^{k+1}, \tau^{k+1}, y^{k+1})$, such that $\mu(x^{k+1}, \tau^{k+1}, y^{k+1})$ is larger than $\mu(x^k, \tau^k, y^k)$ by a "large enough" amount, while

   $$\Omega(x^{k+1}, \tau^{k+1}, y^{k+1}) \leq \delta_2.$$

4. Set $k := k + 1$ and continue.

The missing parts in the algorithm are the search direction and the step lengths. In order to get the best complexity bound, the predictor step must increase μ by a large enough amount at every iteration, and by a constant number of corrector steps we must be able to come back to the small neighborhood. We first show how to calculate these steps, and in the next section, we analyze the performance of them.

6.1.1 Predictor step

Our goal in the predictor step is to increase μ as much as possible while staying in a fixed neighborhood of the central path. An efficient search direction in the predictor step must (i) increase μ by a large rate and, (ii) let us take a long enough step. A power of s.c. functions is giving us an elegant tool to control our movement in the feasible region. The Dikin ellipsoid property SC-2 in Subsection 4.2.1 is the most fundamental property of s.c. functions that implies we can move all the way to the boundary of Dikin ellipsoid and stay feasible. The challenge in our Domain-Driven setup is the nonlinear way that τ is combined with x and y, for example in the proximity measure (6.1). What typically appears in a primal-dual proximity measure in the literature is that the summation of the s.c. barrier and its LF conjugate is composed with an affine function of the variables, which makes the algorithm and analysis much easier. The positive definite matrix that defines the Dikin ellipsoid for our algorithm has a special form that controls the nonlinear displacements in the arguments of Φ and Φ_* in the proximity measure.

Let us define $\bar{H}(x,\tau)$ as ($u := Ax + \frac{1}{\tau}z^0$)

$$\bar{H}(x,\tau) := \begin{bmatrix} \underbrace{\frac{1}{\tau^2}\Phi''(u)}_{=:H} & \underbrace{\frac{-1}{\tau^2}\Phi''(u)\,u - \frac{1}{\tau^2}\Phi'(u)}_{=:h} \\ \left[\frac{-1}{\tau^2}\Phi''(u)\,u - \frac{1}{\tau^2}\Phi'(u)\right]^\top & \underbrace{\frac{2}{\tau^2}\langle\Phi'(u),u\rangle + \frac{1}{\tau^2}\langle u, \Phi''(u)\,u\rangle + \frac{\xi\vartheta}{\tau^2}}_{=:\zeta} \end{bmatrix}. \tag{6.4}$$

One can easily verify that

$$\bar{H}(x,\tau)[(d_z,d_\tau),(d_z,d_\tau)] = \left\|\frac{d_z}{\tau} - \frac{d_\tau}{\tau}u\right\|^2_{\Phi''(u)} + \frac{2d_\tau}{\tau}\left[\frac{d_z}{\tau} - \frac{d_\tau}{\tau}u\right]^\top \Phi'(u) + \xi\frac{d_\tau^2}{\tau^2}\vartheta. \tag{6.5}$$

By using the definition of s.c barriers (4.24) for the second term in the RHS of (6.5), we have

$$\left[\left\|\frac{d_z}{\tau} - \frac{d_\tau}{\tau}u\right\|_{\Phi''(u)} - \frac{d_\tau}{\tau}\sqrt{\vartheta}\right]^2 + (\xi-1)\frac{d_\tau^2}{\tau^2}\vartheta \leq \bar{H}(x,\tau)[(d_z,d_\tau),(d_z,d_\tau)], \tag{6.6}$$

which shows that $\bar{H}(x,\tau)$ is a positive definite matrix for every $\xi > 1$.

Remark 6.1.2. *If we replace u with $\frac{z}{\tau}$ in (6.4), we get the Hessian for the function $\Phi\left(\frac{z}{\tau}\right) + \xi\vartheta\ln(\tau)$ that we proved in Lemma 4.2.3 is a s.c. function. This is another proof that this matrix is positive definite.*

$\bar{H}(x,\tau)$ is positive definite for $\xi > 1$ and so invertible. By substitution, one can directly verify that

$$\left(\begin{bmatrix} H & h \\ h^\top & \zeta \end{bmatrix}\right)^{-1} = \begin{bmatrix} H^{-1} + \eta H^{-1}hh^\top H^{-1} & -\eta H^{-1}h \\ -\eta h^\top H^{-1} & \eta \end{bmatrix}, \quad \eta := \frac{1}{\zeta - h^\top H^{-1}h}. \tag{6.7}$$

Let F be a matrix whose rows give a basis for null$(A^\top)$ and let $\bar{c}$ be any vector such that $A^\top \bar{c} = c$. We define a block matrix U that comes up frequently in our discussion and contains the linear transformations we need

$$U = \begin{bmatrix} A & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\bar{c} & -F^\top \\ c^\top & 0 & 0 \end{bmatrix}. \tag{6.8}$$

Then, for the current point (x,τ,y) and a positive definite matrix $\hat{H}(x,\tau,y)$, that we elaborate more on later, consider the solution of the system:

$$\underbrace{U^\top \begin{bmatrix} \bar{H}(x,\tau) & 0 \\ 0 & \left[\hat{H}(x,\tau,y)\right]^{-1} \end{bmatrix} U}_{\mathcal{H}(\bar{H},\hat{H})} \begin{bmatrix} \bar{d}_x \\ d_\tau \\ d_v \end{bmatrix} = \frac{r^0}{\mu^2}, \qquad r^0 := \begin{bmatrix} -A^\top y^0 - c \\ -y^0_\tau + \langle \bar{c}, z^0 \rangle \\ Fz^0 \end{bmatrix}, \tag{6.9}$$

and define our search direction as:

$$d_x := \bar{d}_x - d_\tau x, \qquad d_y := -d_\tau \bar{c} - F^\top d_v. \tag{6.10}$$

Then for the predictor step we have:

Predictor step: Choose $\alpha_1, \alpha_2 \in \mathbb{R}_+$ and update $x^+ := x + \alpha_1 d_x$, $\tau^+ := \tau + \alpha_2 d_\tau$, and $y^+ := y + \alpha_2 d_y$ such that $\Omega(x^+,\tau^+,y^+) \le \delta_2$ and $\mu(x^+,\tau^+,y^+)$ is maximized. If we set $\alpha_1 := \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$, we can choose α_2 large enough to get the desired complexity bound.

We have different choices for $\hat{H}(x,\tau,y)$ to get our desired properties (such as a low complexity bound). We express a sufficient condition and discuss two choices that satisfy the condition. Let us first see the following critical lemma that we prove in the next section. This lemma shows that when we are close to the central path, $\bar{H}(x,\tau)$ is under control.

Lemma 6.1.1. *Assume that $(x,\tau,y) \in Q_{DD}$ with parameter μ such that $\Omega(x,\tau,y) \le \epsilon < 1$. Then, there exists an absolute constant $\bar{\epsilon} < 1$ depending on ϵ such that*

$$(1-\bar{\epsilon})^2 \bar{H}(x(\mu),\tau(\mu)) \preceq \bar{H}(x,\tau) \preceq \frac{1}{(1-\bar{\epsilon})^2} \bar{H}(x(\mu),\tau(\mu)). \tag{6.11}$$

We will see that to achieve enough increase in μ at every predictor step, it is enough for $\hat{H}(x,\tau,y)$ to satisfy

$$(1-\bar{\epsilon})^2 \left[\bar{H}(x(\mu),\tau(\mu))\right]^{-1} \preceq \mu^2 \left[\hat{H}(x,\tau,y)\right]^{-1} \preceq \frac{1}{(1-\bar{\epsilon})^2} \left[\bar{H}(x(\mu),\tau(\mu))\right]^{-1}. \tag{6.12}$$

Choice 1: In view of (6.11), one obvious choice for $\hat{H}(x,\tau,y)$ is

$$\hat{H}(x,\tau,y) := \mu^2 \bar{H}(x,\tau).$$

Choice 2: By this choice, we get the search direction for the primal-dual conic setup given in [48, 56] that we review in Appendix A. Comparison of (6.9) with (A.17) and (A.18) shows a close connection. In remark 6.1.2, we explained that, by a change of variables, matrix $\bar{H}(x,\tau)$ is the Hessian of $\Phi\left(\frac{z}{\tau}\right) + \xi\vartheta\ln(\tau)$. The LF conjugate of this function is implicitly given in (5.23). By the connection between the Hessians of a function and its LF conjugate in (4.5), we have a choice for $\hat{H}(x,\tau,y)$ to satisfy (6.12). Note that the definition of the objective function in (5.29) is implicit because of the one-dimensional maximization problem; how can we calculate the gradient and Hessian? Here, we show that solving the maximization problem gives the gradient and Hessian explicitly. Using first order optimality conditions, the conjugate function in (5.23) can also be written as

$$\Phi_*(\bar{\tau}y) + y_\tau\bar{\tau} + \xi\vartheta\ln\bar{\tau}, \tag{6.13}$$

where $\bar{\tau}$ is the solution of the system $y_\tau + \langle y, \Phi'_*(\bar{\tau}y)\rangle + \frac{\xi\vartheta}{\bar{\tau}} = 0$. There exists $\bar{z} \in D$ such that (y, y_τ) is the image of $(\bar{z},\bar{\tau})$ under the gradient of $\Phi(\frac{z}{\tau}) - \xi\vartheta\ln(\tau)$, which means $y = \frac{1}{\bar{\tau}}\Phi'(\frac{\bar{z}}{\bar{\tau}})$. By (4.5), the Hessian of (6.13) at (y, y_τ) is the inverse of $\bar{H}$. An explicit formula for the inverse of $\bar{H}$ is given in (6.7). Let us evaluate the components by using $y = \frac{1}{\bar{\tau}}\Phi'(\frac{\bar{z}}{\bar{\tau}})$. Note that H and h are functions of $(\bar{z},\bar{\tau})$.

$$\begin{aligned}
H^{-1} &= \bar{\tau}^2 \left[\Phi''\left(\frac{\bar{z}}{\bar{\tau}}\right)\right]^{-1} = \underbrace{\bar{\tau}^2\Phi''_*(\bar{\tau}y)}_{=:G}, \\
H^{-1}h &= -\frac{\bar{z}}{\bar{\tau}} - \left[\Phi''\left(\frac{\bar{z}}{\bar{\tau}}\right)\right]^{-1}\Phi'\left(\frac{\bar{z}}{\bar{\tau}}\right) = \underbrace{-\Phi'_*(\bar{\tau}y) - \bar{\tau}\Phi''_*(\bar{\tau}y)y}_{=:h_*}, \\
\zeta - h^\top H^{-1}h &= \underbrace{\frac{\xi\vartheta}{\bar{\tau}^2} - \langle y, \Phi''_*(\bar{\tau}y)y\rangle}_{=:1/\eta_*}.
\end{aligned} \tag{6.14}$$

Hence, at every point (y, y_τ), if we solve the one variable equation $y_\tau + \langle y, \Phi'_*(\bar{\tau}y)\rangle + \frac{\xi\vartheta}{\bar{\tau}} = 0$ for $\bar{\tau}$ (equivalently solve the univariate maximization problem (5.23)), we also explicitly get the Hessian (and similarly the gradient) of the conjugate function that we can use as $\hat{H}(x,\tau,y)$

$$\left[\hat{H}(x,\tau,y)\right]^{-1} := \begin{bmatrix} G+\eta_* h_* h_*^\top & -\eta_* h_* \\ -\eta_* h_*^\top & \eta_* \end{bmatrix}. \tag{6.15}$$

For the points close enough to the central path, (6.15) satisfies condition (6.12). This results from the arguments in [48, 56], or the fact that $\Phi\left(\frac{z}{\tau}\right) + \xi\vartheta\ln(\tau)$ is a s.c. function, using Lemma 6.1.1, and the properties of LF conjugates. Calculating $\bar{\tau}$ can be done very efficiently, because we are actually minimizing a s.c. function in view of Lemma 4.2.4. $\square$

Before showing the corrector step, let us justify our predictor step. If we choose $\alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$, then by using the third line of (5.22) for μ, we have

$$\begin{aligned}
& \mu(x^+,\tau^+,y^+) - \mu(x,\tau,y) \\
= \; & \frac{-1}{\xi\vartheta}\left[\alpha_2\langle d_y, z^0\rangle + \alpha_2 d_\tau y_\tau^0 + \langle c + A^\top y^0, \alpha_2 d_\tau x + (\tau+\alpha_2 d_\tau)\alpha_1 d_x\rangle\right] \\
= \; & \frac{-\alpha_2}{\xi\vartheta}\left(\langle d_y, z^0\rangle + d_\tau y_\tau^0 + \langle c + A^\top y^0, d_\tau x + d_x\rangle\right), \quad \text{substituting } \alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau} \\
= \; & \frac{-\alpha_2}{\xi\vartheta}\left(\langle d_y, z^0\rangle + d_\tau y_\tau^0 + \langle c + A^\top y^0, \bar{d}_x\rangle\right), \quad \text{using (6.10)} \\
= \; & \frac{\alpha_2}{\xi\vartheta}\left(\langle d_v, F z^0\rangle + d_\tau(\langle \bar{c}, z^0\rangle - y_\tau^0) - \langle c + A^\top y^0, \bar{d}_x\rangle\right), \quad \text{substituting } d_y = -d_\tau\bar{c} - F^\top d_v \\
= \; & \frac{\alpha_2}{\xi\vartheta}[\bar{d}_x^\top \quad d_\tau \quad d_v^\top] r^0, \quad \text{for } r^0 \text{ defined (6.9).}
\end{aligned} \tag{6.16}$$

Hence, we are maximizing a linear function over the set of feasible points. If we define $d^\top := [\bar{d}_x^\top \quad d_\tau \quad d_v^\top]$, we will see that the Dikin ellipsoid type constraint $d^\top U^\top \mathcal{H}(\bar{H},\hat{H})Ud \leq 1$ guarantees to keep the updated points feasible. The search direction in (6.9) is, up to some scaling, the solution of the following optimization problem

$$\begin{array}{cc} \max & \langle d, r^0\rangle \\ s.t. & d^\top U^\top \mathcal{H}(\bar{H},\hat{H})Ud \leq 1, \end{array} \tag{6.17}$$

which can be seen as minimizing the linear function in a trust region.

6.1.2 Corrector step

After doing a predictor step to increase μ, we need to perform corrector steps to come back into the small neighborhood. Note that our proximity measure $\Omega(x,\tau,y)$ is not a convex function and to decrease it we use a quasi-Newton like step. In most of the literature on this topic, for example papers [48, 56, 46], the corrector step is simply minimizing a s.c. function that can be done efficiently by taking damped Newton steps (Subsection 4.2.1, SC-3). However, as we mentioned before, because of the nonlinearity in our formulations, the proximity measure is not a s.c. function and we cannot use damped Newton steps. However, functions Φ and Φ_* are 1-s.c. functions and we can use their strong properties. Using (4.17) for $\Phi + \Phi_*$, we have

$$
\begin{aligned}
& \Omega(x^+, \tau^+, y^+) - \Omega(x, \tau, y) \\
\leq\ & \langle \Phi', Ax^+ + \tfrac{1}{\tau^+} z^0 - Ax - \tfrac{1}{\tau} z^0 \rangle + \langle \tfrac{\tau^+ y^+}{\mu^+} - \tfrac{\tau y}{\mu}, \Phi'_* \rangle \\
& - \langle \tfrac{\tau^+ y^+}{\mu^+}, Ax^+ + \tfrac{1}{\tau^+} z^0 \rangle + \langle \tfrac{\tau y}{\mu}, Ax + \tfrac{1}{\tau} z^0 \rangle \\
& + \rho \left(- \left\| Ax^+ + \tfrac{1}{\tau^+} z^0 - Ax - \tfrac{1}{\tau} z^0 \right\|_{\Phi''} - \left\| \tfrac{\tau^+ y^+}{\mu^+} - \tfrac{\tau y}{\mu} \right\|_{\Phi''_*} \right),
\end{aligned}
\tag{6.18}
$$

where $\Phi' := \Phi'\left(Ax + \frac{1}{\tau} z^0\right)$ and $\Phi'_* := \Phi'_*\left(\frac{\tau}{\mu} y\right)$. For the predictor step, we focus on the RHS of (6.18) and try to find a search direction to make this term negative with a large enough absolute value. The issue is that this term has nonlinear factors, which come from the cross products of (d_x, d_τ, d_y). We first define the corrector step and then explain our choice. Consider the solution of the system

$$
\begin{aligned}
U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U \begin{bmatrix} \bar{d}_x \\ d_\tau \\ d_v \end{bmatrix} &= -(U^\top \psi^c + \beta r^0), \\
\beta &:= -\frac{\langle r^0, [U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U]^{-1} U^\top \psi^c \rangle}{\langle r^0, [U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U]^{-1} r^0 \rangle},
\end{aligned}
\tag{6.19}
$$

where $\mathcal{H}$ is defined in (6.9) and

$$
\psi^c := \begin{bmatrix} \frac{1}{\tau} \Phi' \\ -\frac{1}{\tau} \langle \Phi', Ax + \frac{1}{\tau} z^0 \rangle + \frac{1}{\mu} \langle y, \Phi'_* \rangle + \frac{1}{\mu} (y_\tau^0 + \tau \langle c, x \rangle) \\ \frac{\tau}{\mu} \Phi'_* \\ \frac{\tau}{\mu} \end{bmatrix}.
\tag{6.20}
$$

We define the corrector search direction as

$$
d_x := \bar{d}_x - d_\tau x, \qquad d_y := -d_\tau \bar{c} - F^\top d_v.
\tag{6.21}
$$

As can be seen, we have

$$
d_x \in \ \left\{ \bar{d}_x, x \right\}, \qquad d_y \in \ \left\{ \bar{c}, F^\top d_v \right\}.
\tag{6.22}
$$

Then, our corrector step is as follows:

Corrector step: Choose $\alpha_1, \alpha_2 \in \mathbb{R}_+$ and update $x^+ := x + \alpha_1 d_x$, $\tau^+ := \tau + \alpha_2 d_\tau$, and $y^+ := y + \alpha_2 d_y$ such that $\mu(x^+, \tau^+, y^+) = \mu(x, \tau, y)$ and $\Omega(x^+, \tau^+, y^+)$ is minimized. If we set $\alpha_1 := \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$, we can choose α_2 large enough to get the desired complexity bound.

Remark 6.1.3. *If we choose $\alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$, then (6.16) holds and for the search directions (6.19) we automatically have $\mu(x^+, \tau^+, y^+) = \mu(x, \tau, y)$, since β is chosen in a way to have $[\bar{d}_x^\top \quad d_\tau \quad d_v^\top] r^0 = 0$.*

Let us elaborate more on the corrector search direction.

Lemma 6.1.2. *For the corrector step, if we choose $\alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$, inequality (6.18) is simplified to*

$$\begin{array}{rl} & \Omega(x^+, \tau^+, y^+) - \Omega(x, \tau, y) \\ \leq & \alpha_2 \left[\bar{d}_x^\top \quad d_\tau \quad d_v^\top\right] U^\top \psi^c \\ & -\frac{\alpha_2^2 d_\tau}{\tau(\tau+\alpha_2 d_\tau)} \langle \Phi', A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0\right) \rangle + \frac{\alpha_2^2 d_\tau}{\mu} \left(\langle d_y, \Phi'_* \rangle + \langle c, \bar{d}_x \rangle\right) \\ & +\rho \left(-\frac{\alpha_2}{\tau+\alpha_2 d_\tau} \left\| A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0\right) \right\|_{\Phi''} - \left\| \frac{\alpha_2 d_\tau y + \alpha_2 \tau d_y + \alpha_2^2 d_\tau d_y}{\mu} \right\|_{\Phi''_*} \right), \end{array} \tag{6.23}$$

where ψ^c is defined in (6.20).

Proof. Let us expand (6.18) when $\alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$. Note that in the corrector step, we always have $\mu^+ = \mu$. First we have

$$\begin{array}{rcl} Ax^+ + \frac{1}{\tau^+} z^0 - Ax - \frac{1}{\tau} z^0 & = & \alpha_1 A d_x - \frac{\alpha_2 d_\tau}{\tau(\tau+\alpha_2 d_\tau)} z^0 \\ & = & \frac{\alpha_2}{\tau+\alpha_2 d_\tau} \left[A d_x - \frac{d_\tau}{\tau} z^0 \right], \quad \text{using } \alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau} \\ & = & \frac{\alpha_2}{\tau+\alpha_2 d_\tau} \left[A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0 \right) \right], \quad \text{using } d_x = \bar{d}_x - d_\tau x. \end{array} \tag{6.24}$$

As an intermediate step, we have

$$\begin{array}{rcl} \tau^+ x^+ & = & (\tau+\alpha_2 d_\tau)(x + \alpha_1 d_x) \\ & = & \tau x + \alpha_2 d_\tau x + \alpha_1 (\tau + \alpha_2 d_\tau) d_x \\ & = & \tau x + \alpha_2 d_\tau x + \alpha_2 d_x, \quad \text{using } \alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau} \\ & = & \tau x + \alpha_2 \bar{d}_x, \quad \text{using } d_x = \bar{d}_x - d_\tau x. \end{array} \tag{6.25}$$

Then, by using the first line of (5.22) and by substituting $\mu^+ = \mu$, we have

$$\begin{aligned}
& -\langle \frac{\tau^+ y^+}{\mu^+}, Ax^+ + \frac{1}{\tau^+} z^0 \rangle + \langle \frac{\tau y}{\mu}, Ax + \frac{1}{\tau} z^0 \rangle \\
= & -\frac{\tau^+}{\mu^+}[-y_\tau^0 - \tau^+ \langle c, x^+\rangle] + \frac{\tau}{\mu}[-y_\tau^0 - \tau\langle c, x\rangle] \\
= & \frac{\alpha_2}{\mu}[d_\tau y_\tau^0 + \tau\langle c, \bar{d}_x\rangle + d_\tau\langle c, \tau x + \alpha_2 \bar{d}_x\rangle], \quad \text{substituting (6.25) and } \mu^+ = \mu.
\end{aligned}$$

For the last piece, we have

$$\frac{\tau^+ y^+}{\mu^+} - \frac{\tau y}{\mu} = \frac{\alpha_2 d_\tau y + \alpha_2 \tau d_y + \alpha_2^2 d_\tau d_y}{\mu}.$$

We can verify by direct substitution that

$$U^\top \psi^c = \begin{bmatrix} \frac{1}{\tau} A^\top \Phi' + \frac{1}{\mu}\tau c \\ -\frac{1}{\tau}\langle \Phi', Ax + \frac{1}{\tau} z^0\rangle + \frac{1}{\mu}\langle y - \tau \bar{c}, \Phi'_*\rangle + \frac{1}{\mu}(y_\tau^0 + \tau\langle c, x\rangle) \\ -\frac{\tau}{\mu} F \Phi'_* \end{bmatrix}. \tag{6.26}$$

If we also use the equality $F^\top d_v = -d_y - d_\tau \bar{c}$, then we have

$$\begin{aligned}
\begin{bmatrix} \bar{d}_x^\top & d_\tau & d_v^\top \end{bmatrix} U^\top \psi^c = & \frac{1}{\tau}\langle \Phi', A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0\right)\rangle + \frac{\tau}{\mu}\langle c, \bar{d}_x\rangle \\
& + \frac{d_\tau}{\mu}\langle y, \Phi'_*\rangle + \frac{\tau}{\mu}\langle d_y, \Phi'_*\rangle + \frac{d_\tau}{\mu}(y_\tau^0 + \tau\langle c, x\rangle).
\end{aligned} \tag{6.27}$$

If we substitute the above equalities in the RHS of inequality (6.18), we get (6.23). □

In the corrector step, we want $\mu^+ = \mu$, hence, by using (6.16), we must have $\langle d, r^0\rangle = 0$ for $d^\top := \begin{bmatrix} \bar{d}_x^\top & d_\tau & d_v^\top \end{bmatrix}$. The corrector step direction in (6.19) is, up to some scaling, the optimal solution of

$$\begin{array}{cc}
\min & \langle d, U^\top \psi^c\rangle \\
s.t. & \langle d, r^0\rangle = 0 \\
& d^\top U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U d \leq 1.
\end{array} \tag{6.28}$$

Note that on the central path we have $U^\top \psi^c + \beta r^0 = 0$ for β defined in (6.19). We prove that if $\Omega(x, \tau, y) \geq \delta_1$, then the optimal objective value of (6.28) is larger than an absolute constant depending on δ_1. We also use the fact that it appears with coefficient α_2 in (6.23) whereas the cross product terms have coefficient α_2^2. Using these facts, we prove that α_2 can be chosen large enough such that $\Omega(x^+, \tau^+, y^+) - \Omega(x, \tau, y)$ is smaller than a negative absolute constant at every corrector step.

Remark 6.1.4. *What we prove for the corrector step is enough for our complexity results. However, corrector steps in most of the other papers in this context (such as [48, 56, 46]) have the stronger property of quadratic convergence for the points close enough to the central path. What we prove in this thesis is similar to property* (4.21)*-(b) of the damped Newton step for s.c. functions, whereas quadratic convergence can be achieved by property* (4.21)*-(c). When the corrector step is simply minimizing a s.c. function, as it is in most of the papers in the literature, quadratic convergence comes for free. As we have explained and will show in the next section, our corrector step is more complicated. We should be able to prove quadratic (or at least super-linear) convergence for our corrector steps, which is included in our future works.*

6.2 Analysis of the algorithms

In this section, we analyze the predictor and corrector steps we defined in the previous section. This analysis lets us modify the framework for primal-dual algorithms in Section 6.1 to achieve the current best iteration complexity bounds. This modification and the main theorem about it come in Section 6.3. Consider the definition of our proximity measure in (6.1). The following lemma shows how to bound the proximity measure based on the local distance of the current primal and dual points:

Lemma 6.2.1. *(a) Assume that $f(x)$ is an a-s.c. function and $f_*(y)$ is its Legendre-Fenchel conjugate, then for every x and y in the domains of f and f_* we have*

$$\rho\left(r\right) \leq f(x) + f_*(y) - \langle y, x\rangle \leq \rho\left(-r\right), \tag{6.29}$$

where $r := a^{-1/2}\|y - f'(x)\|_{[f''(x)]^{-1}}$.
(b) Moreover, assume that there exist $\hat{x}$ and $\hat{y}$ such that $\hat{y} = f'(\hat{x})$ and $\langle x - \hat{x}, y - \hat{y}\rangle = 0$. Then,

$$\rho(r) + \rho(s) \leq f(x) + f_*(y) - \langle y, x\rangle \leq \rho(-r) + \rho(-s), \tag{6.30}$$

where $r := a^{-1/2}\|x - \hat{x}\|_{f''(\hat{x})}$ and $s := a^{-1/2}\|y - \hat{y}\|_{f_*''(\hat{y})}$.

Proof. (a) By writing property (4.17) of the s.c. functions for f_* at two points y and $f'(x)$, we have

$$f_*(y) \leq f_*(f'(x)) + \langle f_*'(f'(x)), y - f'(x)\rangle + \rho(-a^{-1/2}\|y - f'(x)\|_{f_*''(f'(x))}).$$

To get the RHS inequality in (6.29), we use (4.5) to substitute $f'_*(f'(x)) = x$, $f_*(f'(x)) + f(x) = \langle f'(x), x\rangle$, and $f''_*(f'(x)) = [f''(x)]^{-1}$. The LHS inequality can be similarly proved by using the second inequality in property (4.17).
(b) By writing property (4.17) for f at x and $\hat{x}$ and for f_* at y and $\hat{y}$, and adding them together we get the result. $\square$

Corollary 6.2.1. *For the proximity measure defined in* (6.1)*, we have*

$$\rho\left(\left\|\frac{\tau y}{\mu} - \Phi'(u)\right\|_{[\Phi''(u)]^{-1}}\right) \leq \Omega(x,\tau,y) \leq \rho\left(-\left\|\frac{\tau y}{\mu} - \Phi'(u)\right\|_{[\Phi''(u)]^{-1}}\right), \tag{6.31}$$

where as before $u = Ax + \frac{1}{\tau}z^0$.

6.2.1 Predictor step

Let us first show how the predictor step increases μ. For analyzing this, we prove a nice result about the structure of U. Let us start with the following lemma:

Lemma 6.2.2. *Assume that $\mathcal{H}$ is a positive definite matrix and U is a matrix of proper size with linearly independent columns. Then, for any given vector f of proper size, the solution of $U^\top \mathcal{H} U g = U^\top f$ satisfies*

$$Ug = \mathcal{H}^{-1}f - \mathcal{H}^{-1}{U^\perp}^\top \left(U^\perp \mathcal{H}^{-1}{U^\perp}^\top\right)^{-1} U^\perp \mathcal{H}^{-1} f, \tag{6.32}$$

where $U^\perp$ is a matrix whose rows form a basis for the orthogonal subspace of the column space of U.

Proof. Considering $U^\top \mathcal{H} U g = U^\top f$ and the definition of $U^\perp$, there exists w such that $\mathcal{H}Ug = f + {U^\perp}^\top w$. Multiplying both sides by $\mathcal{H}^{-1}$ gives us $Ug = \mathcal{H}^{-1}f + \mathcal{H}^{-1}{U^\perp}^\top w$. We want to calculate w; we multiply both sides of the last equation from the left by $U^\perp$. Note that $U^\perp U = 0$ and $U^\perp \mathcal{H}^{-1}{U^\perp}^\top$ is invertible. If we solve for w and substitute it in $Ug = \mathcal{H}^{-1}f + \mathcal{H}^{-1}{U^\perp}^\top w$, we get (6.32). $\square$

If we multiply both sides of (6.32) from the left by $f^\top$ and substitute $g = (U^\top \mathcal{H} U)^{-1} U^\top f$, we get

$$f^\top U \left(U^\top \mathcal{H} U\right)^{-1} U^\top f = f^\top \mathcal{H}^{-1} f - f^\top \mathcal{H}^{-1}{U^\perp}^\top \left(U^\perp \mathcal{H}^{-1}{U^\perp}^\top\right)^{-1} U^\perp \mathcal{H}^{-1} f. \tag{6.33}$$

We are interested in matrix $U \in \mathbb{R}^{(2m+2)\times(m+1)}$ we defined in (6.8), which has a very special structure. For this U, one option for $U^{\perp}$ is

$$U^{\perp} = \begin{bmatrix} 0 & c & A^{\top} & 0 \\ -\bar{c} & 0 & 0 & 1 \\ -F & 0 & 0 & 0 \end{bmatrix}. \tag{6.34}$$

If we compare U and $U^{\perp}$, we see that the rows of U is a permutation of the columns of $U^{\perp}$. Explicitly

$$U^{\perp} = U^{\top} P, \qquad P := \begin{bmatrix} 0 & I_{m+1} \\ I_{m+1} & 0 \end{bmatrix}. \tag{6.35}$$

Assume that $\mathcal{H}$ and f have the following special forms:

$$\mathcal{H} := \begin{bmatrix} \bar{H} & 0 \\ 0 & \frac{1}{\mu^2}\bar{H}^{-1} \end{bmatrix}, \quad f := \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \tag{6.36}$$

Note that $\mathcal{H}^{-1} = \mu^2 P \mathcal{H} P$. Then, using (6.35), we have

$$f^{\top}\mathcal{H}^{-1}U^{\perp\top}\left(U^{\perp}\mathcal{H}^{-1}U^{\perp\top}\right)^{-1}U^{\perp}\mathcal{H}^{-1}f = \mu^2 \begin{bmatrix} f_2 \\ f_1 \end{bmatrix}^{\top} \mathcal{H}U\left(U^{\top}\mathcal{H}U\right)^{-1}U^{\top}\mathcal{H}\begin{bmatrix} f_2 \\ f_1 \end{bmatrix}.$$

If we further have $f_1 = \mu\bar{H}f_2$ or $f_1 = -\mu\bar{H}f_2$, then (6.33) reduces to

$$f^{\top}U\left(U^{\top}\mathcal{H}U\right)^{-1}U^{\top}f = \frac{1}{2}f^{\top}\mathcal{H}^{-1}f. \tag{6.37}$$

In our setup, we have an explicit formula to calculate $f^{\top}\mathcal{H}^{-1}f$. Let us use (6.37) to derive a result about our setup. We define

$$\psi^p := \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad f_1 := \begin{bmatrix} \frac{1}{\tau}\Phi' \\ -\frac{1}{\tau}\langle \Phi', Ax + \frac{1}{\tau}z^0\rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix}, \quad f_2 := \begin{bmatrix} \frac{\tau}{\mu}\left(Ax + \frac{1}{\tau}z^0\right) \\ \frac{\tau}{\mu} \end{bmatrix}. \tag{6.38}$$

Then we can easily verify that on the central path we have $U^{\top}\psi^p(\mu) = -\frac{1}{\mu}r^0$, where r^0 is defined in (6.9). The matrix $\bar{H}$ we are using in our setup is (6.4), and for that we can verify (note that $u = Ax + \frac{1}{\tau}z^0$)

$$\begin{aligned} &\frac{1}{\mu}\begin{bmatrix} \frac{1}{\tau}\Phi' \\ -\frac{1}{\tau}\langle \Phi', Ax + \frac{1}{\tau}z^0\rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix} = \\ &-\begin{bmatrix} \frac{1}{\tau^2}\Phi''(u) & \frac{-1}{\tau^2}\Phi''(u)\,u - \frac{1}{\tau^2}\Phi'(u) \\ \left[\frac{-1}{\tau^2}\Phi''(u)\,u - \frac{1}{\tau^2}\Phi'(u)\right]^{\top} & \frac{2}{\tau^2}\langle \Phi'(u), u\rangle + \frac{1}{\tau^2}\langle u, \Phi''(u)\,u\rangle + \frac{\xi\vartheta}{\tau^2} \end{bmatrix}\begin{bmatrix} \frac{\tau}{\mu}\left(Ax + \frac{1}{\tau}z^0\right) \\ \frac{\tau}{\mu} \end{bmatrix}. \end{aligned} \tag{6.39}$$

Therefore, $f_1 = -\mu \bar{H} f_2$ and so (6.37) holds for our setup. Now, we prove the following lemma:

Lemma 6.2.3. *Consider $\mathcal{H}$ defined in (6.9) and ψ^p defined in (6.38) for a point $(x, \tau, y) \in Q_{DD}$. Then, we have*

$$\langle U^\top \psi^p, \left[U^\top \mathcal{H}\left(\bar{H}, \mu^2 \bar{H}\right) U\right]^{-1} U^\top \psi^p \rangle = \xi\vartheta. \tag{6.40}$$

Proof. (6.39) confirms that $f_1 = -\mu \bar{H} f_2$, so we have equation (6.37). Hence, we need to show that $(\psi^p)^\top \mathcal{H}^{-1} \psi^p = 2\xi\vartheta$ to get our result. This holds since by direct verification we have

$$-\mu \begin{bmatrix} \frac{1}{\tau}\Phi' \\ -\frac{1}{\tau}\langle \Phi', Ax + \frac{1}{\tau}z^0 \rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix}^\top \begin{bmatrix} \frac{\tau}{\mu}\left(Ax + \frac{1}{\tau}z^0\right) \\ \frac{\tau}{\mu} \end{bmatrix} = \xi\vartheta, \tag{6.41}$$

and $(\psi^p)^\top \mathcal{H}^{-1} \psi^p$, by using (6.39), is exactly the summation of two terms like (6.41). $\square$

Now we are ready to prove the following main proposition about how the predictor step increases μ.

Proposition 6.2.1. *Assume that $(x, \tau, y) \in Q_{DD}$ and conditions (6.11) and (6.12) hold. Let our search direction be the solution of (6.9)-(6.10). If we choose $\alpha_1 = \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$, then, for every $\alpha_2 > 0$ we have*

$$(1-\bar{\epsilon})^2 \alpha_2 \leq \mu(x^+, \tau^+, y^+) - \mu(x, \tau, y) \leq \frac{\alpha_2}{(1-\bar{\epsilon})^2}. \tag{6.42}$$

Proof. By starting from (6.16) for $\mu(x^+, \tau^+, y^+) - \mu(x, \tau, y)$, we can continue

$$\begin{aligned}
& \mu(x^+, \tau^+, y^+) - \mu(x, \tau, y) \\
= \;& \frac{\alpha_2}{\xi\vartheta}[\bar{d}_x^\top \quad d_\tau \quad d_v^\top] r^0 \\
= \;& \frac{\alpha_2}{\xi\vartheta}\frac{1}{\mu^2}\langle r^0, [U^\top \mathcal{H}(\bar{H}, \hat{H}) U]^{-1} r^0 \rangle, \quad \text{using (6.9)} \\
= \;& \frac{\alpha_2}{\xi\vartheta}\langle U^\top \psi^p(\mu), [U^\top \mathcal{H}(\bar{H}, \hat{H}) U]^{-1} U^\top \psi^p(\mu) \rangle, \quad \text{using } U^\top \psi^p(\mu) = -\frac{1}{\mu} r^0.
\end{aligned} \tag{6.43}$$

We get the desired result by using conditions (6.11) and (6.12) and then utilizing Lemma 6.2.3 for the points on the central path. $\square$

Proposition 6.2.1 implies that the amount of increase in μ depends directly on α_2. Therefore, we need to show how large α_2 can be in the predictor step.

By direct substitution in (6.7), one can verify that

$$\begin{bmatrix} f \\ f_\tau \end{bmatrix}^\top \left(\begin{bmatrix} H & h \\ h^\top & \zeta \end{bmatrix} \right)^{-1} \begin{bmatrix} f \\ f_\tau \end{bmatrix} = \langle f, H^{-1} f \rangle + \eta \left(\langle f, H^{-1} h \rangle - f_\tau \right)^2,$$
$$H^{-1} h = -u - [\Phi'']^{-1} \Phi', \qquad \eta = \frac{\tau^2}{\xi \vartheta - \langle \Phi', [\Phi'']^{-1} \Phi' \rangle}. \tag{6.44}$$

The following lemma is critical in our analysis. We explained that matrix $\bar{H}$ in (6.4) and its inverse define the Dikin ellipsoid type properties for our setup. The following lemma lets us break down a local norm defined by $\mathcal{H}$ in (6.9) into the bounds we need for the analysis of our algorithms for the Domain-Driven setup.

Lemma 6.2.4. *Assume that $(x, \tau, y) \in Q_{DD}$ and conditions (6.11) and (6.12) hold. Then, for the solution of (6.9) we have*

$$\begin{aligned} & \left[\left\| \frac{A\bar{d}_x}{\tau} - \frac{d_\tau}{\tau} \left(Ax + \frac{1}{\tau} z^0 \right) \right\|_{\Phi''(u)} - \frac{d_\tau}{\tau} \sqrt{\vartheta} \right]^2 + (\xi - 1) \frac{d_\tau^2}{\tau^2} \vartheta \\ + \quad & \frac{\tau^2}{\mu^2} \langle d_y, [\Phi''(u)]^{-1} d_y \rangle + \frac{\left[\langle \frac{\tau d_y}{\mu}, [\Phi'']^{-1} \Phi' \rangle + \frac{\tau}{\mu} (\langle d_y, Ax + \frac{1}{\tau} z^0 \rangle + \langle c, \bar{d}_x \rangle) \right]^2}{\xi \vartheta - \langle \Phi', [\Phi'']^{-1} \Phi' \rangle} \leq q, \end{aligned} \tag{6.45}$$

where $q := \frac{1}{(1-\bar{\epsilon})^6} \frac{\xi \vartheta}{\mu^2}$.

Proof. Let us define

$$f := U \begin{bmatrix} \bar{d}_x \\ d_\tau \\ d_v \end{bmatrix} = \begin{bmatrix} A\bar{d}_x \\ d_\tau \\ d_y \\ c^\top \bar{d}_x \end{bmatrix}.$$

Using (6.6) and (6.44), we can bound the LHS of (6.45) from above by $f^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) f$, where $\mathcal{H}$ is defined in (6.9). If we substitute f using (6.9) and then use (6.11) and (6.12), we can bound this term as

$$
\begin{aligned}
& f^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) f \\
= \;& \frac{1}{\mu^4} \langle \left[U^\top \mathcal{H}(\bar{H}, \hat{H}) U\right]^{-1} r^0, (U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U) \left[U^\top \mathcal{H}(\bar{H}, \hat{H}) U\right]^{-1} r^0 \rangle \\
\leq \;& \frac{1}{(1-\bar{\epsilon})^4 \mu^4} \langle r^0, \left[U^\top \mathcal{H}(\bar{H}, \hat{H}) U\right]^{-1} r^0 \rangle, \quad \text{using (6.11) and (6.12)} \\
\leq \;& \frac{1}{(1-\bar{\epsilon})^6 \mu^4} \langle r^0, \left[U^\top \mathcal{H}\left(\bar{H}(\mu), \mu^2 \bar{H}(\mu)\right) U\right]^{-1} r^0 \rangle, \quad \text{using (6.11)} \\
= \;& \frac{\langle U^\top \psi^p(\mu), \left[U^\top \mathcal{H}\left(\bar{H}(\mu), \mu^2 \bar{H}(\mu)\right) U\right]^{-1} U^\top \psi^p(\mu) \rangle}{(1-\bar{\epsilon})^6 \mu^2}, \quad \text{using } U^\top \psi^p(\mu) = -\frac{1}{\mu} r^0 \\
= \;& \frac{1}{(1-\bar{\epsilon})^6 \mu^2} \xi \vartheta \quad \text{using Lemma 6.2.3.}
\end{aligned}
\tag{6.46}
$$

□

Let us see what we get if we break down (6.45). First, clearly

$$
(\xi - 1) \frac{d_\tau^2}{\tau^2} \vartheta \leq q \quad \Rightarrow \quad \left(\frac{d_\tau}{\tau}\right)^2 \leq \frac{q}{(\xi - 1)\vartheta} = \frac{\xi}{\xi - 1} \frac{1}{(1-\bar{\epsilon})^6} \frac{1}{\mu^2}. \tag{6.47}
$$

Using (6.45) and (6.47), we get

$$
\begin{aligned}
\frac{1}{\tau} \left\| A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0\right) \right\|_{\Phi''} \;&\leq\; \sqrt{q} + \left|\frac{d_\tau}{\tau}\right| \sqrt{\vartheta} \\
&\leq\; \sqrt{q} + \sqrt{\frac{q}{\xi - 1}}, \quad \text{using (6.47)} \\
&=\; \left(\sqrt{\xi} + \sqrt{\frac{\xi}{\xi - 1}}\right) \frac{\sqrt{\vartheta}}{(1-\bar{\epsilon})^3 \mu}.
\end{aligned}
\tag{6.48}
$$

(6.48) gives a bound on the displacement in $Ax + \frac{1}{\tau} z^0$ because by (6.24)

$$
Ax^+ + \frac{1}{\tau^+} z^0 - Ax - \frac{1}{\tau} z^0 = \frac{\alpha_2}{\tau + \alpha_2 d_\tau} \left[A\bar{d}_x - d_\tau \left(Ax + \frac{1}{\tau} z^0\right)\right], \tag{6.49}
$$

by choosing $\alpha_1 = \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$. Also from (6.45) we get

$$
\frac{\tau^2}{\mu^2} \langle d_y, [\Phi'']^{-1} d_y \rangle \leq q = \frac{1}{(1-\bar{\epsilon})^6} \frac{\xi \vartheta}{\mu^2}. \tag{6.50}
$$

We use (6.47)-(6.49) to give the bound we want on the step length. In view of Corollary 6.2.1, we are interested in the change of

$$\left\| \frac{\tau^+ y^+}{\mu^+} - \Phi'\left(u^+\right) \right\|_{[\Phi''(u^+)]^{-1}}. \tag{6.51}$$

By adding and subtracting some terms, we have

$$\frac{\tau^+ y^+}{\mu^+} - \Phi'\left(u^+\right) - \frac{\tau y}{\mu} + \Phi'\left(u\right) = \left(\frac{\tau^+}{\mu^+} - \frac{\tau}{\mu}\right) y + \frac{\tau^+}{\mu^+}\alpha_2 d_y - \left(\Phi'\left(u^+\right) - \Phi'\left(u\right)\right). \tag{6.52}$$

Let us give a bound on the three terms in (6.52). For the first term, using Proposition 6.2.1, we have

$$\left|\frac{\tau + \alpha_2 d_\tau}{\mu + d_\mu} - \frac{\tau}{\mu}\right| = \left|\frac{\alpha_2 \mu d_\tau - \tau d_\mu}{\mu(\mu + d_\mu)}\right| \leq \alpha_2 \left(\left|\frac{d_\tau}{\tau}\right| + \left|\frac{1}{\mu(1-\bar{\epsilon})^2}\right|\right) \frac{\tau}{\mu}. \tag{6.53}$$

By property (4.16), if we assume that $\Omega(x,\tau,y) \leq \delta_1$, we have

$$[\Phi''(u)]^{-1} \preceq \frac{1}{(1-\sigma(\delta_1))^2} \Phi''_*\left(\frac{\tau y}{\mu}\right). \tag{6.54}$$

Using (6.53) and (6.54), we have

$$\begin{aligned}
\left|\frac{\tau^+}{\mu^+} - \frac{\tau}{\mu}\right| \|y\|_{[\Phi''(u)]^{-1}} &\leq \frac{\alpha_2}{(1-\sigma(\delta_1))^2}\left(\left|\frac{d_\tau}{\tau}\right| + \left|\frac{1}{\mu(1-\bar{\epsilon})^2}\right|\right) \left\|\frac{\tau}{\mu} y\right\|_{\Phi''_*} \\
&\leq \frac{\alpha_2}{(1-\sigma(\delta_1))^2}\left(\left|\frac{d_\tau}{\tau}\right| + \left|\frac{1}{\mu(1-\bar{\epsilon})^2}\right|\right) \sqrt{\vartheta}, \quad \text{using Theorem 4.2.1-(1)} \\
&\leq \left(\sqrt{\frac{\xi}{\xi-1}} \frac{1}{(1-\bar{\epsilon})^3} + \frac{1}{(1-\bar{\epsilon})^2}\right) \frac{\alpha_2}{(1-\sigma(\delta_1))^2} \frac{\sqrt{\vartheta}}{\mu}, \quad \text{using (6.47).}
\end{aligned} \tag{6.55}$$

For the second term in (6.52) we have

$$\begin{aligned}
\frac{\tau^+}{\mu^+}\alpha_2 \|d_y\|_{[\Phi''(u)]^{-1}} &\leq \left[1 + \alpha_2\left(\left|\frac{d_\tau}{\tau}\right| + \left|\frac{1}{\mu(1-\bar{\epsilon})^2}\right|\right)\right] \alpha_2 \left\|\frac{\tau}{\mu} d_y\right\|_{[\Phi''(u)]^{-1}}, \quad \text{using (6.53)} \\
&\leq \left[1 + \alpha_2\left(\left|\frac{d_\tau}{\tau}\right| + \left|\frac{1}{\mu(1-\bar{\epsilon})^2}\right|\right)\right] \alpha_2 \frac{\sqrt{\xi\vartheta}}{(1-\bar{\epsilon})^3 \mu}, \quad \text{using (6.50)} \\
&\leq \left[1 + \left(\sqrt{\frac{\xi}{\xi-1}} \frac{1}{(1-\bar{\epsilon})^3} + \frac{1}{(1-\bar{\epsilon})^2}\right) \frac{\alpha_2}{\mu}\right] \alpha_2 \frac{\sqrt{\xi\vartheta}}{(1-\bar{\epsilon})^3 \mu}, \quad \text{using (6.47).}
\end{aligned} \tag{6.56}$$

For the third term, first by using (6.49) and substituting the bound in (6.48) we have

$$\begin{aligned}\left\|u^{+}-u\right\|_{\Phi''} &= \frac{\alpha_2\tau}{\tau+\alpha_2 d_\tau}\left(\sqrt{\xi}+\sqrt{\frac{\xi}{\xi-1}}\right)\frac{\sqrt{\vartheta}}{(1-\bar{\epsilon})^3\mu}\\ &\leq \underbrace{\frac{1}{1-\sqrt{\frac{\xi}{\xi-1}}\frac{\alpha_2}{(1-\bar{\epsilon})^3\mu}}\left(\sqrt{\xi}+\sqrt{\frac{\xi}{\xi-1}}\right)\frac{\alpha_2\sqrt{\vartheta}}{(1-\bar{\epsilon})^3\mu}}_{=:\bar{\delta}}, \quad \text{using (6.47).} \end{aligned} \tag{6.57}$$

If we choose α_2 such that $\bar{\delta}<1$, then, by Lemma 4.2.2, we have

$$\left\|\Phi'\left(u^{+}\right)-\Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}} \leq \frac{\bar{\delta}}{1-\bar{\delta}}. \tag{6.58}$$

Putting together the above bounds, we can prove the following main result:

Proposition 6.2.2. *Assume that in the predictor step of the algorithm, we choose* $\alpha_1=\frac{\alpha_2}{\tau+\alpha_2 d_\tau}$*. Also assume that* $0.1>\delta_2>4\delta_1>0$*. Then, there exists a positive constant* κ_1 *depending on* δ_1*,* δ_2*,* ξ*, and* $\bar{\epsilon}$ *such that*

$$\alpha_2 \geq \frac{\kappa_1}{\sqrt{\vartheta}}\mu. \tag{6.59}$$

Proof. Consider the bound we have for the proximity measure in Corollary 6.2.1. If we choose α_2 such that $\bar{\delta}$ defined in (6.57) satisfies $\bar{\delta}\leq 1/4$, then using property (4.16), we have

$$\begin{aligned}&\left\|\frac{\tau^{+}y^{+}}{\mu^{+}}-\Phi'\left(u^{+}\right)\right\|_{[\Phi''(u^{+})]^{-1}}\\ \leq\ & \frac{4}{3}\left\|\frac{\tau^{+}y^{+}}{\mu^{+}}-\Phi'\left(u^{+}\right)\right\|_{[\Phi''(u)]^{-1}}\\ \leq\ & \frac{4}{3}\left\|\frac{\tau^{+}y^{+}}{\mu^{+}}-\Phi'\left(u^{+}\right)-\frac{\tau y}{\mu}+\Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}}+\frac{4}{3}\left\|\frac{\tau y}{\mu}+\Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}}\\ \leq\ & \frac{4}{3}\left\|\frac{\tau^{+}y^{+}}{\mu^{+}}-\Phi'\left(u^{+}\right)-\frac{\tau y}{\mu}+\Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}}+\frac{4}{3}\sigma(\delta_1).\end{aligned} \tag{6.60}$$

Similar to the definition of $\sigma(\cdot)$ as the inverse of $\rho(\cdot)$ in (4.11), we define the inverse of $\rho(-\cdot)$ as $\bar{\sigma}(\cdot)$. To satisfy $\Omega(x^{+},\tau^{+},y^{+})\leq\delta_2$, a sufficient condition is

$$\left\|\frac{\tau^{+}y^{+}}{\mu^{+}}-\Phi'\left(u^{+}\right)-\frac{\tau y}{\mu}+\Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}} \leq \frac{3}{4}\bar{\sigma}(\delta_2)-\sigma(\delta_1). \tag{6.61}$$

Note that for this analysis, we need $\frac{3}{4}\bar{\sigma}(\delta_2) > \sigma(\delta_1)$. One way to enforce that is choosing $0.1 > \delta_2 > 4\delta_1$. We have split the term inside the norm in the LHS of (6.61) into three terms in (6.52) and bounded each of them. We want the summation of the bounds in (6.55), (6.56), and (6.58) be at most $\frac{3}{4}\bar{\sigma}(\delta_2) - \sigma(\delta_1) < 1$. In view of the first one in (6.55), we assume that

$$\frac{\alpha_2\sqrt{\vartheta}}{\mu} \leq \underbrace{\frac{1}{2}\left(\sqrt{\frac{\xi}{\xi-1}}\frac{1}{(1-\bar{\epsilon})^3} + \frac{1}{(1-\bar{\epsilon})^2}\right)^{-1}}_{=:\kappa_1^1}. \tag{6.62}$$

Then, adding the bounds in (6.55), (6.56), and (6.58) gives us

$$\begin{aligned} &\left\| \frac{\tau^+ y^+}{\mu^+} - \Phi'\left(u^+\right) - \frac{\tau y}{\mu} + \Phi'\left(u\right) \right\|_{[\Phi''(u)]^{-1}} \\ \leq\ & \underbrace{\left[\frac{\left(\sqrt{\frac{\xi}{\xi-1}}\frac{1}{(1-\bar{\epsilon})^3} + \frac{1}{(1-\bar{\epsilon})^2}\right)}{(1-\sigma(\delta_1))^2} + \frac{3}{2}\frac{\sqrt{\xi}}{(1-\bar{\epsilon})^3} + 4\left(\sqrt{\xi} + \sqrt{\frac{\xi}{\xi-1}}\right)\frac{1}{(1-\bar{\epsilon})^3}\right]}_{=:1/\kappa_1^2}\frac{\alpha_2\sqrt{\vartheta}}{\mu}. \end{aligned} \tag{6.63}$$

We assumed at the beginning of the proof and we also used for (6.58) that $\bar{\delta} \leq 1/4$, which holds if the RHS of (6.63) is at most 1. Therefore, if we choose

$$\frac{\alpha_2\sqrt{\vartheta}}{\mu} = \kappa_1 := \min\left\{\kappa_1^1, \kappa_1^2\left(\frac{3}{4}\bar{\sigma}(\delta_2) - \sigma(\delta_1)\right)\right\},$$

then $\Omega(x^+, \tau^+, y^+) \leq \delta_2$ holds. This implies that (6.59) holds for a constant κ_1 that depends on δ_1, δ_2, ξ, and $\bar{\epsilon}$. $\square$

To complete the whole discussion, we need to prove Lemma 6.1.1. Let us start with the following lemma:

Lemma 6.2.5. *For every set of points (z, τ, y, y_τ) such that $u := \frac{z}{\tau} \in D$, $y \in D_*$, and y_τ satisfies $y_\tau + \frac{1}{\tau}\langle y, z\rangle + \frac{\mu\xi\vartheta}{\tau} = 0$, we have*

$$\left\|\frac{\tau y}{\mu} - \Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}} \leq \beta \leq \sqrt{\frac{\xi}{\xi-1}}\left\|\frac{\tau y}{\mu} - \Phi'\left(u\right)\right\|_{[\Phi''(u)]^{-1}}, \tag{6.64}$$

where

$$\beta(y, y_\tau, z, \tau, \mu) := \left\| \frac{1}{\mu} \begin{bmatrix} y \\ y_\tau \end{bmatrix} - \begin{bmatrix} \frac{1}{\tau}\Phi'(u) \\ -\frac{1}{\tau}\langle \Phi'(u), u\rangle - \frac{\xi\vartheta}{\tau} \end{bmatrix} \right\|_{\frac{1}{\mu^2}[\bar{H}(u,\tau)]^{-1}}, \tag{6.65}$$

for $\bar{H}(u, \tau)$ *defined in* (6.4) *as a function of* u *and* τ.

Proof. Consider the definition of $\bar{H}$ in (6.4) and the formula for its inverse in (6.7). If we substitute $f = \frac{y}{\mu} - \frac{1}{\tau}\Phi'(u)$ and $f_\tau = \frac{y_\tau}{\mu} + \frac{1}{\tau}\langle \Phi'(u), u\rangle + \frac{\xi\vartheta}{\tau}$, we have

$$\begin{aligned}
\langle f, H^{-1} f\rangle &= \left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|^2_{[\Phi''(u)]^{-1}} \\
\eta\left(\langle f, H^{-1}h\rangle - f_\tau\right)^2 &= \frac{\tau^2\left[\langle \frac{y}{\mu} - \frac{1}{\tau}\Phi', [\Phi'']^{-1}\Phi'\rangle - \frac{1}{\mu}\left(y_\tau + \frac{1}{\tau}\langle y, z\rangle + \frac{\mu\xi\vartheta}{\tau}\right)\right]^2}{\xi\vartheta - \langle \Phi', [\Phi'']^{-1}\Phi'\rangle} \\
&= \frac{\tau^2\left[\langle \frac{y}{\mu} - \frac{1}{\tau}\Phi', [\Phi'']^{-1}\Phi'\rangle\right]^2}{\xi\vartheta - \langle \Phi', [\Phi'']^{-1}\Phi'\rangle}, \quad \text{using } y_\tau + \frac{1}{\tau}\langle y, z\rangle + \frac{\mu\xi\vartheta}{\tau} = 0.
\end{aligned} \tag{6.66}$$

Hence,

$$\begin{aligned}
\beta^2 &= \left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|^2_{[\Phi''(u)]^{-1}} + \frac{\left[\langle \frac{\tau y}{\mu} - \Phi', [\Phi'']^{-1}\Phi'\rangle\right]^2}{\xi\vartheta - \langle \Phi', [\Phi'']^{-1}\Phi'\rangle} \\
&\leq \left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|^2_{[\Phi''(u)]^{-1}} + \frac{\left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|^2_{[\Phi''(u)]^{-1}} \vartheta}{(\xi - 1)\vartheta} \\
&= \frac{\xi}{\xi - 1} \left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|^2_{[\Phi''(u)]^{-1}},
\end{aligned} \tag{6.67}$$

where for the inequality we used Cauchy-Schwarz inequality and property (4.25) of ϑ-s.c. barriers $\langle \Phi', [\Phi'']^{-1}\Phi'\rangle \leq \vartheta$. (6.67) immediately gives us the inequalities we want. ☐

Proof of Lemma 6.1.1. Assume that $\Omega(x, \tau, y) \leq \epsilon < 1$, by Lemma 6.2.1, we get

$$\rho\left(\left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|_{[\Phi''(u)]^{-1}}\right) \leq \epsilon \;\Rightarrow\; \left\| \frac{\tau y}{\mu} - \Phi'(u) \right\|_{[\Phi''(u)]^{-1}} \leq \sigma(\epsilon),$$

where $\sigma(\cdot)$, defined in (4.11), is the inverse of $\rho(\cdot)$ for nonnegative values.

Hence, by Lemma 6.2.5, we have $\beta \leq \sqrt{\frac{\xi}{\xi-1}}\sigma(\epsilon)$. In Remark 6.1.2, we mentioned that $\bar{H}(x,\tau)$, with some change of variables, is the Hessian of $\Phi\left(\frac{z}{\tau}\right)-\xi\vartheta\ln(\tau)$, which we proved in Lemma 4.2.3 that is a $\bar{\xi}$-s.c. function. We want to use Lemma 6.2.1 for $f=\Phi\left(\frac{z}{\tau}\right)-\xi\vartheta\ln(\tau)$ and its conjugate at the points $(z,\tau):=(\tau Ax+z^0,\tau)$ and $\frac{1}{\mu}(y,y_\tau):=\frac{1}{\mu}(y,y_\tau^0+\tau\langle c,x\rangle)$, and their counterparts on the central path. One can verify that condition of Lemma 6.2.1-(b) holds for these points, i.e.,

$$\begin{aligned}
& \langle y-y(\mu), z-z(\mu)\rangle + (y_\tau - y_\tau(\mu))(\tau-\tau(\mu)) \\
= \; & \langle A^\top(y-y(\mu)), \tau x - \tau(\mu)x(\mu)\rangle + (\tau-\tau(\mu))\langle c, \tau x - \tau(\mu)x(\mu)\rangle \\
= \; & -(\tau-\tau(\mu))\langle c, \tau x - \tau(\mu)x(\mu)\rangle + (\tau-\tau(\mu))\langle c, \tau x - \tau(\mu)x(\mu)\rangle \\
= \; & 0. \qquad (6.68)
\end{aligned}$$

Note that the terms in the middle of both parts (a) and (b) of Lemma 6.2.1 are the same. If we combine them and ignore one term in the LHS, we get

$$\begin{aligned}
& \rho\left(\frac{1}{\sqrt{\bar{\xi}}}\left(\bar{H}(x(\mu),\tau(\mu))[z-z(\mu),\tau-\tau(\mu)]\right)^{1/2}\right) \leq \rho\left(-\frac{\beta}{\sqrt{\bar{\xi}}}\right) \\
\Rightarrow \; & \left(\bar{H}(x(\mu),\tau(\mu))[z-z(\mu),\tau-\tau(\mu)]\right)^{1/2} \leq \sqrt{\bar{\xi}}\sigma\left(\rho\left(-\frac{1}{\sqrt{\bar{\xi}}}\sqrt{\frac{\xi}{\xi-1}}\sigma(\epsilon)\right)\right).
\end{aligned} \qquad (6.69)$$

Now we just need to use property (4.16) of s.c. functions to get the result of the lemma. □

Before analyzing the corrector step, let us elaborate more on the RHS of (6.69). We have a bound on $\sigma(\epsilon) \leq \sqrt{2\epsilon}+\epsilon$ in (4.12), and for $\epsilon \leq 1$ we can easily verify that $\sqrt{2\epsilon}+\epsilon \leq 3\sqrt{\epsilon}$. Also we can verify that for $t \leq 0.6$, we have $\rho(-t) \leq t^2$. Assume that $\sigma(\epsilon)$ is small enough to have $\sqrt{\frac{\xi}{\bar{\xi}(\xi-1)}}\sigma(\epsilon) \leq 0.6$. Then, the RHS of (6.69) becomes

$$\leq \sqrt{\bar{\xi}}\sigma\left(\frac{\xi}{\bar{\xi}(\xi-1)}\sigma^2(\epsilon)\right) \leq 9\sqrt{\frac{\xi}{\xi-1}}\sqrt{\epsilon}. \qquad (6.70)$$

For a point $(x,\tau,y) \in Q_{DD}$ with parameter μ, let us define

$$d := \begin{bmatrix} \tau(\mu)x(\mu)-\tau x \\ \tau(\mu)-\tau \\ v(\mu)-v \end{bmatrix}. \qquad (6.71)$$

We can easily verify that ($y = y^0 - (\tau - 1)\bar{c} - F^\top v$):

$$Ud = \begin{bmatrix} \tau(\mu)Ax(\mu) + z^0 \\ \tau(\mu) \\ y(\mu) \\ y_\tau^0 + \tau(\mu)\langle c, x(\mu)\rangle \end{bmatrix} - \begin{bmatrix} \tau Ax + z^0 \\ \tau \\ y \\ y_\tau^0 + \tau\langle c, x\rangle \end{bmatrix}. \tag{6.72}$$

We want to use property (4.16) for $r = 1/4$ to change the local norm in (6.69); it suffices to enforce $9\sqrt{\frac{\xi}{\bar{\xi}(\xi-1)}}\sqrt{\epsilon} \le \frac{1}{4}$ in view of (6.70). Consider the proof of Lemma 6.1.1 and also the term for y that we ignored in (6.69). If $\Omega(x, \tau, y) \le \epsilon < 1$, we get

$$\|d\|_{U^\top \mathcal{H}(\bar{H}(x,\tau), \mu^2 \bar{H}(x,\tau))U} \le \underbrace{24\sqrt{\frac{\xi}{\xi - 1}}}_{=:\bar{\xi}_1}\sqrt{\epsilon}, \tag{6.73}$$

which we use in the analysis of the corrector step.

6.2.2 Corrector step

We focus on the case that $\alpha_1 = \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$. By Remark 6.1.3, $\mu^+ = \mu$ for every α_2 and so we just need to show that α_2 can be chosen to get enough reduction in the proximity measure. The most important tool for this analysis is inequality (6.23). In the proof of Lemma 6.1.2, we used one direction of property (4.17). Using the other direction, we get (6.23) where the direction of inequality is changed and $\rho(-\cdot)$ is replaced with $\rho(\cdot)$. If we combine both inequalities, we get

$$\begin{aligned} & \rho(D(\alpha_2)) \\ \le\ & \Omega(x^+, \tau^+, y^+) - \Omega(x, \tau, y) - \alpha_2 \begin{bmatrix} \bar{d}_x^\top & d_\tau & d_v^\top \end{bmatrix} U^\top \psi^c \\ & + \tfrac{\alpha_2^2 d_\tau}{\tau(\tau + \alpha_2 d_\tau)} \langle \Phi', A\bar{d}_x - d_\tau \left(Ax + \tfrac{1}{\tau} z^0\right)\rangle - \tfrac{\alpha_2^2 d_\tau}{\mu}\left(\langle d_y, \Phi'_*\rangle + \langle c, \bar{d}_x\rangle\right) \\ \le\ & \rho\left(-D(\alpha_2)\right), \end{aligned}$$

$$D(\alpha_2) := \frac{\alpha_2}{\tau + \alpha_2 d_\tau}\left\| A\bar{d}_x - d_\tau\left(Ax + \frac{1}{\tau}z^0\right)\right\|_{\Phi''} + \alpha_2 \left\|\frac{d_\tau y + (\tau + \alpha_2 d_\tau) d_y}{\mu}\right\|_{\Phi''_*}. \tag{6.74}$$

Let d be the corrector step derived in (6.19). Then,

$$-\begin{bmatrix} \bar{d}_x^\top & d_\tau & d_v^\top \end{bmatrix} U^\top \psi^c = \left\| U^\top \psi^c + \beta r^0 \right\|^2_{(U^\top \mathcal{H} U)^{-1}}. \tag{6.75}$$

On the other hand, $-\|U^\top \psi^c + \beta r^0\|_{(U^\top \mathcal{H} U)^{-1}}$ is the optimal objective value of (6.28). We give a lower bound for this quantity in the following lemma:

Lemma 6.2.6. *Consider $(x,\tau,y) \in Q_{DD}$ and $0 < \epsilon < 1$ such that $\Omega(x,\tau,y) = \epsilon$. Consider β defined in (6.19) and ψ^c defined in (6.20). Then, for ϵ small enough, i.e., $\epsilon \leq \frac{1}{100\left((\bar{\xi}_2\bar{\xi}_1)^3 + \bar{\xi}_3\bar{\xi}_1^3\right)^2}$, where $\bar{\xi}_1$ is defined in (6.70) and*

$$\begin{aligned} \bar{\xi}_2 &:= \left(3\sqrt{\frac{1}{\xi-1}} + \frac{7}{2}\right), \\ \bar{\xi}_3 &:= \frac{1}{2\sqrt{\xi-1}}\left(\frac{11}{2} + \frac{5}{\sqrt{\xi-1}}\right)\left(3 + \frac{2}{\sqrt{\xi-1}}\right) + \frac{2}{\xi-1}\left(1 + \frac{1}{\sqrt{\xi-1}}\right), \end{aligned} \tag{6.76}$$

we have

$$\left\|U^\top \psi^c + \beta r^0\right\|_{(U^\top \mathcal{H} U)^{-1}} \geq \frac{1}{4\bar{\xi}_1}\sqrt{\epsilon}.$$

Proof. We find an upper bound on the optimal objective value of (6.28) by using a specific feasible solution. Our feasible solution is

$$\frac{d}{\|d\|_{U^\top \mathcal{H}(\bar{H},\mu^2\bar{H})U}}, \tag{6.77}$$

where d is defined in (6.71). We can verify that (6.77) satisfies all the constraints. We have $\|d\|_{U^\top \mathcal{H}(\bar{H},\mu^2\bar{H})U} \leq \bar{\xi}_1\sqrt{\epsilon}$ given in (6.73), which holds if $\sqrt{\frac{\xi}{\bar{\xi}(\xi-1)}}\sigma(\epsilon) \leq 0.6$ and $9\sqrt{\frac{\xi}{\bar{\xi}(\xi-1)}}\sqrt{\epsilon} \leq \frac{1}{4}$. Now, we need to prove that $-\langle d, U^\top \psi^c\rangle$ is large enough. The idea of the proof is that we consider the bounds in (6.74) at $\alpha_2 = 1$ and $\alpha_2 = 2$, and if $-\langle d, U^\top \psi^c\rangle$ is not large enough, we get a contradiction.

Let us define $\sqrt{q} = \bar{\xi}_1\sqrt{\epsilon}$ for $\bar{\xi}_1$ defined in (6.73). Then (6.73) implies that we have (6.45) for q. We use this to find bounds for the terms we have in (6.74). Using both (6.47) and (6.48), we have

$$\frac{1}{\tau + \alpha_2 d_\tau}\left\|A\bar{d}_x - d_\tau\left(Ax + \frac{1}{\tau}z^0\right)\right\|_{\Phi''} \leq \frac{1}{1 - \alpha_2\sqrt{\frac{q}{(\xi-1)\vartheta}}}\left(\sqrt{q} + \sqrt{\frac{q}{\xi-1}}\right). \tag{6.78}$$

For the second term of $D(\alpha_2)$ we use triangle inequality and we have

$$\left\|\frac{d_\tau y}{\mu}\right\|_{\Phi''_*} = \frac{d_\tau}{\tau}\left\|\frac{\tau y}{\mu}\right\|_{\Phi''_*} \leq \sqrt{\frac{q}{\xi-1}}, \quad \text{using (6.47) and (4.39)} \tag{6.79}$$

and

$$\begin{aligned}\left\|\frac{(\tau+\alpha_2 d_\tau)d_y}{\mu}\right\|_{\Phi''_*} &= \frac{(\tau+\alpha_2 d_\tau)}{\tau}\left\|\frac{\tau d_y}{\mu}\right\|_{\Phi''_*}\\ &\le \left(1+\alpha_2\sqrt{\frac{q}{(\xi-1)\vartheta}}\right)\sqrt{q}, \quad \text{using (6.47) and (6.50).} \quad (6.80)\end{aligned}$$

We also have

$$\begin{aligned}&\left|\frac{d_\tau}{\tau(\tau+\alpha_2 d_\tau)}\langle\Phi', A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\rangle\right|\\ \le\ &\left|\frac{d_\tau}{\tau(\tau+\alpha_2 d_\tau)}\right|\|\Phi'\|_{[\Phi'']^{-1}}\left\|A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\right\|_{\Phi''}\\ \le\ &\left|\frac{\frac{d_\tau}{\tau}\sqrt{\vartheta}}{\tau+\alpha_2 d_\tau}\right|\left\|A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\right\|_{\Phi''}, \quad \text{by (4.25)}\\ \le\ &\frac{\sqrt{\frac{q}{\xi-1}}}{1-\alpha_2\sqrt{\frac{q}{(\xi-1)\vartheta}}}\left(\sqrt{q}+\sqrt{\frac{q}{\xi-1}}\right), \quad \text{by (6.47) and (6.78).} \quad (6.81)\end{aligned}$$

We want to make the second line of the term in the middle of inequalities in (6.74) a quadratic in terms of α_2. To do this, we modify (6.74) by adding and subtracting some terms to all sides as:

$$\begin{aligned}&\rho(D(\alpha_2)) - \tfrac{1}{2}(D(\alpha_2))^2 + \hat{D}(\alpha_2)\\ \le\ &\Omega(x^+,\tau^+,y^+) - \Omega(x,\tau,y) - \alpha_2\begin{bmatrix}\bar{d}_x^\top & d_\tau & d_v^\top\end{bmatrix}U^\top\psi^c\\ &+\tfrac{\alpha_2^2 d_\tau}{\tau^2}\langle\Phi', A\bar{d}_x - d_\tau\left(Ax+\tfrac{1}{\tau}z^0\right)\rangle - \tfrac{\alpha_2^2 d_\tau}{\mu}\left(\langle d_y,\Phi'_*\rangle + \langle c,\bar{d}_x\rangle\right) - \tfrac{1}{2}(\bar{D}(\alpha_2))^2\\ \le\ &\rho\left(-D(\alpha_2)\right) - \tfrac{1}{2}(D(\alpha_2))^2 + \hat{D}(\alpha_2),\end{aligned}$$

$$\begin{aligned}\bar{D}(\alpha_2) &:= \frac{\alpha_2}{\tau}\left\|A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\right\|_{\Phi''} + \alpha_2\left\|\frac{d_\tau y+\tau d_y}{\mu}\right\|_{\Phi''_*},\\ \hat{D}(\alpha_2) &:= \frac{1}{2}\left((D(\alpha_2))^2 - (\bar{D}(\alpha_2))^2\right) + \frac{\alpha_3^2 d_\tau^2}{\tau^2(\tau+\alpha_2 d_\tau)}\langle\Phi', A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\rangle. \quad (6.82)\end{aligned}$$

Note that by definition (4.10), we can verify that

$$\rho(-t) - \frac{t^2}{2} \le t^3, \qquad \frac{t^2}{2} - \rho(t) \le \frac{t^3}{5}, \quad \forall t\in(0,0.8). \quad (6.83)$$

Let us assume that $2\sqrt{\frac{q}{(\xi-1)\vartheta}} \leq \frac{1}{2}$, then (6.78), (6.79), (6.80), and (6.81) yield that for $0 \leq \alpha_2 \leq 2$ we have

$$\begin{aligned}
|D(\alpha_2)| &\leq \alpha_2 \underbrace{\left(3\sqrt{\frac{1}{\xi-1}}+\frac{7}{2}\right)}_{=:\bar{\xi}_2} \bar{\xi}_1\sqrt{\epsilon}, \\
|\bar{D}(\alpha_2)| &\leq \alpha_2 \left(2\sqrt{\frac{1}{\xi-1}}+2\right) \bar{\xi}_1\sqrt{\epsilon}, \\
|\hat{D}(\alpha_2)| &\leq \alpha_2^3 \underbrace{\left[\frac{1}{2\sqrt{\xi-1}}\left(\frac{11}{2}+\frac{5}{\sqrt{\xi-1}}\right)\left(3+\frac{2}{\sqrt{\xi-1}}\right)+\frac{2}{\xi-1}\left(1+\frac{1}{\sqrt{\xi-1}}\right)\right]}_{=:\bar{\xi}_3} \bar{\xi}_1^3\epsilon^{3/2}.
\end{aligned} \tag{6.84}$$

For the bound on $|\hat{D}(\alpha_2)|$, we also used the fact that

$$|D(\alpha_2)-\bar{D}(\alpha_2)| \leq \frac{\alpha_2^2 d_\tau}{(\tau+\alpha_2 d_\tau)\tau}\left\|A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\right\|_{\Phi''} + \alpha_2^2\left\|\frac{d_\tau d_y}{\mu}\right\|_{\Phi''_*}.$$

We want to choose ϵ small enough to make the term in the middle of (6.82) be squeezed between $\pm\frac{1}{10}\Omega(x,\tau,y) = \pm\frac{1}{10}\epsilon$. If we have $\bar{\xi}_2\bar{\xi}_1\sqrt{\epsilon} \leq 0.8$, by using (6.83) and (6.84), it suffices to satisfy

$$\left((\bar{\xi}_2\bar{\xi}_1)^3+\bar{\xi}_3\bar{\xi}_1^3\right)\epsilon^{3/2} \leq \frac{1}{10}\epsilon \quad\Rightarrow\quad \epsilon \leq \frac{1}{100\left((\bar{\xi}_2\bar{\xi}_1)^3+\bar{\xi}_3\bar{\xi}_1^3\right)^2}. \tag{6.85}$$

We claim that in this case, $-\langle d, U^\top\psi^c\rangle \geq \frac{1}{4}\Omega(x,\tau,y)$. If we substitute $\alpha_2 = 1$, then $\Omega(x^+,\tau^+,y^+) = 0$ as our point lays on the central path. Suppose for the sake of reaching a contradiction $-\langle d, U^\top\psi^c\rangle < \frac{1}{4}\Omega(x,\tau,y)$. Then, in view of (6.82), we must have

$$\frac{d_\tau}{\tau^2}\langle\Phi', A\bar{d}_x - d_\tau\left(Ax+\frac{1}{\tau}z^0\right)\rangle - \frac{d_\tau}{\mu}\left(\langle\Phi'_*, d_y\rangle+\langle c,\bar{d}_x\rangle\right) - \frac{1}{2}(\bar{D}(1))^2 \geq \left(\frac{3}{4}-\frac{1}{10}\right)\Omega(x,\tau,y).$$

We reach our contradiction when we consider $\alpha_2 = 2$. Note that for $\alpha_2 = 2$ we have $\Omega(x^+,\tau^+,y^+) \geq 0$. The term in the second line of (6.74) is degree 2 of α_2 and so becomes at least $\left(\frac{12}{4}-\frac{4}{10}\right)\Omega(x,\tau,y)$ for $\alpha_2 = 2$. Then, at $\alpha_2 = 2$, (6.74) implies

$$-\Omega(x,\tau,y)+\left(\frac{12}{4}-\frac{4}{10}\right)\Omega(x,\tau,y) \leq \frac{8}{10}\Omega(x,\tau,y),$$

which is a contradiction.

Now, if we consider the feasible solution (6.71) for the optimization problem (6.28) and putting together the bounds $\|d\|_{U^\top \mathcal{H}(\bar{H},\mu^2\bar{H})U} \leq \bar{\xi}_1\sqrt{\epsilon}$ and $-\langle d, U^\top \psi^c\rangle \geq \frac{1}{4}\epsilon$, we get the result of the lemma. Note that the bound on ϵ in (6.85) is stronger than the other assumptions we made in the proof. $\square$

Now we are ready to prove the main proposition for the corrector step.

Proposition 6.2.3. *Consider $(x,\tau,y) \in Q_{DD}$ and $0 < \epsilon < 1$ such that $\Omega(x,\tau,y) = \epsilon$. Assume that the corrector step is calculated by solving (6.19) and we choose $\alpha_1 = \frac{\alpha_2}{\tau+\alpha_2 d_\tau}$. Also assume that ϵ is small in the sense defined in Lemma 6.2.6. Cosider $\bar{\xi}_1$ and $\bar{\xi}_2$ defined in (6.70) and (6.76), respectively. Then, if we choose*

$$\alpha_2 := \frac{1}{2(\bar{\xi}_4+\bar{\xi}_2^2)}, \quad \bar{\xi}_4 := 2\sqrt{\frac{1}{\xi-1}}\left(1+\sqrt{\frac{1}{\xi-1}}\right)+\frac{\sqrt{\xi}+2}{\sqrt{\xi}-1}, \tag{6.86}$$

we have

$$\Omega(x^+,\tau^+,y^+) - \Omega(x,\tau,y) \leq -\frac{\alpha_2}{32\bar{\xi}_1^2}. \tag{6.87}$$

Proof. Assume that $[\bar{d}_x^\top \; d_\tau \; d_v^\top]$ is the solution of (6.19). Then we have

$$[\bar{d}_x^\top \; d_\tau \; d_v^\top]U^\top \mathcal{H} U \begin{bmatrix} \bar{d}_x \\ d_\tau \\ d_v \end{bmatrix} = \|U^\top \psi^c + \beta r^0\|^2_{(U^\top \mathcal{H} U)^{-1}}. \tag{6.88}$$

Hence, we have inequality (6.45) with $q = \|U^\top\psi^c + \beta r^0\|^2_{(U^\top \mathcal{H}U)^{-1}}$, and we already have the bounds (6.78), (6.79), (6.80), and (6.81).

Here, we use (6.45) to get another bound; if we consider the last term in the LHS of (6.45), we get

$$\frac{\tau}{\mu}\left|\langle d_y, Ax + \frac{1}{\tau}z^0\rangle + \langle c, \bar{d}_x\rangle\right| \leq (\sqrt{\xi}+1)\sqrt{\vartheta q}. \tag{6.89}$$

Note that from Corollary 6.2.1, we have $\left\|\frac{\tau y}{\mu} - \Phi'\right\|_{[\Phi'']^{-1}} \leq \sigma(\epsilon)$. Using this and (6.89), we have

$$
\begin{aligned}
& \left|\frac{d_\tau}{\mu}\left(\langle d_y, \Phi'_*\rangle + \langle c, \bar{d}_x\rangle\right)\right| \\
= & \left|\frac{d_\tau}{\mu}\left(\langle d_y, Ax + \frac{1}{\tau}z^0\rangle + \langle c, \bar{d}_x\rangle + \langle d_y, \Phi'_* - Ax + \frac{1}{\tau}z^0\rangle\right)\right| \\
\leq & \left|\frac{d_\tau}{\tau}\right|\left(\frac{\tau}{\mu}\left|\langle d_y, Ax + \frac{1}{\tau}z^0\rangle + \langle c, \bar{d}_x\rangle\right| + \frac{\tau}{\mu}\|d_y\|_{[\Phi'']^{-1}}\left\|\Phi'_* - Ax + \frac{1}{\tau}z^0\right\|_{\Phi''}\right) \\
\leq & \sqrt{\frac{q}{(\xi-1)\vartheta}}\left((\sqrt{\xi}+1)\sqrt{\vartheta q} + \frac{\tau}{\mu}\|d_y\|_{[\Phi'']^{-1}}\left\|\Phi'_* - Ax + \frac{1}{\tau}z^0\right\|_{\Phi''}\right), \text{ by (6.47) and (6.89)} \\
\leq & \sqrt{\frac{q}{(\xi-1)\vartheta}}\left((\sqrt{\xi}+1)\sqrt{\vartheta q} + \sqrt{q}\frac{\sigma(\epsilon)}{1-\sigma(\epsilon)}\right), \text{ by (6.50) and Lemma 4.2.2,} \\
= & \sqrt{\frac{1}{\xi-1}}\left((\sqrt{\xi}+1) + \frac{\sigma(\epsilon)}{\sqrt{\vartheta}(1-\sigma(\epsilon))}\right)q \\
\leq & \frac{\sqrt{\xi}+2}{\sqrt{\xi-1}}q, \quad \text{for the case } \sigma(\epsilon) \leq 0.5.
\end{aligned}
\tag{6.90}
$$

We want to work with the second inequality in (6.74). We already have a bound for $D(\alpha_2)$ in (6.84) and we have

$$
\rho(-t) \leq t^2, \quad \forall t \in (0, 0.6). \tag{6.91}
$$

In view of (6.81) and (6.90), we define

$$
\bar{\xi}_4 := 2\sqrt{\frac{1}{\xi-1}}\left(1 + \sqrt{\frac{1}{\xi-1}}\right) + \frac{\sqrt{\xi}+2}{\sqrt{\xi-1}}. \tag{6.92}
$$

Then, from (6.74) we get

$$
\Omega(x^+, \tau^+, y^+) - \Omega(x, \tau, y) \leq (-\alpha_2 + (\bar{\xi}_4 + \bar{\xi}_2^2)\alpha_2^2)\|U^\top \psi^c + \beta r^0\|^2_{(U^\top \mathcal{H} U)^{-1}}. \tag{6.93}
$$

If we choose $\alpha_2 \leq \frac{1}{2(\bar{\xi}_4 + \bar{\xi}_2^2)}$, then for the RHS we have

$$
\leq -\frac{1}{2}\alpha_2\|U^\top \psi^c + \beta r^0\|^2_{(U^\top \mathcal{H} U)^{-1}} \leq -\frac{\alpha_2}{32\bar{\xi}_1^2}\epsilon, \tag{6.94}
$$

where we used the bound for $\|U^\top \psi^c + \beta r^0\|^2_{(U^\top \mathcal{H} U)^{-1}}$ by Lemma 6.2.6.

□

Before giving the main theorem about the complexity analysis, aside from the above proof, we give a different approach for finding a lower bound on $\|U^\top \psi^c + \frac{1}{\mu} r^0\|^2_{(U^\top \mathcal{H} U)^{-1}}$ when we are away from the central path. This lemma is interesting because we use ideas from quasi-Newton low rank updates. As we explain in the conclusion chapter, these ideas are interesting for our approach.

We can verify that

$$U^\top \underbrace{\begin{bmatrix} y \\ y_\tau^0 + \tau\langle c, x\rangle \\ \tau A x + z^0 \\ \tau \end{bmatrix}}_{=:f} = -r^0, \quad \forall (x,\tau,y) \in Q_{DD}. \tag{6.95}$$

Hence, $\|U^\top \psi^c + \frac{1}{\mu} r^0\|^2_{(U^\top \mathcal{H} U)^{-1}} = \|U^\top (\psi^c - \frac{1}{\mu} f)\|^2_{(U^\top \mathcal{H} U)^{-1}}$. For simplicity, Let us define

$$\psi^c - \frac{1}{\mu} f := \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad f_1, f_2 \in \mathbb{R}^{m+1}. \tag{6.96}$$

Lemma 6.2.7. *Consider $(x,\tau,y) \in Q_{DD}$ and $0 < \delta < 1$ such that $\left\|\frac{\tau y}{\mu} - \Phi'\right\|_{[\Phi'']^{-1}} = \delta$. Then, there exists $\bar{H}^a$ satisfying*

$$\left(-1 + \left[\frac{(1-\delta)^2}{(1+\delta)^2} + (1-\delta)\right]\right) \bar{H}(x,\tau) \preceq \bar{H}^a \preceq \left(-1 + \left[\frac{(1+\delta)^2}{(1-\delta)^2} + (1+\delta)\right] \frac{\xi+3}{\xi-1}\right) \bar{H}(x,\tau),$$

such that for f_1 and f_2 defined in (6.96) we have

$$\frac{1}{\mu} f_1 = \bar{H}^a f_2. \tag{6.97}$$

Proof. We are using ideas from BFGS and DFP updates [57]. Let us consider vectors $\bar{f}_1 := \bar{H}^{-1/2} \frac{1}{\mu} f_1$ and $\bar{f}_2 := \bar{H}^{1/2} f_2$. We can verify that

$$\langle \bar{f}_1, \bar{f}_2 \rangle = \frac{1}{\mu} \langle f_1, f_2 \rangle = \frac{1}{\mu^2} \langle \Phi' - \frac{\tau y}{\mu}, Ax + \frac{1}{\tau} z^0 - \Phi'_* \rangle. \tag{6.98}$$

Note that $\Phi'(\Phi'_*) = \frac{\tau y}{\mu}$, and because Φ is a convex function, the term in (6.98) is non-negative. Actually, this term is positive and by the properties of s.c. barriers, we can bound (6.98) as

$$\frac{1}{\mu^2} \frac{\delta^2}{1+\delta} \le \langle \bar{f}_1, \bar{f}_2 \rangle \le \frac{1}{\mu^2} \frac{\delta^2}{1-\delta}. \tag{6.99}$$

Let us define

$$G := \left(I - \beta \bar{f}_1 \bar{f}_2^\top\right)\left(I - \beta \bar{f}_2 \bar{f}_1^\top\right) + \beta \bar{f}_1 \bar{f}_1^\top, \quad \beta := \frac{1}{\bar{f}_1^\top \bar{f}_2}. \tag{6.100}$$

Note that $G\bar{f}_2 = \bar{f}_1$ and if we can prove that G is close enough to identity matrix, we can prove our result. We want to bound the eigenvalues of G. For every vector $s \in \{\bar{f}_1^\perp\}$ we have $s^\top G s = s^\top s$. Therefore, we just need to check $\bar{f}_1^\top G \bar{f}_1$. We have

$$\frac{\bar{f}_1^\top G \bar{f}_1}{\bar{f}_1^\top \bar{f}_1} = -1 + \beta^2 (\bar{f}_1^\top \bar{f}_1)(\bar{f}_2^\top \bar{f}_2) + \beta(\bar{f}_1^\top \bar{f}_1). \tag{6.101}$$

We have already bounded $1/\beta = \bar{f}_1^\top \bar{f}_2$ in (6.99). For the other two terms, first note that we can directly verify:

$$\bar{f}_2^\top \bar{f}_2 = \frac{1}{\mu^2}\left\|Ax + \frac{1}{\tau} z^0 - \Phi'_*\right\|_{\Phi''}^2. \tag{6.102}$$

By using Lemma 4.2.2, we can bound $\bar{f}_2^\top \bar{f}_2$ as

$$\frac{1}{\mu^2}\frac{\delta^2}{(1+\delta)^2} \leq \bar{f}_2^\top \bar{f}_2 \leq \frac{1}{\mu^2}\frac{\delta^2}{(1-\delta)^2}. \tag{6.103}$$

We also have

$$\bar{f}_1^\top \bar{f}_1 = \frac{1}{\mu^2}\left\|\frac{\tau y}{\mu} - \Phi'\right\|_{[\Phi'']^{-1}}^2 + \frac{1}{\mu^2}\frac{\left[\langle \Phi' - \frac{\tau y}{\mu}, [\Phi'']^{-1}\Phi'\rangle + \langle \frac{\tau y}{\mu}, Ax + \frac{1}{\tau}z^0 - \Phi'_*\rangle\right]^2}{\xi\vartheta - \langle \Phi', [\Phi'']^{-1}\Phi'\rangle}. \tag{6.104}$$

We can bound $\bar{f}_1^\top \bar{f}_1$ as

$$\frac{1}{\mu^2}\delta^2 \leq \bar{f}_1^\top \bar{f}_1 \leq \frac{1}{\mu^2}\left(1 + \frac{4}{\xi - 1}\right)\delta^2. \tag{6.105}$$

If we define

$$\bar{H}^a := \bar{H}^{1/2} G \bar{H}^{1/2}, \tag{6.106}$$

then, using the above bounds, $\bar{H}^a$ satisfies the claimed relations. □

We proved that there exists ϵ_1 and ϵ_2 such that $\epsilon_1 \bar{H} \preceq \bar{H}^a \preceq \epsilon_2 \bar{H}$. Similar to the discussion after Lemma 6.2.2, because $\frac{1}{\mu} f_1 = \bar{H}^a f_2$, we have

$$\begin{aligned}
\max\{\epsilon_2, 1/\epsilon_1\} \left\| U^\top (\psi^c - \frac{1}{\mu} f) \right\|^2_{(U^\top \mathcal{H}(\bar{H}, \mu^2 \bar{H}) U)^{-1}} &\geq \left\| U^\top (\psi^c - \frac{1}{\mu} f) \right\|^2_{(U^\top \mathcal{H}(\bar{H}^a, \mu^2 \bar{H}^a) U)^{-1}} \\
&= \frac{1}{2} \left\| \psi^c - \frac{1}{\mu} f \right\|^2_{(\mathcal{H}(\bar{H}^a, \mu^2 \bar{H}^a))^{-1}} \\
&\geq \min\{\epsilon_1, 1/\epsilon_2\} \frac{1}{2} \left\| \psi^c - \frac{1}{\mu} f \right\|^2_{(\mathcal{H}(\bar{H}, \mu^2 \bar{H}))^{-1}} \\
&\geq \min\{\epsilon_1, 1/\epsilon_2\} \delta, \qquad (6.107)
\end{aligned}$$

where $\left\| \frac{\tau y}{\mu} - \Phi' \right\|_{[\Phi'']^{-1}} = \delta$.

6.3 Complexity of following the path to $\mu = +\infty$

We have analyzed the predictor and corrector search directions in Section 6.2. Now we can modify the statement of our predictor-corrector algorithm to one that provably follows the path in polynomial time.

Polynomial-time Predictor-Corrector Algorithm (PtPCA)

1. Choose constants $0 < 4\delta_1 < \delta_2 \leq \frac{1}{100\left((\bar{\xi}_2 \bar{\xi}_1)^3 + \bar{\xi}_3 \bar{\xi}_1^3\right)^2}$, where $\bar{\xi}_1$, $\bar{\xi}_2$, and $\bar{\xi}_3$ are functions of ξ defined in (6.70) and (6.76). Choose $z^0 \in \text{int} D$ and set $y^0 := \Phi'(z^0)$. Set $x^0 := 0$, $\tau^0 := 1$, and $k = 0$.

   Repeat until the stopping criteria are met:

2. If $\Omega(x^k, \tau^k, y^k) > \delta_1$, calculate the corrector search direction (d_x, d_τ, d_y) by (6.19) and (6.21), and choose α_2 as in (6.86) and $\alpha_1 := \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$. Apply the update in (6.3) to get $(x^{k+1}, \tau^{k+1}, y^{k+1})$.

3. If $\Omega(x^k, \tau^k, y^k) \leq \delta_1$, calculate the predictor search direction (d_x, d_τ, d_y) by (6.9) and (6.10), and choose $\alpha_2 = \frac{\kappa_1}{\sqrt{\vartheta}} \mu$ for κ_1 defined in the proof of Proposition 6.2.2, and $\alpha_1 := \frac{\alpha_2}{\tau + \alpha_2 d_\tau}$. Apply the update in (6.3) to get $(x^{k+1}, \tau^{k+1}, y^{k+1})$.

4. Set $k := k + 1$ and continue.

Our analysis of the predictor and corrector steps implies the following theorem:

Theorem 6.3.1. *For the polynomial-time predictor-corrector algorithm, there exists a positive constant κ_2 depending on ξ such that after N iterations, we have*

$$\mu \geq \exp\left(\frac{\kappa_2}{\sqrt{\vartheta}} N\right). \tag{6.108}$$

Proof. By Proposition 6.2.3, after each predictor step, we have to do at most

$$64(\bar{\xi}_4 + \bar{\xi}_2^2)\bar{\xi}_1^2(\delta_2 - \delta_1),$$

number of corrector steps to satisfy $\Omega(x, \tau, y) \leq \delta_1$. Also, by Proposition 6.2.2, after $\bar{N}$ cycles of predictor-corrector steps, we have

$$\mu \geq \left(1 + \frac{\kappa_1}{\sqrt{\vartheta}}\right)^{\bar{N}}.$$

Therefore, we have (6.108) for $\kappa_2 = O(1)\kappa_1$. □

Some of the consequences of Theorem 6.3.1 are discussed in the next chapter. One immediate consequence is, if our problem and its dual both are strictly feasible, then in $O\left(\sqrt{\vartheta}\ln\left(\frac{\vartheta}{\epsilon}\right)\right)$ iterations, we obtain an ϵ-approximation for an ϵ-perturbation of the problem.

Chapter 7

Output analysis

In previous chapters, we defined our central path and designed algorithms to follow the path efficiently to increase parameter μ to $+\infty$. At this stage, we assume that we have points (x, τ, y) close to the central path for a large enough μ. We need to interpret such points to clarify the status of the problem as accurately as possible. In view of our discussion in Chapter 3, we classify different infeasibility and unboundedness patterns that can happen for problem (5.1), and discuss the cases our algorithm can detect plus their complexity analysis.

7.1 Categorizing problem statuses

Let us first define the following four parameters that are the measurements of primal and dual feasibility:

Definition 7.1.1. *Define $t_p(z^0)$, $t_d(y^0)$, σ_p, and σ_d:*

$$\begin{aligned} t_p(z^0) &:= \sup\{t \geq 1 : \exists x \in \mathbb{E} \ s.t. \ Ax + \frac{1}{t} z^0 \in D\}, \\ t_d(y^0) &:= \sup\{t \geq 1 : \exists y \in D_* \ s.t. \ A^\top y - A^\top y^0 = -(t-1)c\}, \\ \sigma_p &:= \text{dist}(\text{range}(A), D), \\ \sigma_d &:= \text{dist}(\{y : A^\top y = -c\}, D_*), \end{aligned} \tag{7.1}$$

where dist$(\cdot,\cdot)$ returns the smallest distance between two convex sets. We denote σ_p by the measure of primal infeasibility, *and σ_d by the* measure of dual infeasibility.

Remark 7.1.1. *Note that all the above measures are scale dependent. For example, $t_p(z^0)$ attains different values when we change z^0 with respect to the boundary of the set D.*

The following lemma connects the parameters defined in Definition 7.1.1.

Lemma 7.1.1. *(a) Consider the definition of $t_p(z^0)$ and σ_p in Definition 7.1.1 and assume that $\sigma_p > 0$, then*

$$t_p(z^0) \le \frac{\|z^0\|}{\sigma_p}. \tag{7.2}$$

(b) Consider the definition of $t_d(y^0)$ and σ_d in Definition 7.1.1 and assume that $\sigma_d > 0$, then

$$t_d(y^0) \le \frac{\|y^0\|}{\sigma_d}. \tag{7.3}$$

Proof. By definition of σ_p, for every $Ax + \frac{1}{t}z^0 \in D$ we have

$$\sigma_p \le \left\| Ax + \frac{1}{t}z^0 - Ax \right\| = \frac{\|z^0\|}{t},$$

which gives us the result we want. The proof for the second inequality is analogous. □

Here is our classification of the status of problem (5.1):

1. **Solvability:** Problem (5.1) is called *solvable* if there exist a point x such that $Ax \in D$ and a point $y \in D_*$ such that $A^\top y = -c$, with duality gap equal to zero:

$$\langle c, x\rangle + \mathcal{S}(y) = 0, \tag{7.4}$$

   where $\mathcal{S}$ is the support function of D defined in (5.6). Problem is called *strictly primal-dual feasible* if there exist x such that $Ax \in \text{int}D$ and $y \in \text{int}D_*$ such that $A^\top y = -c$. Note that strict feasibility implies solvability. Also note that for both cases we have $t_p(z^0) = +\infty$ and $t_d(y^0) = +\infty$.

2. **Primal infeasibility:** problem (5.1) is called *primal infeasible* if there does not exist $Ax \in D$. It is called *strongly primal infeasible* if

$$\sigma_p = \text{dist}(\text{range}(A), D) > 0. \tag{7.5}$$

   The problem is *strictly primal infeasible* if there exists $\epsilon > 0$ such that every ϵ-perturbation of the problem is strongly primal infeasible.

3. **Unboundedness:** problem (5.1) is called *strongly primal unbounded* if for every $K > 0$, there exists $Ax \in D$ such that $\langle c, x\rangle < -K$. It is called *strictly primal unbounded* if there exists $Ax \in D$ and $Ah \in \text{int}(\text{rec}(D))$ such that $\langle c, h\rangle < 0$. Note that strict unboundedness implies strong unboundedness.

4. **ill-conditioning:** We divide this into two categories:
   (a) both primal and dual are infeasible, i.e., $\sigma_p > 0$ and $\sigma_d > 0$.
   (b) both primal and dual are (approximately) feasible, i.e., $\sigma_p = \sigma_d = 0$, but there does not exist a primal-dual feasible pair with zero duality gap.

The following lemma gives some equivalent properties for the infeasible cases we defined above that are easier to analyze:

Lemma 7.1.2. *Our problem is strongly primal infeasible if and only if there exists $y \in D_*$ such that*

$$A^\top y = 0, \quad \mathcal{S}(y) = -1. \tag{7.6}$$

The problem is strictly primal infeasible if and only if we can choose $y \in \text{int}D_*$ that satisfies (7.6).

Proof. First assume that there exists $y \in D_*$ that satisfies (7.6). Consider two sequences $\{z^k\} \in D$ and $\{Ax^k\}$ such that

$$\lim_k \|z^k - Ax^k\| = \text{dist}(D, \text{range}(A)).$$

We also have

$$\langle y, z^k - Ax^k\rangle = \langle y, z^k\rangle \le \mathcal{S}(y) = -1, \quad \forall k.$$

This implies that $\text{dist}(D, \text{range}(A)) > 0$. For the other direction, assume that $\text{dist}(D, \text{range}(A)) > 0$. Then, by separation theorem, there exist a vector y and $\beta \in \mathbb{R}$ such that

$$\begin{aligned} &\langle y, z\rangle > \beta, \quad \forall z \in \text{range}(A), \\ &\langle y, z\rangle < \beta, \quad \forall z \in D. \end{aligned}$$

The first relation holds only if $A^\top y = 0$, and if we substitute $z = 0$ in it, we get $\beta < 0$. The second relation holds only if $y \in D_*$ by the definition of D_* in (5.2), and since $\beta < 0$, we have $\mathcal{S}(y) < 0$. By some scaling we can assume that $\mathcal{S}(y) = -1$.

The second part of the proof holds by some modification to the above argument.

$\square$

Let us see an example to elaborate more on ill-conditioned cases.

Example 7.1.1. *Consider a problem in the Domain-Driven setup with $D \subset \mathbb{R}^2$ shown in Figure 7.1. Also assume that $A = I_{2\times 2}$ is the identity matrix. Note that the recession cone of D is a ray, which implies that D_* defined in (5.2) is $\{(y_1, y_2) \in \mathbb{R}^2 : y_1 \leq 0\}$ as shown in Figure 7.1. Let us define $c := [0\ 1]^\top$; then, the optimal objective value is 0 that is not attained. The system $A^\top y = -c$ has a unique solution $\bar{y} := [0\ -1]^\top$ that is on the boundary of D_*. It is also clear from the figure that $\mathcal{S}(\bar{y}) \leq 0$. Therefore, both primal and dual problems are feasible; however, we do not have a pair of primal-dual points with zero duality gap.*

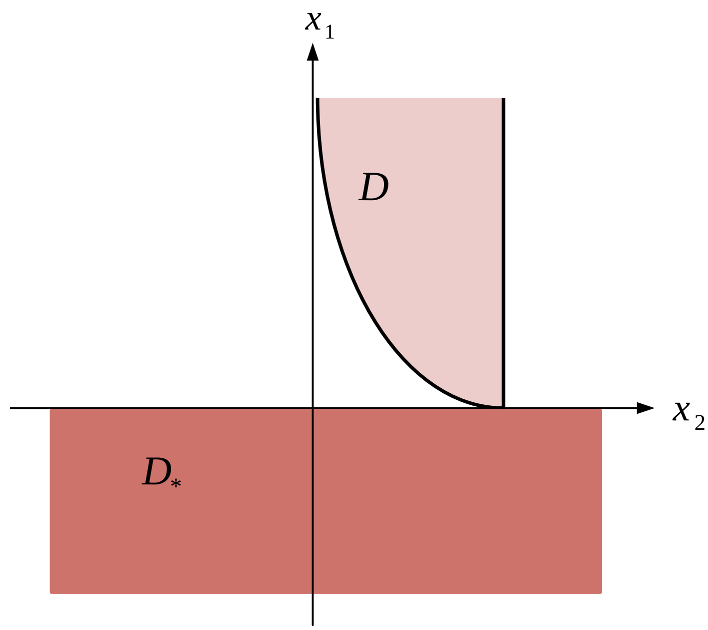

Figure 7.1: An example of a problem (5.1) with $D \subset \mathbb{R}^2$.

Now assume that we change A to $A = [1\ 0]^\top$; $\text{range}(A) = (\mathbb{R}, 0)$*. It is clear from the picture that the feasible region $Ax \in D$ is empty. If we choose $c = -1$, then the system $A^\top y = y_1 = -c$ does not have a solution in D_*. This implies that both primal and dual are infeasible. However, the measure of primal feasibility σ_p we defined in (7.1) is zero and we have approximately feasible points with arbitrarily small objective values. If we shift D to the right, D_* does not change, but we can make σ_p arbitrary large.*

7.2 Solvable cases

To better understand our results, we compare them with Nesterov-Todd-Ye's result in [56]. In Appendix A, we review the infeasible-start approach for the conic formulation proposed in [48, 56]. Assume that in the conic setup, both primal and dual problems are strictly feasible, and let $\hat{\bar{z}} \in \text{int}K$ and $\hat{\bar{s}} \in \text{int}K_*$ such that $\hat{\bar{s}} = -(\Phi^+)'(\hat{\bar{z}})$ and $\langle \hat{\bar{s}}, \hat{\bar{z}} \rangle = \bar{\vartheta}$, where Φ^+ is a $\bar{\vartheta}$-LH s.c. barrier defined on K. Then, the conic *feasibility measure* ρ_f is defined in [56] as

$$\rho_f := \max \left\{ \alpha : \quad \hat{\bar{z}} - \alpha \bar{z}^0 \in K, \quad \hat{\bar{s}} - \alpha \bar{s}^0 \in K_* \right\}. \tag{7.7}$$

For the conic infeasible-start setups in [56] and [48], Theorem 9 in [56] shows that for all the points in Q, defined in (A.10), we have

$$\bar{\tau} \geq \frac{\bar{\vartheta}+1}{\bar{\vartheta}+\rho_f} \rho_f \mu - \frac{1-\rho_f}{\rho_f}, \tag{7.8}$$

and Theorem 5 in [48] shows that for all the points in Q, we have

$$\bar{\tau} \geq \rho_f \mu - \frac{1}{\rho_f}. \tag{7.9}$$

These inequalities are important as they show how $\bar{\tau}$ grows directly with the increase of μ. [56] also considers the case that the problem is solvable. Assuming that $(\bar{z}^*, \bar{s}^*)$ is an arbitrary solution, Theorem 10 in [56] shows that

$$\bar{\tau} \geq \frac{\omega \mu}{\langle \bar{s}^0, \bar{z}^* \rangle + \langle \bar{s}^*, \bar{z}^0 \rangle + 1}, \tag{7.10}$$

where ω is an absolute constant parametrizing closeness to the central path. Here, we consider both the general solvable case and the strict primal-dual feasibility for the Domain-Driven setup; first for the general case of a solvable problem:

Lemma 7.2.1. *Assume that there exist a point $\bar{x}$ such that $A\bar{x} \in D$ and a point $\bar{y} \in D_*$ such that $A^\top \bar{y} = -c$, with duality gap equal to zero, i.e., $\langle c, \bar{x} \rangle + \mathcal{S}(\bar{y}) = 0$. If a point $(x, \tau, y) \in Q_{DD}$ is close to the central path in the sense of (5.30), then, for variable τ we have*

$$\frac{(\xi-1)\mu\vartheta - \mu\kappa\sqrt{\vartheta}}{\tau} \leq \xi\vartheta + \langle y^0 - \bar{y}, z^0 - A\bar{x} \rangle. \tag{7.11}$$

Proof. By substituting $\langle y, A\hat{x}\rangle \le \mathcal{S}(y)$ in (5.31), we get

$$\frac{(\xi-1)\mu\vartheta - \mu\kappa\sqrt{\vartheta}}{\tau} \le -y^0_\tau - \tau\langle c, x\rangle - \langle y, A\hat{x}\rangle. \tag{7.12}$$

Also note that from $\langle c, \bar{x}\rangle + \mathcal{S}(\bar{y}) = 0$ we have

$$\langle \bar{y}, z\rangle \le -\langle c, \bar{x}\rangle, \quad \forall z \in D. \tag{7.13}$$

We have

$$\begin{aligned} -\langle c, x\rangle &= \langle \bar{y}, Ax\rangle = \langle \bar{y}, Ax + \frac{1}{\tau}z^0\rangle - \langle \bar{y}, \frac{1}{\tau}z^0\rangle \\ &\le -\langle c, \bar{x}\rangle - \langle \bar{y}, \frac{1}{\tau}z^0\rangle, \quad \text{using (7.13)}, \end{aligned} \tag{7.14}$$

and also using $A^\top y = A^\top y^0 - (\tau - 1)c$ we can easily get

$$\langle y, A\bar{x}\rangle = \langle A^\top y^0 - (\tau-1)c, \bar{x}\rangle. \tag{7.15}$$

Substituting (7.14) and (7.15) in (7.12), we have

$$\begin{aligned} \frac{(\xi-1)\mu\vartheta - \mu\kappa\sqrt{\vartheta}}{\tau} &\le -y^0_\tau - \langle c, \bar{x}\rangle - \langle \bar{y}, z^0\rangle - \langle y^0, A\bar{x}\rangle \\ &= \xi\vartheta + \langle y^0, z^0\rangle - \langle c, \bar{x}\rangle - \langle \bar{y}, z^0\rangle - \langle y^0, A\bar{x}\rangle, \end{aligned} \tag{7.16}$$

where the last equation is by substituting $y^0_\tau = -\langle y^0, z^0\rangle - \xi\vartheta$ from (5.20). If we replace $c = -A^\top \bar{y}$, we get (7.11). $\square$

Let us rewrite (7.11) as

$$\mu \le \left[\frac{\xi\vartheta + \langle y^0 - \bar{y}, z^0 - A\bar{x}\rangle}{(\xi-1)\vartheta - \kappa\sqrt{\vartheta}}\right]\tau, \tag{7.17}$$

which shows the rate of increase for τ with respect to μ. We can do more for the case that both primal and dual are strictly feasible. Let us define

$$\begin{aligned} \hat{x} &:= \operatorname{argmin}_x\{\Phi(Ax) + \langle c, x\rangle\}, \\ \hat{y} &:= \Phi'(A\hat{x}), \\ \hat{y}_\tau &:= -\xi\vartheta - \langle \hat{y}, A\hat{x}\rangle. \end{aligned} \tag{7.18}$$

By using the first order optimality condition, we have $A^\top \hat{y} = -c$. Now we define the *feasibility measure* as

$$\sigma_f := \sup\left\{\alpha : \hat{y} - \alpha y^0 \in D_*, \quad \frac{A\hat{x} - \alpha z^0}{1-\alpha} \in D, \quad \alpha < 1, \quad \mathcal{S}(\hat{y} - \alpha y^0) + \hat{y}_\tau - \alpha y^0_\tau \leq 0\right\}.$$

Note that by using Theorem 4.2.1 and the fact that $\hat{y} := \Phi'(A\hat{x})$, we have

$$\mathcal{S}(\hat{y}) + \hat{y}_\tau \leq \vartheta + \langle \hat{y}, A\hat{x}\rangle - \xi\vartheta - \langle \hat{y}, A\hat{x}\rangle = -(\xi - 1)\vartheta < 0.$$

Hence, $\sigma_f > 0$. The following theorem gives a result similar to (7.8) and (7.9) for the Domain-Driven setup.

Theorem 7.2.1. *Assume that both primal and dual are strictly feasible and $(x, \tau, y) \in Q_{DD}$ and we have the additional property that $\mathcal{S}(y) + y_\tau \leq 0$, where $y_\tau = y^0_\tau + \tau\langle c, x\rangle$. Then,*

$$\tau - 1 \geq \sigma_f \mu - \frac{1}{\sigma_f}. \tag{7.19}$$

Proof. By definition of σ_f we have

$$\langle \hat{y} - \alpha y^0, \frac{A\hat{x} - \alpha z^0}{1-\alpha}\rangle + \hat{y}_\tau - \alpha y^0_\tau \ \leq \ \mathcal{S}(\hat{y} - \alpha y^0) + \hat{y}_\tau - \alpha y^0_\tau \ \leq 0.$$

Multiplying both sides with $(1-\alpha)$ and reordering the terms give us (for $\alpha = \sigma_f$)

$$\langle \hat{y}, A\hat{x}\rangle + \hat{y}_\tau - \sigma_f\left(\langle \hat{y}, z^0\rangle + \hat{y}_\tau + \langle y^0, A\hat{x}\rangle + y^0_\tau\right) + \sigma_f^2\left(\langle y^0, z^0\rangle + y^0_\tau\right) \leq 0.$$

By (7.18) we have $\langle \hat{y}, A\hat{x}\rangle + \hat{y}_\tau = -\xi\vartheta$ and by (5.20) we have $\langle y^0, z^0\rangle + y^0_\tau = -\xi\vartheta$. Substituting these in the above inequality and dividing both sides by σ_f we get

$$-\left(\langle \hat{y}, z^0\rangle + \hat{y}_\tau + \langle y^0, A\hat{x}\rangle + y^0_\tau\right) \leq \xi\vartheta\left(\frac{1}{\sigma_f} + \sigma_f\right). \tag{7.20}$$

Another useful inequality is derived as follows:

$$\begin{aligned}
& -\langle y, A\hat{x}\rangle - y_\tau - \langle \hat{y}, \tau Ax + z^0\rangle - \tau\hat{y}_\tau \\
= \ & -\langle y, A\hat{x} - \sigma_f z^0\rangle - (1-\sigma_f)y_\tau - \langle \hat{y} - \sigma_f y^0, \tau Ax + z^0\rangle - \tau(\hat{y}_\tau - \sigma_f y^0_\tau) \\
& +\sigma_f\left(-\langle y, z^0\rangle - y_\tau - \langle y^0, \tau Ax + z^0\rangle - \tau y^0_\tau\right), \quad \text{by adding and subtracting some terms,} \\
\geq \ & \ \sigma_f\left(-\langle y, z^0\rangle - y_\tau - \langle y^0, \tau Ax + z^0\rangle - \tau y^0_\tau\right).
\end{aligned} \tag{7.21}$$

The last equation is the place we use $\mathcal{S}(y) + y_\tau \leq 0$ and the fact that $\frac{A\hat{x} - \sigma_f z^0}{1-\sigma_f} \in D$, which together imply that

$$-\langle y, A\hat{x} - \sigma_f z^0 \rangle - (1-\sigma_f) y_\tau \geq 0.$$

We also use the fact that

$$\langle \hat{y} - \sigma_f y^0, Ax + \frac{1}{\tau} z^0 \rangle + (\hat{y}_\tau - \sigma_f y^0_\tau) \ \leq\ \mathcal{S}(\hat{y} - \alpha y^0) + \hat{y}_\tau - \alpha y^0_\tau \ \leq 0.$$

We can expand the term in (7.21) in another way:

$$\begin{aligned}
& -\langle y, A\hat{x} \rangle - y_\tau - \langle \hat{y}, \tau Ax + z^0 \rangle - \tau \hat{y}_\tau \\
= \ & -\langle A^\top y^0 - (\tau-1)c, \hat{x} \rangle - y_\tau + \tau \langle c, x \rangle - \langle \hat{y}, z^0 \rangle - \tau \hat{y}_\tau \\
= \ & (\tau-1)(\langle c, \hat{x} \rangle - \hat{y}_\tau) - \big(\langle \hat{y}, z^0 \rangle + \hat{y}_\tau + \langle y^0, A\hat{x} \rangle + y^0_\tau \big) \\
= \ & (\tau-1)(\langle \hat{y}, A\hat{x} \rangle - \hat{y}_\tau) - \big(\langle \hat{y}, z^0 \rangle + \hat{y}_\tau + \langle y^0, A\hat{x} \rangle + y^0_\tau \big) \\
= \ & (\tau-1)\xi\vartheta - \big(\langle \hat{y}, z^0 \rangle + \hat{y}_\tau + \langle y^0, A\hat{x} \rangle + y^0_\tau \big), \quad \text{using (7.18).}
\end{aligned} \tag{7.22}$$

For the final piece, we find the relation between μ and the last term in (7.21)

$$\begin{aligned}
\xi\vartheta\mu \ &= \ -\tau y_\tau - \langle A^\top y^0 - (\tau-1)c, \tau x \rangle - \langle y, z^0 \rangle, \quad \text{using (5.22)}, \\
&= \ -\tau y_\tau + \tau(\tau-1)\langle c, x \rangle - \langle y^0, \tau Ax + z^0 \rangle + \langle y^0, z^0 \rangle - \langle y, z^0 \rangle \\
&= \ \big(-\langle y, z^0 \rangle - y_\tau - \langle y^0, \tau Ax + z^0 \rangle - \tau y^0_\tau \big) + y^0_\tau + \langle y^0, z^0 \rangle \\
&= \ \big(-\langle y, z^0 \rangle - y_\tau - \langle y^0, \tau Ax + z^0 \rangle - \tau y^0_\tau \big) - \xi\vartheta.
\end{aligned} \tag{7.23}$$

Combining (7.20), (7.21), (7.22), and (7.23) we get

$$\sigma_f(\mu+1)\xi\vartheta \leq (\tau-1)\xi\vartheta + \xi\vartheta \left(\frac{1}{\sigma_f} + \sigma_f \right). \tag{7.24}$$

By cancelling $\xi\vartheta$ from both sides and reordering, we get the result of the theorem. $\square$

Similar to Nesterov, Todd, and Ye [56], we define a point $(x, \tau, y) \in Q_{DD}$ an *ϵ-solution* of our problem if

$$\max \left\{ \frac{1}{\tau} \ , \ \frac{\xi\vartheta}{\tau^2}\mu \right\} \leq \epsilon. \tag{7.25}$$

Lemma 7.2.1 and Theorem 7.2.1 yield the following theorem about the solvable case and its stronger special case of strict primal-dual feasibility for detecting an ϵ-solution.

Theorem 7.2.2. *(a) Assume we have strict primal-dual feasibility for the Domain-Driven problem* (5.1). *Then, our path following algorithm returns an* ϵ*-solution in number of iterations bounded by*

$$O\left(\sqrt{\vartheta}\ln\left(\frac{\vartheta}{\sigma_f \epsilon}\right)\right).$$

(b) Assume that problem (5.1) *is solvable and, in view of Lemma 7.2.1, let*

$$K := \min\{\xi\vartheta + \langle y^0 - \bar{y}, z^0 - A\bar{x}\rangle : A\bar{x} \in D, \bar{y} \in D_*, A^\top \bar{y} = -c, \langle c, x\rangle + \mathcal{S}(\bar{y}) = 0\}.$$

Then, our path following algorithm returns an ϵ*-solution in number of iterations bounded by*

$$O\left(\sqrt{\vartheta}\ln\left(\frac{\vartheta K}{\epsilon}\right)\right).$$

7.3 Weak infeasibility and unboundedness detector

We call these detectors weak because they return approximate certificates for the problem. Assume that for a point $(x, \tau, y) \in Q_{DD}$ and an $\epsilon > 0$ we have

$$\frac{\tau^2}{\vartheta\mu} \leq \epsilon. \tag{7.26}$$

We are interested in the case that at least one of σ_p or σ_d defined in (7.1.1) is positive, and Lemma 7.1.1 implies that τ is bounded. Because τ is bounded and we have

$$A^\top \frac{\tau y}{\mu} = A^\top \frac{\tau y^0}{\mu} - \frac{\tau(\tau-1)c}{\mu},$$

$\frac{\tau y}{\mu}$ converges to a point in the null space of $A^\top$. If we can confirm that $\mathcal{S}\left(\frac{\tau y}{\mu}\right) < 0$, then we have an approximate certificate of infeasibility. On the other hand, if $\langle c, x\rangle$ becomes a very large negative number, then Ax dominates the other term in $Ax + \frac{1}{\tau}z^0$ and we have an approximate certificate of unboundedness. The reason is that for every vector y_c such that $A^\top y = -c$, we have $\|y_c\|\|Ax\| \geq |\langle c, x\rangle| \geq K$.

We say that (x, τ, y) is an ϵ*-certificate of infeasibility* if it satisfies (7.26) and

$$\mathcal{S}\left(\frac{\tau y}{\mu}\right) < -1. \tag{7.27}$$

We say that (x, τ, y) is an *ϵ-certificate of unboundedness* if it satisfies (7.26) and

$$\langle c, x \rangle < -\frac{1}{\epsilon}. \tag{7.28}$$

When we are close to the central path, by Lemma 5.3.1, we have

$$\langle c, x \rangle + \mathcal{S}\left(\frac{y}{\tau}\right) \leq \frac{-y_\tau^0}{\tau} - \left((\xi - 1) - \frac{\kappa}{\sqrt{\vartheta}}\right)\frac{\mu\vartheta}{\tau^2}. \tag{7.29}$$

Using this equation, we can prove the following theorem:

Theorem 7.3.1. *(weak detector) Assume that at least one of σ_p or σ_d defined in (7.1.1) is positive. Then, our path following algorithm returns either an ϵ-certificate of infeasibility or an ϵ-certificate of unboundedness in number of iterations bounded by*

$$O\left(\sqrt{\vartheta}\ln\left(\frac{1}{\vartheta\epsilon}\min\left\{\frac{\|z^0\|}{\sigma_p}, \frac{\|y^0\|}{\sigma_d}\right\}\right)\right). \tag{7.30}$$

Proof. By Lemma 7.1.1, we have a bound on τ. To satisfy (7.26), we must have $\mu \geq \frac{\tau^2}{\vartheta\epsilon}$. We can assume that $\left((\xi - 1) - \frac{\kappa}{\sqrt{\vartheta}}\right) > 0$, then, when $\frac{\mu\vartheta}{\tau^2}$ gets large enough, (7.29) implies that at least one of (7.27) or (7.28) happens. □

Let us see how the weak detector behaves in the infeasibility and unbounded cases we defined above.

7.3.1 Infeasibility

By definition, if there exists $\hat{y} \in D_*$ such that $A^\top \hat{y} = 0$ and $S(\hat{y}) = -1$, we have strong primal infeasibility, and it becomes strict if we can choose $y \in \text{int}D_*$. If we assume that the problem is not ill-conditioned, then, the dual problem is feasible, i.e., there exists $y_c \in D_*$ such that $A^\top y_c = -c$. The following lemma shows that this property holds for the strict primal infeasible case:

Lemma 7.3.1. *If there exists $\hat{y} \in \text{int}D_*$ such that $A^\top \hat{y} = 0$, then for any vector c, there exists $y_c \in \text{int}D_*$ such that $A^\top y_c = -c$.*

Proof. As A has full column rank, there exists a solution for $A^\top y = -c$. Then, for a large enough coefficient α, the point $y + \alpha\hat{y}$ is in $\text{int}D_*$, and we also have $A^\top(y + \alpha\hat{y}) = A^\top y = -c$. □

Such y_c lets us bound $-\langle c, x\rangle$. We have

$$-\langle c, x\rangle \leq \langle y_c, Ax\rangle = \langle y_c, Ax + \frac{1}{\tau}z^0\rangle - \langle y_c, \frac{z^0}{\tau}\rangle \leq \mathcal{S}(y_c) - \langle y_c, \frac{z^0}{\tau}\rangle. \tag{7.31}$$

When $y_c \in \mathrm{int}D_*$, we have $\mathcal{S}(y_c) \leq \vartheta + \langle y_c, \Phi'_*(y_c)\rangle$ by Theorem 4.2.1 that gives us an explicit bound. When $-\langle c, x\rangle$ is bounded, Theorem 7.3.1 implies that our weak detector can detect infeasibility in polynomial time. The next lemma is related to Theorem 7.3.1 and shows that, when $-\langle c, x\rangle$ is bounded, for every point $(x, \tau, y) \in Q_{DD}$ close to the central path with a large μ, we have $\mathcal{S}(y) < 0$.

Lemma 7.3.2. *Assume that a point $(x, \tau, y) \in Q_{DD}$ satisfies the assumptions of Lemma 5.3.1. If there exists a constant K such that*

$$\tau[-y_\tau^0 - \tau\langle c, x\rangle] \leq K,$$

and we have $\mu \geq \frac{K}{(\xi-1)\vartheta - \kappa\sqrt{\vartheta}}$, then,

$$\mathcal{S}\left(\frac{\tau y}{\mu}\right) < 0.$$

Proof. By the result of Lemma 5.3.1, if we multiply both sides of (5.31) by τ^2 and reorder the terms, we have

$$\mathcal{S}\left(\tau y\right) \leq \tau[-y_\tau^0 - \tau\langle c, x\rangle] - [(\xi - 1)\vartheta - \kappa\sqrt{\vartheta}]\mu. \tag{7.32}$$

The result of the lemma yields if we substitute the assumptions in (7.32). □

7.3.2 Unboundedness

Let us see the connection of unboundedness with σ_d.

Lemma 7.3.3. *Assume that problem (5.1) is strongly (or strictly) unbounded. Then, $\sigma_d > 0$, where σ_d is defined in Definition (7.1.1).*

Proof. The definition of unboundedness implies that there exists $Ah \in \mathrm{rec}(D)$ such that $\langle c, h\rangle < 0$. If $\sigma_d = 0$, there exists $\{y^k\} \in D_*$ such that $\lim_k \|A^\top y^k + c\| = 0$. By characterization of D_* in (5.2), we have

$$0 \geq \langle y^k, Ah\rangle = \langle A^\top y^k, h\rangle = \langle A^\top y^k + c, h\rangle - \langle c, h\rangle, \quad \forall k.$$

This gives a contradiction when k tends to $+\infty$. □

This lemma shows that if the problem is unbounded, τ is bounded by $\|y^0\|/\sigma_d$ in Lemma 7.1.1. Using just the fact that the problem is feasible, there exists $A\hat{x} \in D$ and we have

$$\mathcal{S}\left(\frac{\tau y}{\mu}\right) \geq \langle \frac{\tau y}{\mu}, A\hat{x}\rangle = \frac{\tau}{\mu}\langle A^\top y, \hat{x}\rangle = \frac{\tau\langle A^\top y^0 - (\tau-1)c, \hat{x}\rangle}{\mu}. \tag{7.33}$$

The above discussion shows that for the points $(x, \tau, y) \in Q_{DD}$ that μ satisfies

$$2\mu \geq \underbrace{\frac{\|y^0\|}{\sigma_d}|\langle A^\top y^0, \hat{x}\rangle| + \frac{\|y^0\|^2}{\sigma_d^2}|\langle c, \hat{x}\rangle|}_{=:K_{unb,1}}, \tag{7.34}$$

we have $\mathcal{S}\left(\frac{\tau y}{\mu}\right) \geq -\frac{1}{2}$. Therefore, by Theorem 7.3.1, our weak detector returns an ϵ-certificate of unboundedness in polynomial time. Considering the argument of Theorem 7.3.1 and (7.34), we have that for every $\epsilon > 0$, after at most

$$O\left(\sqrt{\vartheta}\ln\left(\frac{K_{unb,1}}{\vartheta} + \frac{1}{\vartheta\epsilon}\frac{\|y^0\|^2}{\sigma_d^2}\right)\right) \tag{7.35}$$

iterations we have $\langle c, x\rangle \leq -\frac{1}{\epsilon}$.

7.4 Strict infeasibility and unboundedness detector

In the previous section, we saw a weak detector for infeasibility and unboundedness. In this section, we show that in the case of strict infeasibility and unboundedness, we can actually find a certificate for the exact problem in polynomial time. The idea is that we need to project our current point onto a proper set using a suitable norm.

7.4.1 Infeasibility

By definition, if there exists $\hat{y} \in \text{int}D_*$ such that $A^\top\hat{y} = 0$ and $S(\hat{y}) = -1$, we have strict primal infeasibility. To get the exact certificate, we show how to project y onto $\text{null}(A^\top)$ with respect to a suitable norm. Let us first show that the points of the form $Ax + \frac{1}{\tau}z^0$ are bounded.

Lemma 7.4.1. *Assume that there is a point* $y \in \text{int}D_*$ *such that* $A^\top y = 0$*. Then, there exists a point* $\hat{u} := A\hat{x} + \frac{1}{\hat{\tau}}z^0 \in \text{int}D$ *such that for every* $\frac{\xi}{\xi+1} \leq \tau \leq t_p(z^0)$ *and every* x *such that* $Ax + \frac{1}{\tau}z^0 \in D$*, we have*

$$\left\| Ax + \frac{1}{\tau}z^0 - \hat{u} \right\|_{\Phi''(\hat{u})} \leq \vartheta + 2\sqrt{\vartheta}. \tag{7.36}$$

Proof. Let us for simplicity define $\gamma_\xi := \frac{\xi+1}{\xi}$. First assume that $Ax + \gamma_\xi z^0 \in \text{int}D$ is feasible. Then, because there exists a point $y \in \text{int}D_*$ such that $A^\top y = 0$, $\Phi(Ax + \gamma_\xi z^0)$ has a minimizer that we denote as x^1 and we have $A^\top \Phi'(Ax^1 + \gamma_\xi z^0) = 0$. If we have $\langle \Phi'(Ax^1 + \gamma_\xi z^0), z^0 \rangle \leq 0$, then, for every $\frac{\xi}{\xi+1} \leq \tau \leq t_p(z^0)$ and every x such that $Ax + \frac{1}{\tau}z^0 \in D$ we have

$$\langle \Phi'\left(Ax^1 + \gamma_\xi z^0\right), Ax + \frac{1}{\tau}z^0 - \left(Ax^1 + \gamma_\xi z^0\right)\rangle = \langle \Phi'(Ax^1 + z^0), \frac{1}{\tau}z^0 - \frac{\xi+1}{\xi}z^0 \rangle \geq 0.$$

Therefore, by using property (4.29) of s.c. barriers, (7.36) holds for $\hat{u} := Ax^1 + \gamma_\xi z^0$. If otherwise $\langle \Phi'(Ax^1 + \gamma_\xi z^0), z^0 \rangle > 0$, because, by strict infeasibility, there exists a point $\hat{y} \in \text{int}D_*$ such that $A^\top \hat{y} = 0$ and $\langle \hat{y}, z^0 \rangle \leq \mathcal{S}(\hat{y}) < 0$, by convexity, there exists a point $y \in \text{int}D_*$ such that $A^\top y = 0$ and $\langle y, z^0 \rangle = 0$. This implies that the function $\Phi(Ax + \gamma z^0)$, which is a s.c. barrier in terms of (x, γ), has a minimizer that we denote by $Ax^2 + \gamma^2 z^0$. Because this point in a minimizer, for every $1 \leq \tau \leq t_p(z^0)$ and every x such that $Ax + \frac{1}{\tau}z^0 \in D$ we have

$$\langle \Phi'(Ax^2 + \gamma^2 z^0), Ax + \frac{1}{\tau}z^0 - (Ax^2 + \gamma^2 z^0) \rangle \geq 0, \tag{7.37}$$

and by the same argument $\hat{u} = Ax^2 + \gamma^2 z^0$ satisfies (7.36).

Now assume that $Ax + \gamma_\xi z^0 \in \text{int}D$ is not feasible. We claim that $\Phi(Ax + \gamma z^0)$ again has a minimizer by showing that the set of points $Ax + \gamma z^0 \in D$ is bounded. Suppose otherwise. Then, because $\frac{1}{t_p(z^0)} \leq \gamma \leq \gamma_\xi$, the set D has a nonzero recessive direction Ah. Consider a point $z \in \text{int}D$ such that $A^\top \Phi'(z) = 0$. Then, by a property of s.c. barriers (see for example [45]-Corollary 3.2.1), we have

$$0 = \langle \Phi'(z), Ah \rangle \geq \sqrt{\langle Ah, \Phi''(z)Ah \rangle} \quad \Rightarrow \quad Ah = 0. \tag{7.38}$$

This is a contradiction and so the set of points $Ax + \gamma z^0 \in D$ is bounded. By property SCB-3 in subsection (4.2.2), $\Phi(Ax + \gamma z^0)$ has a minimizer and we can repeat the same argument as above. $\square$

For the main proof of this section, we define a series of points that get close to the points on the central path for large enough μ. Consider the following optimization problem for $\tau \geq 1$.

$$\begin{array}{rrcl} \min & \Phi_*(y) & & \\ & A^\top y & = & 0 \\ & \langle y, z^0 \rangle & = & -\tau\xi\vartheta. \end{array} \tag{7.39}$$

Let us denote the solution of this problem by $\bar{y}(\tau)$. If we write the optimality conditions for $\bar{y}(\tau)$, we have $\Phi'_*(\bar{y}(\tau)) = A\bar{x}(\tau) + \frac{1}{t(\tau)}z^0$, for some $\bar{x}(\tau)$ and $t(\tau)$. We claim that $t(\tau) \geq \frac{\xi}{\xi+1}$. By Theorem 4.2.1, we have

$$\begin{aligned} \langle \bar{y}(\tau), z^0 \rangle - \vartheta \leq \langle \bar{y}(\tau), \Phi'_*(\bar{y}(\tau)) \rangle = \langle \bar{y}(\tau), A\bar{x}(\tau) + \frac{1}{t(\tau)}z^0 \rangle = \frac{1}{t(\tau)}\langle \bar{y}(\tau), z^0 \rangle \\ \Rightarrow \quad -\tau\xi\vartheta - \vartheta \leq \frac{-1}{t(\tau)}\tau\xi\vartheta \quad \Rightarrow \quad \frac{1}{t(\tau)} \leq \frac{\tau\xi + 1}{\tau\xi} \leq \frac{\xi+1}{\xi}. \end{aligned} \tag{7.40}$$

This implies that all the points $A\bar{x}(\tau) + \frac{1}{t(\tau)}z^0$ are bounded using Lemma 7.4.1. Now we are ready to prove the following lemma that shows $\bar{y}(\tau)$ gets very close to $\frac{\tau}{\mu}y$ in local norm when μ is large enough.

Lemma 7.4.2. *Consider a point (x, τ, y) on the central path and $\bar{y}(\tau)$ as the solution of (7.39). Then, there exists a constant K_3 depending on the initial points such that*

$$\frac{\left\|\bar{y}(\tau) - \frac{\tau}{\mu}y\right\|^2_{\Phi''_*(\frac{\tau}{\mu}y)}}{1 + \left\|\bar{y}(\tau) - \frac{\tau}{\mu}y\right\|_{\Phi''_*(\frac{\tau}{\mu}y)}} \leq \frac{K_3}{\mu}. \tag{7.41}$$

Proof. By using property (4.18) of s.c. functions, we have

$$\frac{\left\|\bar{y}(\tau) - \frac{\tau}{\mu}y\right\|^2_{\Phi''_*(\frac{\tau}{\mu}y)}}{1 + \left\|\bar{y}(\tau) - \frac{\tau}{\mu}y\right\|_{\Phi''_*(\frac{\tau}{\mu}y)}} \leq \langle \frac{\tau}{\mu}y - \bar{y}(\tau), Ax + \frac{1}{\tau}z^0 - A\bar{x}(\tau) - \frac{1}{t(\tau)}z^0 \rangle. \tag{7.42}$$

Because $\bar{y}(\tau)$ is the solution of (7.39), we have

$$\langle -\bar{y}(\tau), Ax + \frac{1}{\tau}z^0 - A\bar{x}(\tau) - \frac{1}{t(\tau)}z^0 \rangle = \left(\frac{1}{\tau} - \frac{1}{t(\tau)}\right)\tau\xi\vartheta. \tag{7.43}$$

We also have

$$\begin{aligned}\langle\frac{\tau}{\mu}y, Ax+\frac{1}{\tau}z^0-A\bar{x}(\tau)-\frac{1}{t(\tau)}z^0\rangle &= \frac{\tau}{\mu}\langle y, Ax-A\bar{x}(\tau)\rangle \\ &\quad+\left(\frac{1}{\tau}-\frac{1}{t(\tau)}\right)\langle\frac{\tau}{\mu}y, z^0\rangle. \end{aligned} \tag{7.44}$$

For the first term of (7.44), by using Lemma 7.4.1, we have

$$\begin{aligned}\frac{\tau}{\mu}\langle y, Ax-A\bar{x}(\tau)\rangle &= \frac{\tau}{\mu}\langle A^\top y, x-\bar{x}(\tau)\rangle = \frac{\tau}{\mu}\langle A^\top(y^0+(\tau-1)y_c), x-\bar{x}(\tau)\rangle \\ &= \frac{\tau}{\mu}\langle y^0+(\tau-1)y_c, Ax+\frac{1}{\tau}z^0-A\bar{x}(\tau)-\frac{1}{t(\tau)}z^0\rangle \\ &\quad -\frac{\tau}{\mu}\left(\frac{1}{\tau}-\frac{1}{t(\tau)}\right)\langle y^0+(\tau-1)y_c, z^0\rangle \\ &\le \frac{2\tau}{\mu}\|y^0+(\tau-1)y_c\|_{[\Phi''(\hat{u})]^{-1}}(\vartheta+2\sqrt{\vartheta})+\frac{\tau}{\mu}|\langle y^0+(\tau-1)y_c, z^0\rangle|. \end{aligned} \tag{7.45}$$

For the second term of (7.44), note that by definition of μ in (5.22), we have

$$\begin{aligned} &\frac{\xi\vartheta}{\tau}\mu = -y^0_\tau-\tau\langle c, x\rangle-\langle y, Ax+\frac{1}{\tau}z^0\rangle \\ \Rightarrow\quad &\frac{\tau}{\mu}\langle y, z^0\rangle = \frac{\tau^2(-y^0_\tau-\langle c, x\rangle-\langle y^0, Ax\rangle)}{\mu}-\tau\xi\vartheta \\ &\le \frac{\tau^2(|y^0_\tau|+\|y_c+y^0\|_{[\Phi''(\hat{u})]^{-1}}(\vartheta+2\sqrt{\vartheta}+\|\hat{u}\|_{\Phi''(\hat{u})}+\|z^0\|_{\Phi''(\hat{u})}))}{\mu}-\tau\xi\vartheta. \end{aligned} \tag{7.46}$$

An important fact here is that the last term in (7.46) cancels out (7.43). If we substitute (7.46) and (7.45) in (7.44) and substitute the result and (7.43) in (7.42), we get (7.41), where

$$\begin{aligned} K_3 &:= 2t_p(z^0)\|y^0+(\tau-1)y_c\|_{[\Phi''(\hat{u})]^{-1}}(\vartheta+2\sqrt{\vartheta})+t_p(z^0)|\langle y^0+(\tau-1)y_c, z^0\rangle| \\ &\quad +t_p^2(z^0)(|y^0_\tau|+\|y_c+y^0\|_{[\Phi''(\hat{u})]^{-1}}(\vartheta+2\sqrt{\vartheta}+\|\hat{u}\|_{\Phi''(\hat{u})}+\|z^0\|_{\Phi''(\hat{u})})). \end{aligned} \tag{7.47}$$

□

Now we can prove the following proposition for our strict detector:

Proposition 7.4.1. *(Strict primal infeasibility detector) Assume that there exists $\hat{y} \in \text{int}D_*$ such that $A^\top \hat{y} = 0$ and $\mathcal{S}(\hat{y}) = -1$. Our path-following algorithm returns a point $y \in D_*$ with $A^\top y = 0$ and $\mathcal{S}(y) \le -1$ in at most*

$$O\left(\sqrt{\vartheta} \ln K_3\right)$$

iterations.

Proof. By Lemma 7.4.2, if we make $\mu \ge 100K_3$, then we have $\|\bar{y}(\tau) - \frac{\tau}{\mu} y\|^2_{\Phi''_*(\frac{\tau}{\mu}y)} \le 0.1$. Now, if we project $\frac{\tau}{\mu} y$ by the norm defined by $\Phi''_*(\frac{\tau}{\mu}y)$ on the subspace $A^\top y = 0, \langle y, z^0 \rangle = -\tau\xi\vartheta$, the resulted point $\hat{y}$ must have a distance (in local norm) to $\bar{y}(\tau)$ smaller than 1 and so it lies in $\text{int}D_*$. We just need to show that $\mathcal{S}(\hat{y}) < 0$. Let (x, τ, y) be on the central path, then (let $\Phi'_* := \Phi'_*\left(\frac{\tau}{\mu} y\right)$)

$$\begin{aligned}
\langle \hat{y}, \Phi'_*(\hat{y}) \rangle &= \langle \hat{y}, \Phi'_*(\hat{y}) - \Phi'_* \rangle + \langle \hat{y}, \Phi'_* \rangle \\
&= \langle \hat{y}, \Phi'_*(\hat{y}) - \Phi'_* \rangle + \langle \hat{y}, Ax + \frac{1}{\tau} z^0 \rangle \\
&\le \|\hat{y}\|_{\Phi''_*(\hat{y})} \|\Phi'_*(\hat{y}) - \Phi'_*\|_{[\Phi''_*(\hat{y})]^{-1}} + \frac{1}{\tau} \langle \hat{y}, z^0 \rangle, \quad \text{using } A^\top \hat{y} = 0 \\
&\le \sqrt{\vartheta} \|\Phi'_*(\hat{y}) - \Phi'_*\|_{[\Phi''_*(\hat{y})]^{-1}} - \xi\vartheta, \quad \text{using } \|\hat{y}\|_{\Phi''_*(\hat{y})} \le \sqrt{\vartheta}.
\end{aligned} \tag{7.48}$$

Note that $\|\Phi'_*(\hat{y}) - \Phi'_*\|_{[\Phi''_*(\hat{y})]^{-1}}$ is smaller than 1 and so $\langle \hat{y}, \Phi'_*(\hat{y}) \rangle \le -\tilde{\xi}\vartheta$ for some $\tilde{\xi} > 1$. By Theorem 4.2.1 we have $\mathcal{S}(\hat{y}) < 0$ as we want. □

7.4.2 Unboundedness

Problem (5.1) is called *strictly primal unbounded* if $Ax \in D$ is feasible and there exists $Ah \in \text{int}(\text{rec}(D))$ such that $\langle c, h \rangle < 0$. Note that this definition implies that $Ax \in \text{int}D$ is also feasible. For the case of strict unboundedness, for every μ, we define $\bar{x}(K)$ as the unique solution of the following problem:

$$\begin{array}{ll}
\min & \frac{\mu}{\tau}\Phi(Ax) - \langle A^\top y^0, x \rangle \\
& \qquad\qquad \langle c, x \rangle \ \le \ -K,
\end{array} \tag{7.49}$$

and we also define $\bar{y}(K) := \frac{\mu}{\tau}\Phi'(A\bar{x}(K))$. By writing the optimality conditions, we have

$$A^\top \bar{y}(K) = A^\top y^0 - t(K)c,$$

for some proper $t(K) \geq 0$ that satisfies $t(K) \leq t_d(y^0)$. Similar to what we did for the infeasible case, we want to show that by a proper projection, we can obtain our exact certificate. For $K > 0$, let (x, τ, y) be a point on the central path such that $\langle c, x\rangle = \langle c, \bar{x}(K)\rangle$. Then, by using property (4.18), we have (as before $u = Ax + \frac{1}{\tau}z^0$)

$$\begin{aligned}
\frac{\|u - A\bar{x}(K)\|^2_{\Phi''(u)}}{1 + \|u - A\bar{x}(K)\|_{\Phi''(u)}} &\leq \frac{\tau}{\mu}\langle y - \bar{y}(K), Ax + \frac{1}{\tau}z^0 - A\bar{x}(K)\rangle \\
&= -\frac{\tau}{\mu}\langle(\tau - t(K))c, x - \bar{x}(K)\rangle + \frac{1}{\mu}\langle y - \bar{y}(K), z^0\rangle \\
&= \frac{1}{\mu}\langle y - \bar{y}(K), z^0\rangle. \qquad (7.50)
\end{aligned}$$

By showing that the y vectors are bounded, we get our strict unboundedness detector. Assume that the intersection of D_* with the null space of $A^\top$ is the zero vector. Then, the set of $y \in D_*$ that satisfies $A^\top y = A^\top y^0 - (\tau - 1)c$ for a $1 \leq \tau \leq t_d(y^0)$ is bounded. In view of this, we have the following proposition which is our strict unboundedness detector:

Proposition 7.4.2. *(Strict primal unboundedness detector) Assume that problem (5.1) is strictly primal unbounded. Then, the following constant*

$$K_5 := \max\{\|y\| : y \in D_*, A^\top y = A^\top y^0 - (\tau - 1)c,\ 1 \leq \tau \leq t_d(y^0)\} \qquad (7.51)$$

is well-defined, and for every $K > 0$, our algorithm returns a point $Ax \in \operatorname{int}D$ with $\langle c, x\rangle < -K$ in at most

$$O\left(\sqrt{\vartheta}\ln\left(\frac{K_{unb,1}}{\vartheta} + \frac{K}{\vartheta}\frac{\|y^0\|^2}{\sigma_d^2} + K_5\|z^0\|\right)\right)$$

iterations, where $K_{unb,1}$ is defined in (7.34).

Proof. Consider $Ah \in \operatorname{int}(\operatorname{rec}(D))$; by definition of D_* is (4.37), for every $y \in D_*$ we have $y^\top Ah < 0$. Hence, $\operatorname{null}(A^\top) \cap D_* = \{0\}$, and by the above explanation, K_5 in (7.51) is well-defined. Also from (7.50) and definition of K_5, we have

$$\frac{\|u - A\bar{x}(K)\|^2_{\Phi''(u)}}{1 + \|u - A\bar{x}(K)\|_{\Phi''(u)}} \leq \frac{2}{\mu}K_5\|z^0\|. \qquad (7.52)$$

This means that if for example $\mu \geq 200K_5\|z^0\|$, then $\frac{\|u - A\bar{x}(K)\|^2_{\Phi''(u)}}{1+\|u - A\bar{x}(K)\|_{\Phi''(u)}} \leq 0.1$ and the projection of u with respect to the norm defined by $\Phi''(u)$ into the set $\{Ax : \langle c, x\rangle \leq -K\}$ is in $\operatorname{int}D$. Also note that after at most (7.35) number of iterations, we get a point with $\langle c, x\rangle \leq -K$. Putting these two together, we get the statement of the proposition. ☐

Chapter 8

Software and applications

We have designed and analyzed algorithms that achieve the current best iteration complexity bounds in solving a problem in the Domain-Driven setup to a prescribed accuracy. Even with the best theoretical results, we must show the potential in practice to fully justify our methods. This chapter is about this transition from theory to applications and implementation of algorithms. The resulting code based on our algorithms is still under development and is called DDS (abbreviation for Domain-Driven Solver). This code solves a large group of problems including those we listed in Chapter 2. We discuss several parts about the code such as the format of the input and the numerical ideas and tricks we used in it, for example in calculating the gradients and Hessians efficiently. Note that in the current version, the algorithm we use is simpler than the one we discussed in previous chapters. This simpler algorithm uses a primal-dual path following algorithm that has one parameter τ in addition to the primal and dual variables. This algorithm is described in Appendix B. We are still working on incorporating more sophisticated algorithmic ideas and we expect to get much improvement. In this chapter, we mostly focus on the implementation of different set constraints. One advantage of the Domain-Driven method is that we can add different set constraints without worrying about the core algorithm that can be improved independently.

Before showing the format of input for our set constraints, let us see the format of input for some other famous solvers.

8.1 Format of the input for two famous solvers

Before giving the format of the input for the problems that our code accepts, let us take a look at some other well-known solvers. [37] is a survey by Mittelmann about solvers for conic optimization, which gives an overview of the major codes available for the solution of linear semidefinite (SDP) and second-order cone (SOCP) programs. Many of these codes also solve linear programs (LP). We consider SeDuMi from the list. We also look at CVX, a user-friendly interface for convex optimization. CVX is not a solver, but is a modeling system that (by following some rules) detects if a given problem is convex and remodels it as the input for solvers such as SeDuMi.

8.1.1 SeDuMi

For SeDuMi, our problem is in the format:

$$\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b, \\ & x \in K, \end{array} \tag{8.1}$$

where our cone K can be a direct sum of nonnegative rays (leading to LP problems), second-order cones or semidefinite cones. We give as the input A, b and c and a structure array K. The vector of variables has a "direct sum" structure. In other words, the set of variables is the direct sum of free, linear, quadratic, or semidefinite variables. The fields of the structure array K contain the number of constraints we have from each type and their sizes. SeDuMi can be called in Matlab by the command

```
[x,y] = sedumi(A,b,c,K);
```

and the variables are distinguished by K as follows:

1. $K.f$ is the number of free variables, i.e., in the variable vector x, `x(1:K.f)` are free variables.

2. $K.l$ is the number of nonnegative variables.

3. $K.q$ lists the dimension of Lorentz constraints.

4. $K.s$ lists the dimensions of positive semidefinite constraints.

For example, if `K.l=10`, `K.q=[3 7]` and K.s=[4 3], then `x(1:10)` are non-negative. Then we have `x(11) >= norm(x(12:13))`, `x(14) >= norm(x(15:20))`, and `mat(x(21:36),4)` and `mat(x(37:45),3)` are positive semidefinite. To insert our problem into SeDuMi, we have to write it in the format of (8.1) . We also have the choice to solve the dual problem because all of the above cones are self-dual.

8.1.2 CVX

CVX is an interface that is more user-friendly than solvers like SeDuMi. It provides many options for giving the problem as an input, and then translates them to an eligible format for a solver such as SeDuMi. We can insert our problem constraint-by-constraint into CVX, but they must follow a protocol called *Disciplined convex programming* (DCP). DCP has a rule-set that the user has to follow, which allows CVX to verify that the problem is convex and convert it to a solvable form. For example, we can write a <= constraint only when the left side is convex and the right side is concave, and to do that, we can use a large class of functions from the library of CVX.

The advantage of CVX is that we do not have to be worried about the structure of the variables, and instead we can insert our problem in a more natural way. For example, consider the following problem:

$$\min\{\|Ax-b\|_2 : \ l \le x \le u\}.$$

We can insert this problem in CVX as:

```
cvx_begin
       variable x(n);
       minimize( norm(A*x-b) );
       subject to
                x >= l;
                x <= u;
cvx_end
```

However, to insert it into SeDuMi, we have to do some modification, for example like this:

$$\begin{array}{ll} \min & t \\ \text{s.t.} & y = Ax - b, \\ & \bar{x} = x - l, \\ & \hat{x} = u - x, \\ & \|y\| \le t, \qquad \bar{x}, \hat{x} \ge 0. \end{array} \tag{8.2}$$

8.2 How to use the DDS code

In this section, we explain the format of the input for solving different classes of the optimization problems.

8.2.1 Solving linear programming and SOCP with DDS

Assume that we want to solve the following optimization problem with our code:

$$\begin{aligned}
\min \quad & c^\top x \\
\text{s.t.} \quad & A_L x + b_L \geq 0, \\
& \|A_S^i x + b_S^i\| \leq (g_S^i)^\top x + d_S^i, \quad i = 1, \ldots, \ell,
\end{aligned} \tag{8.3}$$

where A_L is an $m_L \times n_L$ matrix and A_S^i is an $m_S^i \times n_S^i$ matrix for $i = 1, \ldots, \ell$. To give this problem as an input to DDS, we construct the following matrices:

$$A = \begin{bmatrix} A_L \\ (g_S^1)^\top \\ A_S^1 \\ \vdots \\ (g_S^\ell)^\top \\ A_S^\ell \end{bmatrix}, \quad b = \begin{bmatrix} b_L \\ d_S^1 \\ b_S^1 \\ \vdots \\ d_S^\ell \\ b_S^\ell \end{bmatrix}$$

$$Z.L = m_L, \quad Z.SOCP = [m_S^1, \ldots, m_S^\ell]^\top. \tag{8.4}$$

Then $x = DDS(c, A, b, Z)$ solves the above optimization problem. Let us see an example:

Example 8.2.1. *Assume that we have*

$$\begin{aligned}
\min \quad & c^\top x \\
s.t. \quad & [-2, 1]x \leq 1, \\
& \left\| \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix} x + \begin{bmatrix} 3 \\ 4 \end{bmatrix} \right\| \leq 2.
\end{aligned} \tag{8.5}$$

Then we define

$$A = \begin{bmatrix} -2 & 1 \\ 0 & 0 \\ 2 & 1 \\ 1 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

$$Z.L = 1, \quad Z.SOCP = 2. \tag{8.6}$$

8.2.2 Adding SDP to DDS

Consider an SDP constraint in standard inequality form:

$$F_0^i + x_1 F_1^i + \ldots + x_n F_n^i \succeq 0, \quad i = 1, \ldots, \ell. \tag{8.7}$$

F_j^i's are $n_i \times n_i$ symmetric matrices. The above optimization problem is in the matrix form. To formulate it in our setup, we need to write it in the vector form. We have two functions coming with our code, $sm2vec(\cdot)$ and $vec2sm(\cdot)$. $sm2vec(\cdot)$ takes an $n \times n$ symmetric matrix and change it into a n^2 vector by stacking the columns of it on top of one another. $vec2sm(\cdot)$ changes a vector into a symmetric matrix such that

$$vec2sm(sm2vec(X)) = X. \tag{8.8}$$

By this definition, it is easy to check that for any two $n \times n$ symmetric matrices X and Y we have

$$\langle X, Y \rangle = sm2vec(X)^\top sm2vec(Y). \tag{8.9}$$

To give (8.7) to DDS as an input, we define:

$$\begin{aligned} b &:= \begin{bmatrix} sm2vec(F_0^1) \\ \vdots \\ sm2vec(F_0^\ell) \end{bmatrix}, \quad A := \begin{bmatrix} sm2vec(F_1^1), \ldots, sm2vec(F_n^1) \\ \vdots \\ sm2vec(F_1^\ell), \ldots, sm2vec(F_n^\ell) \end{bmatrix}, \\ Z.SDP &= [n^1 \quad \ldots \quad n^\ell]^\top. \end{aligned} \tag{8.10}$$

Then $DDS(c, A, b, Z)$ solves the above optimization problem. To add linear or SOCP constraints, we stack b and A made in the previous subsection on the top of those constructed for SDP in this subsection.

Example 8.2.2. *Assume that we want to find scalars x_1, x_2, and x_3 such that $x_1+x_2+x_3 \geq 1$ and the maximum eigenvalue of $A_0 + x_1 A_1 + x_2 A_2 + x_3 A_3$ is minimized, where*

$$A_0 = \begin{bmatrix} 2 & -0.5 & -0.6 \\ -0.5 & 2 & 0.4 \\ -0.6 & 0.4 & 3 \end{bmatrix}, \ A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \ A_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \ A_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

We can write this problem as

$$\begin{array}{ll} \min & t \\ s.t. & -x_1 - x_2 - x_3 \leq -1, \\ & tI - (A_0 + x_1 A_1 + x_2 A_2 + x_3 A_3) \succeq 0. \end{array} \tag{8.11}$$

To solve this problem, we define:

$$b := \begin{bmatrix} -1 \\ -2 \\ 0.5 \\ 0.6 \\ 0.5 \\ -2 \\ -0.4 \\ 0.6 \\ -0.4 \\ -3 \end{bmatrix}, \quad A := \begin{bmatrix} -1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$c = (0,0,0,1)^\top, \quad Z.L = 1, \quad Z.SDP = 3.$$

Then $DDS(c, A, b, Z)$ gives the answer $x = (1.1265, 0.6, -0.4, 3)$, which means the minimum largest eigenvalue is 3.

Let us elaborate more on calculating the gradient and Hessian for SDP part. For SDP (8.7), we have:

$$\begin{aligned} &\Phi(Z) = -\ln(\det(F_0 + Z)), \\ &\Phi_*(Y) = -n - \langle F_0, Y\rangle - \ln(\det(-Y)). \end{aligned} \tag{8.12}$$

For function $f = -\ln(\det(X))$, we have:

$$\begin{aligned} \langle f'(X), H\rangle &= -\mathrm{Tr}(X^{-1}H), \\ \langle f''(X)H, H\rangle &= \mathrm{Tr}(X^{-1}HX^{-1}H). \end{aligned} \tag{8.13}$$

To implement our algorithm, for each matrix X, we need to find the corresponding gradient g_X and Hessian H_X, such that for any symmetric positive semidefinite matrix X and symmetric matrix H we have:

$$\begin{aligned} -\mathrm{Tr}(X^{-1}H) &= -g_X^\top sm2vec(H), \\ \mathrm{Tr}(X^{-1}HX^{-1}H) &= sm2vec(H)^\top H_X sm2vec(H). \end{aligned} \tag{8.14}$$

It can be shown that $g_X = sm2vec(X^{-1})$ and $H_X = X^{-1} \otimes X^{-1}$, where $\otimes$ stands for the Kronecker product of two matrices. Although this representation is theoretically nice, it is not efficient to calculate the inverse of a matrix explicitly. As we explain, we do not form any inverse of the matrix in our code. Consider forming $A^\top \Phi''(u)A$ for calculating the search directions in (B.3). Usually in practice, matrix A is tall and thin in our setup. Hence, it may not be efficient to form $\Phi''(u)$ as its size may be much bigger than $A^\top \Phi''(u) A$. In our code, we do not form $\Phi''(u)$ and we have a function

```
hessian_A(b,Z,w,A)
```

that directly returns $A^\top \Phi''(u) A$. Note that $\Phi''(u)$ has a block diagonal structure, and each block of the SDP part is equal to H_X defined in (8.14) for a properly chosen X. Hence, calculating $A^\top \Phi''(u) A$ for the SDP part reduces to calculating $v^\top H_X w$ for two vectors v and w. Using (8.14), we have

$$v^\top H_X w = \mathrm{Tr}(X^{-1} vec2sm(v) X^{-1} vec2sm(w)). \tag{8.15}$$

Now, we calculate $X^{-1} vec2sm(v)$ and $X^{-1} vec2sm(w)$ by solving linear systems of equations instead of explicitly forming X^{-1}.

Other numerical difficulties happen for calculating the damped Newton steps in (B.6) and (B.7), specially when the iterates are getting close to the boundary. In DDS, we have different functions to calculate each part of the terms in (B.6) and (B.7). In our functions, we use the properties of Kronecker product that for matrices A, B, and X of proper size, we have

$$\begin{aligned} &(B^\top \otimes A) sm2vec(X) = sm2vec(AXB), \\ &(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}. \end{aligned} \tag{8.16}$$

For calculating the dual damped Newton step in (B.7), we need to calculate $[\Phi''_*(y)]^{-1}$, but we avoid it in our code by using the following formula

$$[\Phi''_*(y)]^{-1} = \Phi''(\Phi'_*(y)). \tag{8.17}$$

For the blocks of the SDP part, we do not even need to use (8.17), because by using the second property in (8.16), for $H_X = X^{-1} \otimes X^{-1}$ we have

$$(H_X)^{-1} = (X^{-1} \otimes X^{-1})^{-1} = X \otimes X. \tag{8.18}$$

In our code, as we explained above, we do not explicitly form $[\Phi''_*(y)]^{-1}$ and the function

```
hessian_Leg_inv_A(b,Z,y,A)
```

returns $A^\top[\Phi''_*(y)]^{-1}A$ directly. We also have the following functions to calculate different parts of the dual damped Newton step direction. The structure of the functions are almost the same, but there are small differences to make it more efficient.

```
hessian_Leg_A(Z,y,A)
hessian_Leg_inv_V_grad(b,Z,y)
hessian_Leg_inv_V(b,Z,y,v)
```

8.2.3 Adding sets created by the epigraph of a matrix norm

Assume that we have constraints of the form

$$\begin{aligned} & X - UU^\top \succeq 0, \\ & X = X_0 + \sum_{i=1}^{\ell} x_i X_i, \\ & U = U_0 + \sum_{i=1}^{\ell} x_i U_i, \end{aligned} \tag{8.19}$$

where X_i, $i \in \{0,\dots,\ell\}$, are $m \times m$ symmetric matrices, and U_i, $i \in \{0,\dots,\ell\}$, are $m \times n$ matrices. We are interested in the case that $m \ll n$. We discussed in Chapter 2 how we use the m-s.c. barrier (2.9) for the set $\{(X,U) : X - UU^\top \succeq 0\}$, with LF conjugate (5.5). We also mentioned that a special but very important application is minimizing the nuclear norm of the matrix.

Our code accepts constraints in the form (8.19) and uses the more efficient s.c. barrier to handle them. We have two functions $m2vec$ and $vec2m$ for converting matrices (not necessarily symmetric) to vectors and vise versa. The field of variable Z for the constraints in the form of (8.19) is $Z.EO2N$ (abbreviated Epigraph of Operator 2-Norm). $Z.EO2N$ is a $k \times 2$ matrix, where k is the number of constraints of this type, and each row is of the form $[m \;\; n]$. For each constraint of the form (8.19), the corresponding parts in A and b are

$$A = \begin{bmatrix} m2vec(U_1, n) & \cdots & m2vec(U_\ell, n) \\ sm2vec(X_1) & \cdots & sm2vec(X_\ell) \end{bmatrix}, \quad b = \begin{bmatrix} m2vec(U_0, n) \\ sm2vec(X_0) \end{bmatrix}. \tag{8.20}$$

Example 8.2.3. *Assume that we have matrices*

$$U_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad U_1 = \begin{bmatrix} -1 & -1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad U_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{8.21}$$

and our goal is to solve

$$\begin{aligned} \min \quad & t \\ s.t. \quad & UU^\top \preceq tI, \\ & U = U_0 + x_1 U_1 + x_2 U_2. \end{aligned} \tag{8.22}$$

Then the parameters of the code are

$$\begin{aligned} & A = \begin{bmatrix} m2vec(U_1, 3) & m2vec(U_2, 3) & zeros(6,1) \\ zeros(4,1) & zeros(4,1) & sm2vec(I_{2\times 2}) \end{bmatrix}, \quad b = \begin{bmatrix} m2vec(U_0, 3) \\ zeros(4,1) \end{bmatrix}, \\ & c = (0, 0, 1), \quad Z.EO2N = [2 \ \ 3]. \end{aligned} \tag{8.23}$$

CVX does not accept a constraint of the form $X - UU^\top \succeq 0$ *and we need to give the SDP representation. By doing that, both codes give the solution* 0.407105*. However, if we change* c *to* $c = (0, 0, -1)$*, the problem is unbounded and our code returns unboundedness, but CVX fails to solve the problem.*

Example 8.2.4. *We consider minimizing the nuclear norm over a subspace. Consider the following optimization problem:*

$$\begin{aligned} \min \quad & \|X\|_* \\ s.t. \quad & \mathrm{Tr}(XU_1) = 1 \\ & \mathrm{Tr}(XU_2) = 2, \end{aligned} \tag{8.24}$$

where

$$U_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad U_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{8.25}$$

By using (2.10)*, the dual of this problem is*

$$\begin{aligned} \min \quad & -u_1 - 2u_2 \\ s.t. \quad & \|u_1 U_1 + u_2 U_2\| \leq 1. \end{aligned} \tag{8.26}$$

To solve this problem with our code, we define

$$\begin{aligned} & A = \begin{bmatrix} m2vec(U_1, 4) & m2vec(U_2, 4) \\ zeros(4,1) & zeros(4,1) \end{bmatrix}, \quad b = \begin{bmatrix} zeros(8,1) \\ sm2vec(I_{2\times 2}) \end{bmatrix}, \\ & c = (-1, -2), \quad Z.EO2N = [2 \ \ 4]. \end{aligned} \tag{8.27}$$

If we solve the problem, the optimal value is -2.2360*. Now the dual solution is* (Y,V) *and* $\frac{1}{\tau}V$ *is the solution of* (8.24) *with objective value* 2.2360*. We have*

$$X^* := \frac{1}{\tau}V = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{8.28}$$

What we did in the last example can be done in general. For the optimization problem

$$\begin{array}{ll} \min & \|X\|_* \\ s.t. & \operatorname{Tr}(XU_i) = c_i, \quad i \in \{1, \cdots, l\}, \end{array} \tag{8.29}$$

where X is $n \times m$, we solve the dual problem by defining

$$\begin{array}{l} A = \begin{bmatrix} m2vec(U_1, n) & \cdots & m2vec(U_l, n) \\ zeros(m^2, 1) & \cdots & zeros(m^2, 1) \end{bmatrix}, \quad b = \begin{bmatrix} zeros(mn, 1) \\ sm2vec(I_{m\times m}) \end{bmatrix}, \\ Z.EO2N = [m \;\; n]. \end{array} \tag{8.30}$$

Then, $\frac{1}{\tau}V$ is the optimal solution for (8.29).

Let us see how to calculate the first and second derivatives of functions in (2.9) and (5.5).

Proposition 8.2.1. *(a) Consider* $\Phi(X,U)$ *defined in* (2.9)*. Let, for simplicity,* $\bar{X} := X - UU^\top$*, then, we have*

$$\begin{array}{rcl} \Phi'(X,U)[(d_X, d_U)] & = & \operatorname{Tr}(-\bar{X}^{-1}d_X + \bar{X}^{-1}(d_U U^\top + U d_U^\top)), \\ \Phi''(X,U)[(d_X, d_U), (\bar{d}_X, \bar{d}_U)] & = & \operatorname{Tr}(\bar{X}^{-1}d_X\bar{X}^{-1}\bar{d}_X) \\ & & -\operatorname{Tr}(\bar{X}^{-1}\bar{d}_X\bar{X}^{-1}(d_U U^\top + U d_U^\top)) \\ & & -\operatorname{Tr}(\bar{X}^{-1}d_X\bar{X}^{-1}(\bar{d}_U U^\top + U \bar{d}_U^\top)) \\ & & +\operatorname{Tr}(\bar{X}^{-1}(d_U U^\top + U d_U^\top)\bar{X}^{-1}(\bar{d}_U U^\top + U \bar{d}_U^\top)) \\ & & +2\operatorname{Tr}(\bar{X}^{-1}d_U\bar{d}_U^\top). \end{array} \tag{8.31}$$

(b) Consider $\Phi_*(Y,V)$ *defined in* (5.5)*, we have*

$$\begin{array}{rcl} \Phi_*'(Y,V)[(d_Y, d_V)] & = & -\frac{1}{2}\operatorname{Tr}(V^\top Y^{-1}d_V) + \frac{1}{4}\operatorname{Tr}(Y^{-1}VV^\top Y^{-1}d_Y) - \operatorname{Tr}(Y^{-1}d_Y), \\ \Phi_*''(Y,V)[(d_Y, d_V), (\bar{d}_Y, \bar{d}_V)] & = & -\frac{1}{2}\operatorname{Tr}(d_V^\top Y^{-1}\bar{d}_V) \\ & & +\frac{1}{2}\operatorname{Tr}(Y^{-1}d_V V^\top Y^{-1}\bar{d}_Y) + \frac{1}{2}\operatorname{Tr}(Y^{-1}\bar{d}_V V^\top Y^{-1}d_Y) \\ & & -\frac{1}{2}\operatorname{Tr}(Y^{-1}d_Y Y^{-1}\bar{d}_Y Y^{-1}VV^\top) + \operatorname{Tr}(Y^{-1}d_Y Y^{-1}\bar{d}_Y). \end{array}$$

Proof. For the proof we use the fact that if $g = -\ln(\det(X))$, then $g'(X)[H] = \mathrm{Tr}(X^{-1}H)$. Also note that if we define

$$g(\alpha) := -\ln(\det(X + \alpha d_X - (U + \alpha d_U)(U + \alpha d_U)^\top)), \tag{8.32}$$

then

$$g'(0) = \Phi'(X,U)[(d_X, d_U)], \quad g''(0) = \Phi''(X,U)[(d_X, d_U), (d_X, d_U)],$$

and similarly for $\Phi_*(Y,V)$. We do not bring all the details, but we show how the proof works. For example, let us define

$$f(\alpha) := \mathrm{Tr}((Y + \alpha d_Y)^{-1} V V^\top Y^{-1} d_Y), \tag{8.33}$$

and we want to calculate $f'(0)$. We have

$$\begin{aligned}
f'(0) &:= \lim_{\alpha \to 0} \frac{f(\alpha) - f(0)}{\alpha} \\
&= \mathrm{Tr}\left(\lim_{\alpha \to 0} \frac{(Y + \alpha d_Y)^{-1} V V^\top Y^{-1} d_Y - Y^{-1} V V^\top Y^{-1} d_Y}{\alpha} \right) \\
&= \mathrm{Tr}\left(\lim_{\alpha \to 0} \frac{(Y + \alpha d_Y)^{-1} \left[V V^\top Y^{-1} d_Y - (I + \alpha d_Y Y^{-1}) V V^\top Y^{-1} d_Y \right]}{\alpha} \right) \\
&= \mathrm{Tr}\left(\lim_{\alpha \to 0} (Y + \alpha d_Y)^{-1} \left[d_Y Y^{-1} V V^\top Y^{-1} d_Y \right] \right) \\
&= \mathrm{Tr}\left(Y^{-1} d_Y Y^{-1} V V^\top Y^{-1} d_Y \right).
\end{aligned} \tag{8.34}$$

□

Note that all the above formulas for the derivatives are in matrix form. Let us explain briefly how to convert them to the vector form for the code. We explain it for the derivatives of $\Phi(X,U)$ and the rest are similar. From (8.31) we have

$$\begin{aligned}
\Phi'(X,U)[(d_X, d_U)] &= \mathrm{Tr}(-\bar{X}^{-1} d_X) + \mathrm{Tr}(\bar{X}^{-1} d_U U^\top) + \mathrm{Tr}(X^{-1} U d_U^\top)), \\
&= \mathrm{Tr}(-\bar{X}^{-1} d_X) + 2\mathrm{Tr}(U^\top \bar{X}^{-1} d_U).
\end{aligned} \tag{8.35}$$

Hence, if g is the gradient of $\Phi(X,U)$ in the vector form, we have

$$g = \begin{bmatrix} 2 \times m2vec(X^{-1}U, n) \\ -sm2vec(X^{-1}) \end{bmatrix}. \tag{8.36}$$

The second derivatives are trickier. Assume that for example we want the vector form h for $\Phi''(X,U)[(d_X, d_U)]$. By using (8.31) we can easily get each entry of h; consider the identity matrix of size $m^2 + mn$. If we choose $(\bar{d}_X, \bar{d}_U)$ to represent the jth column of this identity matrix, we get $h(j)$. Practically, this can be done by a for loop, which is not efficient. What we did in the code is to implement this using matrix multiplication.

8.2.4 Adding quadratic constraints

Assume that we want to add the following constraints to our code:

$$x^\top A_i^\top A_i x + b_i^\top x + d_i \leq 0, \quad i = 1, \ldots, \ell, \tag{8.37}$$

where each A_i is $m_i \times n$ with rank m_i. Each of the constraints in (8.37) can be written as

$$\begin{aligned} & u^\top u + w + d \leq 0 \\ & u = A_i x, \quad w = b_i^\top x, \quad d = d_i. \end{aligned} \tag{8.38}$$

We can associate the following s.c. barrier and its LF conjugate to quadratic constraints:

$$\begin{aligned} \Phi(u, w) &= -\ln(-(u^\top u + w + d)), \\ \Phi_*(y, \eta) &= \frac{y^\top y}{4\eta} - 1 - d\eta - \ln(\eta). \end{aligned} \tag{8.39}$$

To give constraints in (8.37) as an input to our code, we construct the following matrices:

$$\begin{aligned} & A = \begin{bmatrix} b_1^\top \\ A_1 \\ \vdots \\ b_l^\top \\ A_\ell \end{bmatrix}, \quad b = \begin{bmatrix} d_1 \\ 0 \\ \vdots \\ d_\ell \\ 0 \end{bmatrix} \\ & Z.QC = [m_1, \ldots, m_\ell]^\top. \end{aligned} \tag{8.40}$$

8.2.5 Adding constraints defined by epigraph of univariate functions

As we mentioned in Chapter 2, our code accepts constraints of the form (2.3) for f_i's from the first 5 rows of Table 2.2. However, every interesting univariate convex function can be added in the same fashion. The corresponding s.c. barriers and their LF conjugate for the first three rows are shown in Table (5.1.2). We see numbers 4 and 5 explicitly later in the chapter. In view of (5.4), to represent a constraint of the from (2.3), for given $\gamma \in \mathbb{R}$ and $\beta_i \in \mathbb{R}$, $i \in \{1, \cdots, \ell\}$, we can define our set D_i as

$$D_i := \{(w, s_i, t_i) : w + \gamma \leq 0, \quad f_i(s_i + \beta_i) \leq u_i, \ \forall i\}, \tag{8.41}$$

and our matrix A represents $w = \sum_{i=1}^{\ell} \alpha_i u_i + g^\top x$ and $s_i = a_i^T x$, $i \in \{1, \cdots, \ell\}$. As can be seen, to show our set as above, we need to add some artificial variables u_i's to our formulations. DDS code does it internally and we do not need to insert them. Let us assume that we want to add the following k constraints to our code

$$\sum_{type} \sum_{i=1}^{\ell_{type}^j} -\alpha_i^{j,type} f_{type}((a_i^{j,type})^\top x + \beta_i^{j,type}) + g_j^\top x + \gamma_j \leq 0, \qquad j = 1, \ldots, k.$$

From now on, *type* indexes the rows of Table 2.2. To add these constraints, we add two fields to Z. The first one, $Z.TD1$, is a matrix with 3 columns. In each row, the first entry is the index of constraint, the second entry is the type, and the third entry is the number of functions in that type we have in that constraint. Let us say that in the jth constraint, we have l_2^j functions of type 2 and l_3^j functions of type 3, then the corresponding columns in $Z.TD1$ are as follows

$$Z.TD1 = \begin{bmatrix} & \vdots & \\ j & 2 & l_2^j \\ j & 3 & l_3^j \\ & \vdots & \end{bmatrix}. \tag{8.42}$$

Note that the index of constraints must start from 1 and all the rows corresponding to each constraint must be consecutive. The types can be in any order, but they must match with the rows of A and b.

We also add $Z.TD2$ which is a row vector that contains all the coefficients in each constraint. Note that the coefficients must be in the same order as their corresponding rows in A. If in the first constraint we have 2 functions of type 2 and 1 function of type 3, it starts as

$$Z.TD2 = [\alpha_1^{1,2}, \alpha_2^{1,2}, \alpha_1^{1,3}, \ldots]. \tag{8.43}$$

To add the rows to A, for each constraint j, we first add g_j, then $a_i^{j,type}$'s in the order that matches $Z.TD1$ and $Z.TD2$. We do the same thing for vector b (first γ_j, then $\beta_i^{j,type}$'s). The part of A and b corresponding to the jth constraint is as follows if we have all five types

$$A = \begin{bmatrix} g_j^\top \\ a_1^{j,1} \\ \vdots \\ a_{l_1^j}^{j,1} \\ \vdots \\ a_1^{j,5} \\ \vdots \\ a_{l_5^j}^{j,5} \end{bmatrix}, \quad b = \begin{bmatrix} \gamma_j \\ \beta_1^{j,1} \\ \vdots \\ \beta_{l_1^j}^{j,1} \\ \vdots \\ \beta_1^{j,5} \\ \vdots \\ \beta_{l_5^j}^{j5} \end{bmatrix} \tag{8.44}$$

Let us see the following example:

Example 8.2.5. *Assume that we want to solve*

$$\begin{aligned} \min \quad & c^\top x \\ s.t. \quad & -\ln(x_2 + 2x_3 + 55) + 2e^{x_1+x_2+1} + x_1 - 2 \leq 0, \\ & -3\ln(x_1 + 2x_2 + 3x_3 - 30) + e^{-x_3-3} - x_3 + 1 \leq 0, \\ & x \geq 0. \end{aligned} \tag{8.45}$$

For this problem, we define:

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & 1 & 2 \\ 1 & 1 & 0 \\ 0 & 0 & -1 \\ 1 & 2 & 3 \\ 0 & 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2 \\ 55 \\ 1 \\ 1 \\ -30 \\ -3 \end{bmatrix},$$

$$Z.L = 3, \quad Z.TD1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \\ 2 & 2 & 1 \end{bmatrix}, \quad Z.TD2 = [1 \ \ 2 \ \ 3 \ \ 1]. \tag{8.46}$$

The first three rows of A and b are for linear constraints. Here is the code in CVX:

```
cvx_begin
variable x(3)
minimize (c'*x)
subject to
    x>=0;
    -log(x(2)+2*x(3)+55) + 2*exp(x(1)+x(2)+1)+x(1)-2 <= 0;
    -3*log(x(1)+2*x(2)+3*x(3)-30) + exp(-x(3)-3) -x(3)+1 <= 0;
cvx_end
```

CVX uses successive approximation method for these kinds of problems. When $c = (1, 1, 1)$*, both codes return the correct answer with objective value of* 10.0165*. When we put* $c = (1, 1, -1)$*, the problem is unbounded. CVX does not return a meaningful solution, but our code returns* $(0, 0, 1.23 \times 10^6)$ *as a certificate.*

Let us add a function of type 3 to the first constraint and change sign constraints:

$$\begin{aligned} \min \quad & c^\top x \\ s.t. \quad & -\ln(x_2 + 2x_3 + 55) + 2e^{x_1+x_2+1} + (x_2 - 3x_3)\ln(x_2 - 3x_3) + x_1 - 2 \leq 0, \\ & -3\ln(x_1 + 2x_2 + 3x_3 - 30) + e^{-x_3-3} - x_3 + 1 \leq 0, \\ & x_2, x_3 \geq 0, \;\; x_1 \leq 0. \end{aligned} \tag{8.47}$$

For CVX to recognize it as a convex optimization problem, we use entropy function $entr(z) = -z\ln(z)$ *from its library. For* $c = (0, 1, 1)$*, both codes return* $x = (-13.2167, 14.4958, 4.8322)$ *as the optimal solution. If we change* $c = (0, 1, 1)$*, the problem becomes unbounded. DDS returns* $10^8 \times (-4.9050, 1.6350, 0.5450)$ *as a certificate, but CVX does not return a meaningful solution.*

Adding the constraints for $|z|^p \leq t, \; p \geq 1$:

Now we want to add the sets defined by constraints $|z|^p \leq t, \; p \geq 1$. These functions are of type 4. The corresponding s.c. barrier is $\Phi(z, t) = -\ln(t^{\frac{2}{p}} - z^2) - 2\ln(t)$. Let us first see how to calculate the LF conjugate. We need to solve the following optimization problem:

$$\min_{z,t} \left\{ yz + \eta t + \ln(t^{\frac{2}{p}} - z^2) + 2\ln(t) \right\}. \tag{8.48}$$

The optimal solution satisfies:

$$y = \frac{2z}{t^{\frac{2}{p}} - z^2}, \quad \eta = -\frac{\frac{2}{p} t^{\frac{2}{p}-1}}{t^{\frac{2}{p}} - z^2} - \frac{2}{t}. \tag{8.49}$$

By doing some algebra, we can see that z and t satisfy:

$$
\begin{aligned}
&y\left(\frac{2(\frac{1}{p}+1)+\frac{1}{p}yz}{-\eta}\right)^{\frac{2}{p}} - yz^2 - 2z = 0,\\
&t = \frac{2(\frac{1}{p}+1)+\frac{1}{p}yz}{-\eta}.
\end{aligned} \tag{8.50}
$$

Let us define $z(y, \eta)$ as the solution of the first equation in (8.50). For each pair (y, η), we can calculate $z(y, \eta)$ by few iterations of Newton method. Then, the first and second derivative can be calculated in terms of $z(y, \eta)$.

In our code, we have two functions for these derivatives.

```
p1_TD(y,eta,p)     % returns  z
p1_TD_der(y,eta,p)  % returns [z_y  z_eta  z_y,y  z_y,eta  z_eta,eta]
```

To add these kind of functions to our code, we follow the same rule as before, but the type of the functions is 4. The difference with previous 3 cases is that we also need to give the value of p for each function. To do that, we need to add another field to Z which is $Z.TD3$. $Z.TD3$ is in the same length as $Z.TD2$ and it has zero for functions of types 1 to 3. For functions of type 4 and 5, we put the power p in exactly the same place we put the coefficient of the function. Let us see an example:

Example 8.2.6.

$$
\begin{aligned}
\min \quad & c^\top x\\
s.t. \quad & 2\exp(2x_1+3) + |x_1+x_2+x_3|^2 + 4.5|x_1+x_2|^{2.5} + |x_2+2x_3|^3 + x_1 - 2 \le 0.
\end{aligned}
$$

For this problem, we define:

$$
A = \begin{bmatrix} 1 & 0 & 0\\ 2 & 0 & 0\\ 1 & 1 & 1\\ 1 & 1 & 0\\ 0 & 1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -2\\ 3\\ 0\\ 0\\ 0 \end{bmatrix},
$$

$$
Z.TD1 = \begin{bmatrix} 1 & 2 & 1\\ 1 & 4 & 3 \end{bmatrix}, \quad Z.TD2 = [2 \;\; 1 \;\; 4.5 \;\; 1], \quad Z.TD3 = [0 \;\; 2 \;\; 2.5 \;\; 3].
$$

Our code solves this problem and returns objective value -2.87198. CVX also solves the problem by using successive approximation method and returns the same solution.

Adding the constraints for $-z^p \le t,\ 0 \le p \le 1, z > 0$:

Now we want to add the sets defined by constraints $-z^p \le t,\ 0 \le p \le 1, z > 0$. These functions are of type 5. The corresponding s.c. barrier is $\Phi(z,t) = -\ln(z^p + t) - \ln(z)$. Let us first see how to calculate the LF conjugate. We need to solve the following optimization problem:

$$\min_{z,t} \left\{ yz + \eta t + \ln(z^p + t) + \ln(z) \right\}. \tag{8.51}$$

The optimal solution satisfies:

$$y = \frac{-pz^{(p-1)}}{z^p + t} - \frac{1}{z}, \quad \eta = -\frac{1}{z^p + t}. \tag{8.52}$$

By doing some algebra, we can see that z satisfies:

$$y - \eta p z^{(p-1)} + \frac{1}{z} = 0. \tag{8.53}$$

Similar to the previous case, let us define $z(y,\eta)$ as the solution of the first equation in (8.53). For each pair (y,η), we can calculate $z(y,\eta)$ by few iterations of Newton method. Then, the first and second derivative can be calculated in terms of $z(y,\eta)$. The important point is that when we calculate $z(y,\eta)$, then the derivatives can be calculated by explicit formulas. In our code, we have two functions

```
p2_TD(y,eta,p)     % returns  z
p2_TD_der(y,eta,p)  % returns [z_y  z_eta  z_y,y  z_y,eta  z_eta,eta]
```

The inputs to the above functions can be vectors. Table 8.2.5 is the continuation of Table 5.1.1.

8.3 Equality constraints

In the Domain-Driven formulation (5.1), to have simplicity, we prefer that D does not contain a straight line, which is equivalent to the non-degeneracy of the corresponding s.c. barrier $\Phi(\cdot)$. This restriction makes it difficult to insert equality constraints. With equality constraints, the feasible region may be as

$$\{x : Bx = d, Ax \in D\}, \tag{8.54}$$

Table 8.1: s.c. barriers and their LF conjugate for rows 4 and 5 of Table 2.2

	s.c. barrier $\Phi(z,t)$	$\Phi_*(y,\eta)$
4	$-\ln(t^{\frac{2}{p}} - z^2) - 2\ln(t)$	$-\left(\frac{2}{p} + (\frac{1}{p} - 1)yz(y,\eta)\right) - 2 + 2\ln\left(\frac{2(\frac{1}{p}+1)+\frac{1}{p}yz(y,\eta)}{-\eta}\right)$ $+\ln\left(\left(\frac{2(\frac{1}{p}+1)+\frac{1}{p}yz(y,\eta)}{-\eta}\right)^{\frac{2}{p}} - z^2(y,\eta)\right)$
5	$-\ln(z^p + t) - \ln(z)$	$\eta(p-1)z^p(y,\eta) - 2 - \ln(-\eta) + \ln(z(y,\eta))$

where B and d are a matrix and a vector of appropriate sizes. Mathematically, dealing with that is not a problem. For any matrix Z that its columns form a basis for the null space of B, we can write all the solutions of $Bx = d$ as $x^0 + Zw$, where x^0 is any solution of $Bx = d$, then the feasible region in (8.54) is equivalent to:

$$D_w := \{w : AZw \in (D - Ax^0)\}. \tag{8.55}$$

$D - Ax^0$ is a translation of D with the s.c. barrier $\Phi(z - Ax^0)$. Now we have to work with the matrix AZ instead of A. Even though this procedure is straightforward in theory, there might be numerical challenges in application. For example, if we have a nice structure for A, such as sparsity, multiplying with Z may ruin the structure.

Finding Z can be done efficiently by using QR factorization:

$$B^T = \begin{bmatrix} Y & Z \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix}$$

We can also use Y and R to find a solution of $Bx = d$ as $x^0 = YR^{-1}d$. QR factorization is ideal from a numerical stability point of view. One problem with this approach is that if A is sparse, it may be very costly to maintain sparsity in AZ. Hence, this approach might only be efficient for medium-size problems, unless one finds efficiently a Z matrix that maintains the sparsity of A in AZ.

There is another approach for finding Z that is less costly, but also can give rise to numerical instabilities. We can denote this approach as the elimination approach. Assuming that B has full row rank, there exist permutation matrices P such that

$$BP = \begin{bmatrix} B_1 & B_2 \end{bmatrix},$$

where B_1 is a non-singular matrix. Then we can define:

$$Z = P \begin{bmatrix} -B_1^{-1} B_2 \\ I \end{bmatrix}, \quad x^0 = P \begin{bmatrix} B_1^{-1} d \\ 0 \end{bmatrix}.$$

This approach is very efficient if the number of equality constraints is much less than the number of variables.

8.4 More numerical examples

In this section, we present some numerical examples of running our code. The first set of examples are from the Dimacs library for LP, SOCP, and SDP [58]. We bring the name and properties of the example and the output of our code. Note that the problems in the library are for the standard equality form and we solve the dual of the problems. Here is a typical output for the problem torusmp-8-50.

```
Iteration:   t       Rel-Duality-Gap    Pfeas        Dfeas        Obj-Val
  1:   | 1.36e+00    |  1.37e+00    |  8.60e-01   |  0.00e+00   |  1.789771e+01
  2:   | 1.71e+00    |  7.10e-01    |  7.46e-01   |  0.00e+00   |  5.378506e+01
  3:   | 2.13e+00    |  4.41e-01    |  6.16e-01   |  0.00e+00   |  1.071573e+02
  4:   | 2.67e+00    |  3.00e-01    |  4.90e-01   |  0.00e+00   |  1.703928e+02


 33:   | 1.34e+07    |  3.74e-08    |  7.89e-08   |  0.00e+00   |  5.278086e+02
 34:   | 2.67e+07    |  1.87e-08    |  3.95e-08   |  0.00e+00   |  5.278086e+02
 35:   | 5.34e+07    |  9.35e-09    |  1.97e-08   |  0.00e+00   |  5.278086e+02
 36:   | 1.07e+08    |  4.68e-09    |  9.86e-09   |  0.00e+00   |  5.278087e+02

 Status: Solved; returned vector x is an optimal solution.
 Primal feasibility; norm(z0/t)/(1+norm(Ax+z0/t))= 9.86e-09 <= tol=1.00e-08.
 Dual feasibility; norm(A*(y/t)+c)/(1+norm(c))= 0.00e+00 <= tol = 1.00e-08.
 Optimal objective value: 5.27808654e+02.
```

Table 8.2 shows the result for some problems from the Dimacs library.

Table 8.2: Numerical results for some problem from the Dimacs library for $tol = 10^{-8}$.

Name	size of A	SDP	SOCP	LP	Iterations
torusmp-8-50	262144×512	$[1; 512]$	-	-	36
torusg3-8	262144×512	$[1; 512]$	-	-	38
qssp30	7566×3691	-	$[1891; 1891 \times 4]$	2	83
turss5	3301×208	$[34; 33 \times 10, 1]$	-	-	86
turss8	11914×496	$[34; 33 \times 19, 1]$	-	-	91
nb	2383×123	-	$[793; 793 \times 3]$	4	49
nb_L1	3176×915	-	$[793; 793 \times 3]$	797	66
nb_L2	4195×123	-	$[839; 1 \times 1677, 838 \times 3]$	4	38
nb_L2_bessel	2641×123	-	$[839; 1 \times 123, 838 \times 3]$	4	41
copo14	3108×1285	$[14; 14 \times 14]$	-	364	54
copo23	13938×5820	$[23; 23 \times 23]$	-	1771	72
filter48	3284×969	$[1; 48]$	$[1; 49]$	931	114
filtinf1	3395×983	$[1; 49]$	$[1; 49]$	945	9 (unbounded)

Chapter 9

Conclusion

In this thesis, we designed infeasible-start primal-dual algorithms for convex optimization that not only achieve the current best iteration complexity bounds for "modifying the RHS" type formulations, but also show considerable potential in practice. We defined the Domain-Driven setup for convex optimization. This setup is minimizing a linear function over the intersection of an affine subspace and a convex set D, which is represented as the domain of a s.c. barrier Φ. Φ and its Legendre-Fenchel conjugate Φ_* can be evaluated efficiently in the Domain-Driven setup. Several interesting classes of convex optimization problems were proved to be in the Domain-Driven setup. We emphasized the generality of the setup by showing that direct sum operator lets us solve optimization problems with arbitrarily large number of constraints from each set type. We defined a notion of duality gap for the Domain-Driven setup and designed an infeasible-start primal-dual central path with parameter μ.

A class of algorithms was designed to follow the path efficiently to $\mu = +\infty$. We showed how to interpret a point returned by the algorithms to determine the status of the problem. We defined several possible statuses for a problem in the Domain-Driven setup and proved that our algorithms can detect many of them in polynomial time, with the current best iteration complexity bounds. We finished the thesis with the implementation chapter and the introduction of our code DDS.

The continuation of this research leans more towards the applications, and improving and expanding the code. Our plan is to release a robust code that solves a large class of convex optimization problems. We can split our future goals into two categories.

9.1 Improving the algorithm in the code

The algorithm we use in the current version of the code is using one additional variable τ as the parameter of the path. We have not been able to prove the best theoretical results for this practical version of the algorithm. The algorithm we designed and analyzed in this thesis is more complicated and uses two additional parameters τ and μ. There is a paradox about optimization algorithms that the ones working well in practice are not necessarily the best performers in terms of worst-case complexity analyses. However, we believe that a rigorous and robust code must be supported by a strong theory. We will continue looking for opportunities to improve the software by continuing our experiments on the ideas arising from our theoretical analysis. We also intend to use our computational experience with our software to generate new algorithmic ideas as well as ideas for complexity analysis.

In addition to the core algorithm, there are many other factors that affect the performance of a code. One of the most important ones is the linear system solver. As we explained briefly in Chapter 8 and also in Appendix B, we have to solve at least one linear system in each iteration, and the performance of the linear solver is a bottleneck for the performance of the code. There are many different approaches for solving linear systems that exploit the structure of the system, such as sparsity pattern. To have a competent code, we have to find a robust linear system solver that best suits our setup and code.

The algorithms we designed in this thesis are second-order algorithms, which means they use the exact Hessian of the underlying functions. Even though these algorithms are the best in terms of accuracy, they are not generally suitable for huge scale data. An approach that we want to use in our code is quasi-Newton-type methods that have been proven to provide good performance in large scale data. In these methods, second-order information is used; however, the Hessian matrix is derived by low-rank updates. Tunçel in [74] and later with Myklebust in [44] proposed interior-point methods that use low-rank updates for the Hessian in each iteration and achieve the current best theoretical results. We are interested in studying and implementing this idea in our code, which seems to have the robustness of interior-point methods and the speed of quasi-Newton approaches.

9.2 Expanding the code

We introduced a list of set constraints/functions and explained how to make arbitrarily large optimization problems by direct sum of different set types. A power of the Domain-Driven setup is that this list can be expanded with the same algorithm as the core. If we find a s.c. function that its LF conjugate can also be calculated efficiently, we basically have a type of constraints added to our list. This motivates us to look for possible functions and applications to expand the scope of the code. We can even take a drastic step by considering convex functions that are not s.c. barrier. If we are given a non-degenerate convex function with an efficiently computable LF conjugate, all the steps of our algorithms are well-defined. We may lose the theoretical guarantees we proved in this thesis, but we can add such a function to our code and try its efficiency in practice.

References

[1] B. Alkire and L. Vandenberghe. Convex optimization problems involving finite autocorrelation sequences. *Mathematical Programming*, 93(3, Ser. A):331–359, 2002.

[2] B. P. W. Ames and S. A. Vavasis. Nuclear norm minimization for the planted clique and biclique problems. *Mathematical Programming*, 129(1):69–89, 2011.

[3] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

[4] A. Ben-Tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, volume 2. SIAM, 2001.

[5] A. Ben-Tal and A. Nemirovski. Robust optimization–methodology and applications. *Mathematical Programming*, 92(3):453–480, 2002.

[6] J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization: Theory and Examples*. Springer Science & Business Media, 2010.

[7] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, 1994.

[8] S. Boyd, S.-J. Kim, L. Vandenberghe, and A. Hassibi. A tutorial on geometric programming. *Optimization and Engineering*, 8(1):67–127, 2007.

[9] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[10] E. Candes and B. Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

[11] R. Chares. *Cones and Interior-Point Algorithms for Structured Convex Optimization Involving Powers and Exponentials.* PhD thesis, Université Catholique de Louvain, Louvain-la-Neuve, 2008.

[12] IBM ILOG CPLEX. V12. 1: User's Manual for CPLEX. *International Business Machines Corporation*, 46(53):157, 2009.

[13] D. L. Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[14] S.-C. Fang, J. R. Rajasekera, and H.-S. J. Tsao. *Entropy optimization and Mathematical Programming*, volume 8. Springer Science & Business Media, 1997.

[15] J. Faraut and A. Korányi. *Analysis on Symmetric Cones.* Oxford University Press, 1994.

[16] A. V. Fiacco and G. P. McCormick. The sequential unconstrained minimization technique for nonlinear programing, a primal-dual method. *Management Science*, 10(2):360–366, 1964.

[17] A. Forsgren, P. E. Gill, and J. D. Griffin. Iterative solution of augmented systems arising in interior methods. *SIAM Journal on Optimization*, 18(2):666–690, 2007.

[18] R. W. Freund, F. Jarre, and S. Schaible. On self-concordant barrier functions for conic hulls and fractional programming. *Mathematical Programming*, 74(3):237–246, 1996.

[19] C. C. Gonzaga. An algorithm for solving linear programming problems in $O(n^3L)$ operations. In *Progress in Mathematical Programming*, pages 1–28. Springer, 1989.

[20] M. Grant, S. Boyd, and Y. Ye. CVX: MATLAB software for disciplined convex programming, 2008.

[21] S. Granville, J. Mello, and A. C. G. Melo. Application of interior point methods to power flow unsolvability. *IEEE Transactions on Power Systems*, 11(2):1096–1103, 1996.

[22] T. Hastie, R. Tibshirani, and M. Wainwright. *Statistical Learning with Sparsity: The Lasso and Generalizations.* CRC press, 2015.

[23] J.-B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis.* Springer Science & Business Media, 2001.

[24] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[25] K. Kobayashi, S. Kim, and M. Kojima. Correlative sparsity in primal-dual interior-point methods for LP, SDP, and SOCP. *Applied Mathematics and Optimization*, 58(1):69–88, 2008.

[26] M. Kojima, N. Megiddo, and S. Mizuno. A primal-dual infeasible-interior-point algorithm for linear programming. *Mathematical Programming*, 61(1):263–280, 1993.

[27] M. Kojima, S. Mizuno, and A. Yoshise. A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, 44(1-3):1–26, 1989.

[28] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. In *Progress in Mathematical Programming*, pages 29–47. Springer, 1989.

[29] M. Kojima, S. Mizuno, and A. Yoshise. A little theorem of the big-M in interior point algorithms. *Mathematical Programming*, 59(1):361–375, 1993.

[30] Y. T. Lee and A. Sidford. Path finding I: Solving linear programs with $\tilde{O}(\sqrt{(}rank))$ linear system solves. *arXiv preprint arXiv:1312.6677*, 2013.

[31] Y. Lucet. What shape is your conjugate? a survey of computational convex analysis and its applications. *SIAM Review*, 52(3):505–542, 2010.

[32] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Duality and self-duality for conic convex programming. Technical report, Erasmus University Rotterdam, Erasmus School of Economics (ESE), Econometric Institute, 1996.

[33] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Conic convex programming and self-dual embedding. *Optimization Methods and Software*, 14(3):169–218, 2000.

[34] I. J Lustig, R. E. Marsten, and D. F. Shanno. Computational experience with a primal-dual interior point method for linear programming. *Linear Algebra and its Applications*, 152:191–222, 1991.

[35] K. A. McShane, C. L. Monma, and D. Shanno. An implementation of a primal-dual interior point method for linear programming. *ORSA Journal on Computing*, 1(2):70–83, 1989.

[36] N. Megiddo. Pathways to the optimal set in linear programming. In *Progress in Mathematical Programming*, pages 131–158. Springer, 1989.

[37] H. D. Mittelmann. The state-of-the-art in conic optimization software. In *Handbook on Semidefinite, Conic and Polynomial Optimization*, pages 671–686. Springer, 2012.

[38] S. Mizuno. Polynomiality of infeasible-interior-point algorithms for linear programming. *Mathematical Programming*, 67(1):109–119, 1994.

[39] S. Mizuno, M. Kojima, and M. J. Todd. Infeasible-interior-point primal-dual potential-reduction algorithms for linear programming. *SIAM Journal on Optimization*, 5(1):52–67, 1995.

[40] S. Mizuno, M. J. Todd, and Y. Ye. On adaptive-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18(4):964–981, 1993.

[41] R. D. C. Monteiro and I. Adler. Interior path following primal-dual algorithms. Part I: Linear programming. *Mathematical Programming*, 44(1):27–41, 1989.

[42] R. D. C. Monteiro and I. Adler. Interior path following primal-dual algorithms. Part II: Convex quadratic programming. *Mathematical Programming*, 44(1):43–66, 1989.

[43] APS Mosek. The MOSEK optimization software. *Online at http://www. mosek. com*, 54, 2010.

[44] T. Myklebust and L. Tunçel. Interior-point algorithms for convex optimization based on primal-dual metrics. *arXiv preprint arXiv:1411.2129*, 2014.

[45] A. Nemirovski. Interior point polynomial time methods in convex programming. *Lecture Notes*, 2004.

[46] A. Nemirovski and L. Tunçel. Cone-free primal-dual path-following and potential reduction polynomial time interior-point methods. *Mathematical Programming*, 102:261–294, 2005.

[47] Y. Nesterov. The method for linear programming which requires $O(n^3L)$ operations. ekonomika i matem. metody 24. *Russian; English translation in Matekon: Translations of Russian and East European Math. Economics*, pages 174–176, 1988.

[48] Y. Nesterov. Infeasible-start interior-point primal-dual methods in nonlinear programming. *Université catholique de Louvain, Center for Operations Research and Econometrics (CORE)*, (1995067), 1995.

[49] Y. Nesterov. Long-step strategies in interior-point primal-dual methods. *Mathematical Programming*, 76(1):47–94, 1997.

[50] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2004.

[51] Y. Nesterov. Towards non-symmetric conic optimization. *Optimization Methods and Software*, 27(4-5):893–917, 2012.

[52] Y. Nesterov and A. Nemirovski. *Interior point polynomial methods in Convex Programming*. SIAM Series in Applied Mathematics, SIAM: Philadelphia, 1994.

[53] Y. Nesterov and A. Nemirovski. Multi-parameter surfaces of analytic centers and long-step surface-following interior-point methods. *Mathematics of Operations Research*, 23:1–38, 1998.

[54] Y. Nesterov and M. J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22(1):1–42, 1997.

[55] Y. Nesterov and M. J. Todd. Primal-dual interior-point methods for self-scaled cones. *SIAM Journal on Optimization*, 8(2):324–364, 1998.

[56] Y. Nesterov, M. J. Todd, and Y. Ye. Infeasible-start primal-dual methods and infeasibility detectors for nonlinear programming problems. *Mathematical Programming*, 84:227–267, 1999.

[57] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Science & Business Media, 2006.

[58] G. Pataki and S. Schmieta. The DIMACS library of semidefinite-quadratic-linear programs. Technical report, Preliminary draft, Computational Optimization Research Center, Columbia University, New York, 2002.

[59] J. Peña and H. S. Sendov. Separable self-concordant spectral functions and a conjecture of Tunçel. *Mathematical Programming*, 125(1):101–122, 2010.

[60] J. Peña and H. S. Sendov. Spectral self-concordant functions in the space of two-by-two symmetric matrices. *Optimization*, 60(4):441–449, 2011.

[61] V. H. Quintana, G. L. Torres, and J. Medina-Palomo. Interior-point methods and their applications to power systems: a classification of publications and software codes. *IEEE Transactions on Power Systems*, 15(1):170–176, 2000.

[62] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[63] J. Renegar. A polynomial-time algorithm, based on Newton's method, for linear programming. *Mathematical Programming*, 40(1-3):59–93, 1988.

[64] R. T. Rockafellar. *Convex analysis*. Princeton University Press, 2015.

[65] R. T. Rockafellar and R. J.-B. Wets. *Variational Analysis*. Springer Science & Business Media, 2009.

[66] R. Schneider. *Convex Bodies: The Brunn–Minkowski Theory*. Cambridge University Press, 2013.

[67] A. Seifi and L. Tunçel. A constant-potential infeasible-start interior-point algorithm with computational experiments and applications. *Computational Optimization and Applications*, 9(2):107–152, 1998.

[68] A. Skajaa and Y. Ye. A homogeneous interior-point algorithm for nonsymmetric convex conic optimization. *Mathematical Programming*, 150(2):391–422, 2015.

[69] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.

[70] J. F. Sturm. Similarity and other spectral relations for symmetric cones. *Linear Algebra and its Applications*, 312(1-3):135–154, 2000.

[71] J. F. Sturm. Implementation of interior point methods for mixed semidefinite and second order cone optimization problems. *Optimization Methods and Software*, 17(6):1105–1154, 2002.

[72] K.-C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3– a MATLAB software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11(1-4):545–581, 1999.

[73] K.-C. Toh, R. H. Tütüncü, and M. J. Todd. On the implementation and usage of SDPT3–a MATLAB software package for semidefinite-quadratic-linear programming, version 4.0. 2006.

[74] L. Tunçel. Generalization of primal-dual interior-point methods to convex optimization problems in conic form. *Foundations of Computational Mathematics*, 1(3):229–254, 2001.

[75] L. Tunçel. *Polyhedral and semidefinite programming methods in combinatorial optimization*, volume 27. American Mathematical Soc., 2010.

[76] R. J. Vanderbei. Symmetric quasidefinite matrices. *SIAM Journal on Optimization*, 5(1):100–113, 1995.

[77] R. J. Vanderbei and T. J. Carpenter. Symmetric indefinite systems for interior point methods. *Mathematical Programming*, 58(1, Ser. A):1–32, 1993.

[78] X. Xu, P.-F. Hung, and Y. Ye. A simplified homogeneous and self-dual linear programming algorithm and its implementation. *Annals of Operations Research*, 62(1):151–171, 1996.

[79] G. Xue and Y. Ye. An efficient algorithm for minimizing a sum of p-norms. *SIAM Journal on Optimization*, 10.2:551– 579, 2000.

[80] Y. Ye, M. J. Todd, and S. Mizuno. An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research*, 19(1):53–67, 1994.

[81] Y. Zhang. On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem. *SIAM Journal on Optimization*, 4(1):208–227, 1994.

Appendix A

Converting Domain-Driven setup into conic optimization

One way to approach the problems in Domain-Driven setup, at least theoretically, is to convert them into a conic formulation and use the strong machinery of primal-dual conic optimization. To do that, we represent D as the intersection of its conic hull and an affine subspace. Then, we can associate a logarithmically homogeneous self-concordant barrier to the conic hull of D as done in [52]:

$$\Phi^+(z, z_\tau) = \xi_1[\Phi(z/z_\tau) - \xi_2 \vartheta \ln z_\tau], \tag{A.1}$$

where ξ_1 and ξ_2 are appropriate constants. Nesterov and Nemirovskii in [52] prove that $\xi_1 = 400$ and $\xi_2 = 2$ work for every $\Phi(\cdot)$. The constants in the construction are improved in [18] to $\xi_1 = 25$ and $\xi_2 = 7$.

If $\Phi_*(\cdot)$, the Legendre-Fenchel conjugate of $\Phi(x)$ defined in (A.1), is also available, we can obtain the Legendre-Fenchel conjugate of $\Phi^+(z, z_\tau)$ as a function of $\Phi_*(\cdot)$ as follows:

$$(\Phi^+)_*(y, y_\tau) = \max_{\gamma > 0} \left[\xi_1 \Phi_*(\gamma y/\xi_1) + y_\tau \gamma + \xi_1 \xi_2 \vartheta \ln \gamma\right]. \tag{A.2}$$

$(\Phi^+)_*$ means that we first change into the conic form and then take the Legendre-Fenchel conjugate. Note that $(\Phi^+)_*(-s, -s_\tau)$ is a logarithmically homogeneous self-concordant barrier for the dual cone K^* (let K be the conic hull of D we define later). This shows that the conjugate function at each point can be calculated by a one-dimensional maximization. Having these functions at hand, we can apply the standard primal-dual interior-point methods for conic formulations.

Let us define K as the closed conic hull of $D \subset \mathbb{Y}$ in the space $\mathbb{Y} \oplus \mathbb{R}$:

$$K = \text{cl} \left\{ \begin{bmatrix} z \\ z_\tau \end{bmatrix} : z_\tau > 0, \ \frac{z}{z_\tau} \in D \right\}. \tag{A.3}$$

In the Domain-Driven formulation (5.1), the feasible set consists of the points x such that $z = Ax \in D$. Let F be a matrix whose rows give a basis for $\text{null}(A^\top)$. Then it is easy to check that $z = Ax$ if and only if $Fz = 0$. Hence, we can write our optimization problem in the conic form as:

$$\begin{array}{ll} \inf & \langle \bar{c}, z \rangle \\ & \begin{bmatrix} F & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z \\ z_\tau \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} z \\ z_\tau \end{bmatrix} \in K, \end{array} \tag{A.4}$$

where $\bar{c}$ is such that $A^\top \bar{c} = c$. Let us define $\hat{c} := (\bar{c}, 0)^\top$ and write the equality constraint of (A.4) as $\bar{A}\bar{z} = b$. Then, we have a conic primal-dual optimization setup as follows:

$$\begin{array}{lll} \inf & \langle \hat{c}, \bar{z} \rangle \\ \text{s.t.} & \bar{A}\bar{z} = b, \\ & \bar{z} \in K. \end{array} \qquad \begin{array}{lll} \inf & \langle b, \bar{v} \rangle \\ \text{s.t.} & \bar{s} = \hat{c} + \bar{A}^*\bar{v}, \\ & \bar{s} \in K^*. \end{array} \tag{A.5}$$

Assume that both the primal and dual systems in (A.5) are strictly feasible, then the pairs of primal-dual optimal solutions for (A.5) are equivalent to the points $\{(\bar{z}, \bar{\tau}, \bar{s}, \bar{v}) : \bar{\tau} > 0\}$ that satisfy the following system:

$$\begin{array}{l} \bar{A}\bar{z} = \bar{\tau} b, \\ \bar{s} = \bar{\tau}\hat{c} + \bar{A}^*\bar{v}, \\ \langle \hat{c}, \bar{z} \rangle + \langle b, \bar{v} \rangle = 0, \\ \bar{z} \in K, \ \ \bar{s} \in K^*. \end{array} \tag{A.6}$$

Let us study the structure of K^*. By definition, we have:

$$K^* = \left\{ \begin{bmatrix} s \\ s_\tau \end{bmatrix} : \langle s, z \rangle + s_\tau z_\tau \geq 0, \ \forall \begin{bmatrix} z \\ z_\tau \end{bmatrix} \in K \right\}. \tag{A.7}$$

If $z_\tau > 0$, then the inequality inside the definition (A.7) is equivalent to $\frac{1}{z_\tau}\langle s, z \rangle + s_\tau \geq 0$. By definition of K, this inequality implies that $\langle s, u \rangle$ is bounded below for all points $u \in D$. Let $\text{rec}(D)$ be the recession cone of D, then we must have:

$$\langle -s, h \rangle \leq 0, \ \forall h \in \text{rec}(D).$$

Hence, we can write K^* as:

$$K^* = \left\{ \begin{bmatrix} s \\ s_\tau \end{bmatrix} : s \in [\text{rec}(D)]^*, \langle s, w\rangle + s_\tau \geq 0,\ \forall w \in D \right\}. \tag{A.8}$$

The following proposition makes a connection between K^* and the polar of D, denoted by D^o, which is defined in Definition 4.1.4.

Proposition A.0.1. *In view of* (A.8)*, we have*

$$\begin{array}{lll} K^* & \subseteq & \left\{ \begin{bmatrix} s \\ s_\tau \end{bmatrix} : s \in [\text{rec}(D)]^*, \begin{bmatrix} s \\ s_\tau \end{bmatrix} \in \text{cone} \begin{bmatrix} -D^o \\ 1 \end{bmatrix} \cup \begin{bmatrix} [\text{cone}(D)]^* \\ \mathbb{R}_- \end{bmatrix} \right\} \\ K^* & \supseteq & \left\{ \begin{bmatrix} s \\ s_\tau \end{bmatrix} : s \in [\text{rec}(D)]^*, \begin{bmatrix} s \\ s_\tau \end{bmatrix} \in \text{cone} \begin{bmatrix} -D^o \\ 1 \end{bmatrix} \cup \begin{bmatrix} [\text{cone}(D)]^* \\ 0 \end{bmatrix} \right\}. \end{array} \tag{A.9}$$

Proof. To prove the $\subseteq$ direction, assume that $[s^\top, s_\tau]^\top \in K^*$. By definition (A.8) we have $\langle s, w\rangle + s_\tau \geq 0$ for all $w \in D$. If $s_\tau > 0$, then we have

$$\langle \frac{s}{s_\tau}, w\rangle + 1 \geq 0, \quad \forall w \in D \quad \Rightarrow \quad -\frac{s}{s_\tau} \in D^o.$$

Now assume that $s_\tau \leq 0$, then we have

$$\begin{array}{rl} \langle s, w\rangle \geq -s_\tau \geq 0, \quad \forall w \in D & \Rightarrow \quad \langle s, w\rangle \geq 0, \quad \forall w \in \text{cone}(D) \\ & \Rightarrow \quad s \in [\text{cone}(D)]^*. \end{array}$$

To prove the $\supseteq$ direction, assume that $[s^\top, s_\tau]^\top$ is in the RHS of (A.9). We need to consider two cases: for the first one

$$\begin{array}{rl} \begin{bmatrix} s \\ s_\tau \end{bmatrix} \in \text{cone} \begin{bmatrix} -D^o \\ 1 \end{bmatrix} & \Rightarrow \quad \langle -\frac{s}{s_\tau}, w\rangle + 1 \geq 0, \quad \forall w \in D \\ & \Rightarrow \quad \langle s, w\rangle + s_\tau \geq 0, \quad \forall w \in D. \end{array}$$

For the second case, $s \in [\text{cone}(D)]^*$ and $s_\tau = 0$, where the inclusion is clear from the definitions. □

A solution of (A.6) is a recession direction of a cone that can be found approximately by minimizing the corresponding self-concordant barrier over an unbounded set. Assume that we are given points $\bar{z}^0 \in \text{int}K$ and $\bar{s}^0 \in \text{int}K^*$. Let us define set Q as

$$\begin{array}{rcl} Q : & = & \{(\bar{z}, \bar{\tau}, \bar{s}, \bar{v}) : \ \bar{A}\bar{z} = \bar{A}\bar{z}^0 + \bar{\tau} b, \quad \bar{s} = \bar{s}^0 + \bar{\tau}\hat{c} + \bar{A}^\top \bar{v}, \\ & & \langle \hat{c}, \bar{z}\rangle + \langle b, \bar{v}\rangle = \langle \hat{c}, \bar{z}^0\rangle, \quad \bar{z} \in \text{int}K, \ \bar{s} \in \text{int}K^* \}. \end{array} \tag{A.10}$$

The shifted central path in [48] is defined as solution set $(\bar{z}(\mu), \bar{\tau}(\mu), \bar{s}(\mu), \bar{v}(\mu))$, $\mu > 0$, of the system

$$\begin{array}{ll} \min & \frac{1}{\mu}\left[\langle \bar{s}^0, \bar{z}\rangle + \langle \bar{s}, \bar{z}^0\rangle\right] + \Phi^+(\bar{z}) + (\Phi^+)_*(-\bar{s}) \\ \text{s.t.} & (\bar{z}, \bar{\tau}, \bar{s}, \bar{v}) \in Q. \end{array} \tag{A.11}$$

Remark A.0.1. *It is shown in [48] and we can also easily verify that* $\langle \bar{s}^0, \bar{z}\rangle + \langle \bar{s}, \bar{z}^0\rangle = \langle \bar{s}, \bar{z}\rangle + \langle \bar{s}^0, \bar{z}^0\rangle$*. Hence, in problem (A.11), we are minimizing* $\Phi^+(\bar{z}) + (\Phi^+)_*(-\bar{s})$ *and at the same time we are allowing* $\langle \bar{s}^0, \bar{z}\rangle + \langle \bar{s}, \bar{z}^0\rangle$ *to go to* $+\infty$*. It seems contradictory, but the balance between these two increases gives us the desired property. This kind of balance is analogous to what is done with usual potential reduction methods in interior-point methods (as the feasible iterates approach the optimal solution set, the barrier part of the potential function tends to* $+\infty$ *where as the logarithm of the duality gap tends to* $-\infty$*). We are interested in* $(\bar{z}/\bar{\tau}, \bar{s}/\bar{\tau})$*, and by using Theorem 1 in [48], for the points on the central path we have:*

$$\langle \frac{\bar{z}}{\bar{\tau}}, \frac{\bar{s}}{\bar{\tau}}\rangle = \frac{\langle \bar{s}^0, \bar{z}\rangle + \langle \bar{s}, \bar{z}^0\rangle - \langle \bar{s}^0, \bar{z}^0\rangle}{\bar{\tau}^2} = \frac{\xi_1\xi_2\vartheta\mu}{\bar{\tau}^2}.$$

This equality is saying that if μ *and* τ *increase at the same rate, the duality gap converges to zero. Note that following the path defined in (A.11) by increasing* μ *is equivalent to minimizing the function* $\Phi^+(\bar{z}) + (\Phi^+)_*(-\bar{s})$*, which gives us an approximation of a recession direction for (A.6).*

It is proved in [48] that for the points on the central path we have

$$\bar{s}(\mu) = -\mu(\Phi^+)'(\bar{z}(\mu)). \tag{A.12}$$

By considering this equation and some simplification, we can show that the central path is also the solution set $(w(\mu), \bar{\tau}(\mu), v(\mu))$, $\mu > 0$, of the following system:

$$\begin{array}{ll} (a) & -s \in \text{int}D_*, \quad \dfrac{Aw + z^0}{z^0_\tau + \bar{\tau}} \in \text{int}D, \quad z^0_\tau + \bar{\tau} > 0, \\ (b) & s - s^0 = \bar{\tau}\bar{c} + F^\top v, \\ (c) & s = -\dfrac{\mu\xi_1}{z^0_\tau + \bar{\tau}}\Phi'\left(\dfrac{Aw + z^0}{z^0_\tau + \bar{\tau}}\right), \\ (d) & \dfrac{\mu\xi_1\xi_2}{z^0_\tau + \bar{\tau}} = \dfrac{s^0_\tau - \langle c, w\rangle + \langle s, \frac{Aw+z^0}{z^0_\tau+\bar{\tau}}\rangle}{\vartheta}. \end{array} \tag{A.13}$$

If we define suitable variables to remove the linear equations in (A.11), we can write it as an unconstrained optimization problem. Let us define

$$\begin{aligned}\Psi(w,\bar{\tau},v) \quad &:= \quad \xi_1\Phi\left(\frac{Aw+z^0}{z_\tau^0+\bar{\tau}}\right) - \xi_1\xi_2\vartheta\ln(z_\tau^0+\bar{\tau}) + \max_{\gamma>0}\left[\xi_1\Phi_*(-\gamma s/\xi_1) - s_\tau\gamma + \xi_1\xi_2\vartheta\ln\gamma\right] \\ & \qquad s = s^0+\bar{\tau}\bar{c}+F^\top v, \quad s_\tau = s_\tau^0 - c^\top w. \end{aligned} \tag{A.14}$$

Then we can verify that

$$\langle \bar{s}^0, \bar{z}\rangle + \langle \bar{s}, \bar{z}^0\rangle = [w^\top \ \bar{\tau} \ \ v^\top]r^0 + 2(\langle s^0, z^0\rangle + s_\tau^0 z_\tau^0), \quad r^0 := \left[\begin{array}{c} A^\top s^0 - c \\ s_\tau^0 + \langle \bar{c}, z^0\rangle \\ F z^0 \end{array}\right]. \tag{A.15}$$

Writing the first order optimality conditions for (A.11) and considering (A.14) and (A.15), we get the following lemma:

Lemma A.0.1. *For every $(w(\mu),\bar{\tau}(\mu),v(\mu))$ on the central path we have*

$$\Psi'(w(\mu),\bar{\tau}(\mu),v(\mu)) = -\frac{r^0}{\mu}, \tag{A.16}$$

where r^0 is defined in (A.15).

In [48] and [56], the path-following algorithms are predictor-corrector that the search direction in the predictor step is approximately tangent to the central path (exactly tangent when the starting point in on the central path). If we take the derivative of both sides of (A.16) with respect to μ, we get the tangent to the central path. When we are off the central path, we may choose to use the same formula as an approximation. Therefore, the predictor search direction is the solution of

$$\Psi''(w,\bar{\tau},v)d = \Psi''(w,\bar{\tau},v)\left[\begin{array}{c} d_w \\ d_\tau \\ d_v \end{array}\right] = \frac{1}{\mu^2}r^0, \tag{A.17}$$

where we explicitly have

$$\Psi''(w,\bar{\tau},v) \quad = \quad U^\top \left[\begin{array}{cc} \bar{H}(z,z_\tau) & 0 \\ 0 & \bar{H}_*(-s,-s_\tau) \end{array}\right] U. \tag{A.18}$$

U is the same matrix we defined in (6.8), and $\bar{H}(z,z_\tau)$ and $\bar{H}_*(y,y_\tau)$ are the Hessians of $\Phi^+(z,z_\tau)$ and $(\Phi^+)_*(y,y_\tau)$, respectively.

Appendix B

Predictor-corrector algorithm used in the code

As we mentioned at the beginning of Chapter 8, the algorithm we use in DDS is simpler than the one we designed in this thesis. In this Chapter, we briefly discuss this predictor-corrector algorithm. To define our infeasible-start primal-dual central path, we pick an arbitrary point $z^0 \in \text{int}D$ and define $y^0 := \Phi'(z^0)$. Then, the solution set of the system

$$\begin{array}{ll} (a) & y \in \text{int}D_*, \quad Ax + \frac{1}{\tau}z^0 \in \text{int}D, \\ (b) & A^\top y + (-c - A^\top y^0) = -\tau c, \\ (c) & y = \Phi'(Ax + \frac{1}{\tau}z^0), \end{array} \tag{B.1}$$

for $\tau \geq 1$ forms our central path. Note that here we treat τ as the parameter of the central path in contrast to previous sections where it was a variable. Moreover, we removed the (old) central path parameter μ from the formulation. We denote the points on the central path by $(x(\tau), y(\tau))$. For our central path, for every pair of points (x, y), we define the proximity measure as

$$\Omega(x, \tau, y) = \Phi\left(Ax + \frac{1}{\tau}z^0\right) + \Phi_*(y) - \langle y, Ax + \frac{1}{\tau}z^0 \rangle. \tag{B.2}$$

For the above setup, we use a long step predictor-corrector algorithm. Let us define the following search directions and updates (we define $u := Ax + \frac{1}{\tau}z^0$):

$$\begin{aligned}
d_{x,1} &:= \left(A^\top \Phi''(u)A\right)^{-1} A^\top \Phi''(u)z^0, \\
d_{x,2} &:= \left(A^\top \Phi''(u)A\right)^{-1} (-c), \\
x' &:= \frac{1}{(1+\delta)\tau^2} d_{x,1} + d_{x,2}, \\
u' &:= Ax' - \frac{1}{(1+\delta)\tau^2} z^0, \\
y' &:= \Phi''(u)\left(Ax' - \frac{1}{(1+\delta)\tau^2} z^0\right), \quad (\Rightarrow y' = \Phi''(u)u'), \\
x^+ &= x + \delta\tau x', \\
y^+ &= y + \delta\tau y'.
\end{aligned} \tag{B.3}$$

Then, our long-step algorithm is as follows:

Simple Predictor-Corrector Algorithm

1. Parameters $\delta_2 > \delta_1 > 0$ are given by the user. Choose $z^0 \in \mathrm{int}D$ and set $x^0 = 0$, $y^0 = \Phi'(z^0)$, $\tau^0 = 1$, and $k = 0$.
   Repeat until the stopping criteria are met:
2. **Predictor step**: If $\Omega(x^k, \tau^k, y^k) \leq \delta_1$, calculate $d_{x,1}$ and $d_{x,2}$ in (B.3) and find the maximum δ such that (x^+, y^+) defined in (B.3) satisfies $\Omega(x^+, \tau^+, y^+) \leq \delta_2$ for $\tau^+ = (1+\delta)\tau$.
3. **Corrector step**: Apply damped Newton steps (will be described later) to get the point (x^{k+1}, y^{k+1}) that satisfies $\Omega(x^{k+1}, \tau^{k+1}, y^{k+1}) \leq \delta_1$ for $\tau^{k+1} = \tau^k$.

B.1 Predictor and corrector steps

The formula for the search directions is given in (B.3). If we calculate d_y explicitly based on $d_{x,1}$ and $d_{x,2}$ we get

$$\begin{aligned}
d_y &= \Phi''(u)A\frac{1}{(1+\delta)\tau^2} d_{x,1} + \Phi''(u)Ad_{x,2} - \frac{1}{(1+\delta)\tau^2}\Phi''(u)z^0 \\
&= \frac{1}{(1+\delta)\tau^2} \underbrace{\left[\Phi''(u)Ad_{x,1} - \Phi''(u)z^0\right]}_{=:d_{y,1}} + \underbrace{\Phi''(u)Ad_{x,2}}_{=:d_{y,2}}
\end{aligned}$$

Doing some simple algebra, we can derive the following system for calculating $d_{x,1}$, $d_{x,2}$, $d_{y,1}$ and $d_{y,2}$:

$$\begin{bmatrix} A^\top & 0 \\ I & -[\Phi''(u)]A \end{bmatrix} \begin{bmatrix} d_{y,1} & d_{y,2} \\ d_{x,1} & d_{x,2} \end{bmatrix} = \begin{bmatrix} 0 & -c \\ -\Phi''(u)z^0 & 0 \end{bmatrix}. \tag{B.4}$$

This shows that for calculating d_x and d_y we need to solve one augmented system of equations. Moreover, this system has exactly the form of the systems we need for calculating corrector steps. For the corrector step, we take damped Newton steps defined in (4.20) for s.c. functions. Let us define the following function for a fixed τ:

$$F_\tau(x) \quad := \quad \tau\langle c, x\rangle + \Phi(Ax + \frac{1}{\tau}z^0) + \langle -c - A^\top y^0, x\rangle. \tag{B.5}$$

Then, the primal damped Newton step is

$$\begin{aligned} x^+ \quad &= \quad x + \frac{1}{1+\lambda(F_\tau, x)}e(x), \quad e(x) := -\left[F_\tau''(x)\right]^{-1} F_\tau'(x), \\ & \qquad \lambda(F_\tau, x) := \left(\langle e(x), \Phi''(u)e(x)\rangle\right)^{1/2}, \end{aligned} \tag{B.6}$$

and the dual damped Newton step is

$$\begin{aligned} y^+ \quad = \quad & y + \frac{1}{1+\lambda_*(y)}e(y), \\ & e(y) := -[\Phi_*''(y)]^{-1}\left[I - A(A^\top[\Phi_*''(y)]^{-1}A)^{-1}A^\top[\Phi_*''(y)]^{-1}\right](\Phi_*'(y) - z^0/\tau), \\ & \lambda_*(y) := \left(\langle e(y), \Phi_*''(y)e(y)\rangle\right)^{1/2}. \end{aligned} \tag{B.7}$$

The problem with numerical calculation of the dual damped Newton step is the number of times we have to find the inverse of $\Phi_*''(y)$ (solve the corresponding system), which makes it impossible to evaluate it accurately specially at the final iterations. Here we can use a trick to avoid part of the ill-conditioning. Let us define

$$p := [(A^\top[\Phi_*''(y)]^{-1}A)^{-1}A^\top[\Phi_*''(y)]^{-1}](\Phi_*'(y) - z^0/\tau).$$

Also note that $A^\top e(y) = 0$. Hence, $e(y)$ is a solution of the system

$$\begin{aligned} e(y) \quad &= \quad [\Phi_*''(y)]^{-1}(\Phi_*'(y) - z^0/\tau) - [\Phi_*''(y)]^{-1}Ap, \\ A^\top e(y) \quad &= \quad 0. \end{aligned} \tag{B.8}$$

We can show that $e(y)$ is the unique solution. This means that $e(y)$ can be calculated by solving one of the following two systems:

$$\begin{aligned}
\begin{bmatrix} A^\top & 0 \\ I & [\Phi_*''(y)]^{-1}A \end{bmatrix} \begin{bmatrix} e(y) \\ p \end{bmatrix} &= \begin{bmatrix} 0 \\ [\Phi_*''(y)]^{-1}(\Phi_*'(y) - z^0/\tau) \end{bmatrix}, \\
\begin{bmatrix} A^\top & 0 \\ [\Phi_*''(y)] & A \end{bmatrix} \begin{bmatrix} e(y) \\ p \end{bmatrix} &= \begin{bmatrix} 0 \\ (\Phi_*'(y) - z^0/\tau) \end{bmatrix}.
\end{aligned} \tag{B.9}$$

Note that these systems are bigger than the original systems; however, they can be very sparse. Similarly, we can write a bigger system that gives us the primal damped Newton step.

$$\begin{bmatrix} A^\top & 0 \\ I & -[\Phi''(u)]A \end{bmatrix} \begin{bmatrix} p \\ e(x) \end{bmatrix} = \begin{bmatrix} -F_\tau'(x) \\ 0 \end{bmatrix}. \tag{B.10}$$

An interesting fact is that the LHS matrices in systems (B.9) and (B.10) are very similar and it would be more efficient if we can combine them to solve one system for calculating $e(x)$ and $e(y)$. Here is one idea to achieve this goal. The points on the central path corresponding to τ satisfy the following system

$$\begin{aligned}
y - \Phi'\left(Ax + \frac{1}{\tau}z^0\right) &= 0, \\
A^\top y - c - A^\top y^0 + \tau c &= 0.
\end{aligned} \tag{B.11}$$

If we write the Newton system for solving it, we get

$$\begin{bmatrix} A^\top & 0 \\ I & -[\Phi''(u)]A \end{bmatrix} \begin{bmatrix} e(y) \\ e(x) \end{bmatrix} = \begin{bmatrix} 0 \\ \Phi'(u) - y \end{bmatrix}. \tag{B.12}$$

If we evaluate $e(x)$ and $e(y)$, $e(x)$ is exactly the one given by the above equations. Comparing this system with the first system in (B.9) shows that $e(y)$s are not exactly the same. However, it works quite the same in practice while reducing the arithmetic operations.

Note that solving systems of the above form, which are denoted as *augmented systems* in the literature, have been studied for specific problems such as linear and quadratic programming [77, 76, 17, 25].

B.2 Stopping criteria

For the case that the problem is solvable, we are looking for an approximately feasible point with a small duality gap. Let us remind ourselves of our definition for the duality gap in Definition 5.2.1. For every point $x \in \mathbb{R}^n$ such that $Ax \in D$ and every point $y \in D_*$ such that $A^\top y = -c$, the duality gap is defined as:

$$\langle c, x\rangle + \mathcal{S}(y), \tag{B.13}$$

where $\mathcal{S}$ is the support function of D. Because $\Phi_*(\cdot)$ is the Legendre-Fenchel conjugate of a ϑ-s.c. barrier, by Theorem 4.2.1, we can calculate the support function efficiently as for every $k > 0$ and every $y \in \text{int}D_*$,

$$\mathcal{S}(y) - \frac{\vartheta}{k} \leq \langle \Phi_*'(ky), y\rangle \leq \mathcal{S}(y). \tag{B.14}$$

This means that we can calculate the duality gap with a desired accuracy if we have the points $Ax \in D$ and $y \in \text{int}D_*$ such that $A^\top y = -c$. In the case that our problem is solvable, by running our path following algorithm, for every $\tau \geq 1$ we have a point x such that $Ax + \frac{1}{\tau}z^0 \in D$ and a point $y \in D_*$ such that $A^\top(y/\tau) + ((-c - A^\top y^0)/\tau) = -c$. This means that when $\tau \to +\infty$, the points x and y converge to primal and dual feasible points and the duality gap converges to zero. For the desired accuracy ϵ, we define the following parameter (we actually set $k = \vartheta/\epsilon$ in (B.14))

$$\text{gap} := \langle \frac{y}{\tau}, \Phi_*'(\vartheta y/\epsilon) - Ax\rangle + \frac{\epsilon}{\tau}. \tag{B.15}$$

Let x^* be an optimal solution for our problem. Then, we can write

$$\begin{aligned}
\langle c, x\rangle - \langle c, x^*\rangle &= \langle \frac{-A^\top y}{\tau} + \frac{(-c - A^\top y^0)}{\tau}, x - x^*\rangle \\
&= \langle \frac{y}{\tau}, Ax^* - Ax\rangle + \langle \frac{(-c - A^\top y^0)}{\tau}, x - x^*\rangle \\
&= \text{gap} - \frac{\epsilon}{\tau} + \frac{\epsilon}{\vartheta\tau}\langle \vartheta y/\epsilon, Ax^* - \Phi_*'(\vartheta y/\epsilon)\rangle + \frac{1}{\tau}\langle(-c - A^\top y^0), x - x^*\rangle \\
&\leq \text{gap} - \frac{\epsilon}{\tau} + \frac{\epsilon}{\tau} + \frac{1}{\tau}\langle(-c - A^\top y^0), x - x^*\rangle, \quad \text{using (4.27)} \\
&= \text{gap} + \frac{1}{\tau}\langle(-c - A^\top y^0), x - x^*\rangle.
\end{aligned} \tag{B.16}$$

As τ gets large, the last term converges to zero. This means that the gap is a certificate of closeness to the optimal solution. We show that the gap reduces to the conic duality gap when we solve a conic optimization problem. Assume that we have a conic optimization problem in standard equality form, we write it in a way that its dual matches our format:

$$\begin{aligned}
&\text{(P)} \quad \min\{\langle -b, y\rangle : A^\top y = -c, y \in K^*\}, \\
&\text{(D)} \quad \min\{\langle c, x\rangle : Ax - b \in K\}.
\end{aligned} \tag{B.17}$$

We can solve the dual problem by our code with $DDS(c, A, b, Z)$. Assume that K is equipped with a ϑ-LH s.c. barrier $H(\cdot)$, and let us define $\Phi(u) := H(u - b)$. By this definition, the LF conjugate of $\Phi(\cdot)$ becomes

$$\Phi_*(y) = \langle b, y\rangle + H_*(y). \tag{B.18}$$

Note that $H_*(y)$ is also a ϑ-LH s.c. barrier. If we substitute this into (B.15), we get

$$\begin{aligned} \text{gap} &= \langle \frac{y}{\tau}, b\rangle + \langle -A^\top \frac{y}{\tau}, x\rangle + \frac{\epsilon}{\vartheta\tau}\langle \vartheta y/\epsilon, H_*'(\vartheta y/\epsilon)\rangle + \frac{\epsilon}{\tau} \\ &= \langle \frac{y}{\tau}, b\rangle + \langle -A^\top \frac{y}{\tau}, x\rangle - \frac{\epsilon}{\tau} + \frac{\epsilon}{\tau} \\ &= \langle \frac{y}{\tau}, b\rangle + \langle -A^\top \frac{y}{\tau}, x\rangle, \end{aligned} \tag{B.19}$$

where the second equation is by using a property of ϑ-LH s.c. barriers that $\langle y, H_*'(y)\rangle = -\vartheta$ for any $y \in \text{int}K^*$. When $\tau \to +\infty$, then $-A^\top y/\tau \to c$ and y/τ converges to a feasible solution for (P), and hence gap converges to the conventional conic duality gap.

Let our accuracy ϵ be stored in OPTIONS.tol. Similar to the stopping criteria of SDPT3 [73], our code stops and returns an optimal solution and a certificate if

$$\max\{\text{relgap}, \text{Pfeas}, \text{Dfeas}, 1/\tau\} \leq \text{OPTIONS.tol}, \tag{B.20}$$

where

$$\text{relgap} := \frac{\text{gap}}{1 + |\text{gap}^0|}, \quad \text{Pfeas} := \frac{\|\frac{1}{\tau}z^0\|}{1 + \|Ax + \frac{1}{\tau}z^0\|}, \quad \text{Dfeas} := \frac{\|A^\top y/\tau + c\|}{1 + \|c\|}. \tag{B.21}$$

$\text{gap}^0 := \langle y^0, \Phi_*'(\vartheta y^0/\epsilon)\rangle + \epsilon$ is the initial gap. We can set OPTIONS.tol, otherwise it is equal to 10^{-8} by default.

For the infeasible case, we can prove that

$$y^I := \lim_{\tau \to (t_p(z^0))^-} y^I(\tau), \quad y^I(\tau) := \frac{y}{\|z^0 - Ax - \frac{1}{\tau}z^0\|_{\Phi''(Ax + \frac{1}{\tau}z^0)}}, \tag{B.22}$$

is a certificate of infeasibility that satisfies

$$\begin{aligned} &(1) \quad A^\top y^I = 0, \\ &(2) \quad \sup\{\langle y^I, z\rangle : z \in D\} < 0. \end{aligned} \tag{B.23}$$

In our code, we check the increase of τ and if $\frac{\Delta\tau}{\tau} \leq 0.001$, we examine to see if we have an approximate certificate for infeasibility or unboundedness. Our code returns infeasibility and y^I as its certificate if we have the following two conditions:

$$\begin{array}{ll} (1) & \left\|A^\top y^I\right\| \leq \text{OPTIONS.tol} \\ (2) & \langle y^I, \Phi'_*(\vartheta y^I/\epsilon)\rangle + \epsilon < 0. \end{array} \tag{B.24}$$

First we show that condition (2) in (B.24) reduces to the certificate of infeasibility for the conic formulation. Let us substitute (B.18) in (B.24)-(2), then we get

$$0 > \langle y^I, \Phi'_*(\vartheta y^I/\epsilon)\rangle + \epsilon = \langle y^I, b\rangle + \frac{\epsilon}{\vartheta}\langle \vartheta y^I/\epsilon, H'_*(\vartheta y^I/\epsilon)\rangle + \epsilon = \langle y^I, b\rangle - \epsilon + \epsilon = \langle y^I, b\rangle. \tag{B.25}$$

(B.25) and (B.24)-(1) form the conventional certificate of infeasibility for the conic optimization problem. Secondly, by having (B.24), one can easily check infeasibility. Assume that y^I satisfies (B.24) and there exists $Ax \in D$. Then we have

$$\begin{aligned} & \epsilon > \langle y^I, Ax - \Phi'_*(\vartheta y^I/\epsilon)\rangle \approx \langle y^I, -\Phi'_*(\vartheta y^I/\epsilon)\rangle \\ \Rightarrow \quad & \langle y^I, \Phi'_*(\vartheta y^I/\epsilon)\rangle + \epsilon > 0, \end{aligned} \tag{B.26}$$

which is a contradiction to (B.24)-(2).

On the other hand, if $\frac{\Delta\tau}{\tau} \leq 0.001$ and

$$\langle c, x\rangle > \frac{1}{\text{OPTIONS.tol}} \quad \text{and} \quad \frac{\|Ax + \frac{1}{\tau}z^0\|}{1 + \|\frac{1}{\tau}z^0\|} \geq \frac{1}{\text{OPTIONS.tol}}, \tag{B.27}$$

then our code stops and returns x as a certificate of unboundedness.