Technical University of Denmark

Presolving and regularization in mixed-integer second-order cone optimization

Friberg, Henrik Alsing; Stolpe, Mathias; Andersen, Kent Høj; Andersen, Erling D.; Dahl, Joachim

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

HENRIK ALSING FRIBERG

DTU

PRESOLVING AND REGULARIZATION

IN MIXED-INTEGER SECOND-ORDER CONE OPTIMIZATION

DTU Wind Energy
PhD Thesis 2016

DTU Wind Energy
Department of Wind Energy

COMPANY SUPERVISOR
Erling D. Andersen,
MOSEK ApS.

UNIVERSITY SUPERVISOR
Mathias Stolpe,
Department of Wind Energy,
Technical University of Denmark.

COMPANY CO-SUPERVISOR
Joachim Dahl,
MOSEK ApS.

THIRD PARTY SUPERVISOR
Kent Andersen,
Department of Mathematical Sciences,
University of Aarhus.

Technical University of Denmark
Department of Wind Energy
Frederiksborgvej 399, Building 118
DK-4000 Roskilde
Denmark
Phone: 46 77 50 85
Email: info@vindenergi.dtu.dk
www.vindenergi.dtu.dk

This PhD thesis was funded by MOSEK ApS and the Danish Ministry of Science, Innovation and Higher Education, and submitted to the Technical University of Denmark for the degree of Doctor of Philosophy on March 10, 2016. Members of the assessment committee were Prof. Tamás Terlaky from Lehigh University (USA), Gabor Pataki from the University of North Carolina (USA), and Martin S. Andersen from the Technical University of Denmark.

Abstract

Mixed-integer second-order cone optimization is a powerful mathematical framework capable of representing both *logical conditions* and *nonlinear relationships* in mathematical models of industrial optimization problems. What is more, solution methods are already part of many major commercial solvers including that of MOSEK [72] as well as XPRESS [31], GUROBI [46] and CPLEX [50]. This thesis concerns the performance and reliability of these solvers and makes two contributions; a theoretical one and a practical one.

In the theoretical part of the thesis a fundamental issue with reliability, affecting both continuous and mixed-integer conic optimization in general, is discovered and treated. This part of the thesis continues the studies of facial reduction preceding the work of Borwein and Wolkowicz [17] in 1981, when the first algorithmic cure for these kinds of reliability issues were formulated. An important distinction to make between continuous and mixed-integer optimization, however, is that the reliability issues occurring in mixed-integer optimization cannot be blamed on the practitioner's formulation of the problem. Specifically, as shown, the causes for these issues may well lie within the modifications to the formulation performed by the solution method itself. Hence, this calls for native support of facial reduction mechanisms within the commercial solvers to function reliably. In pursuit of such mechanisms, many fast and accurate heuristics are explored, supplementing the main discovery of this thesis that facial reduction can be interleaved with common optimization methods of high efficiency. Finally, a branch-and-bound method utilizing these mechanisms is established.

In the practical part of the thesis, a lack of consensus regarding basic definitions, representations and file formats is found, thereby increasing barriers for benchmarking with decreased market transparency as result. These differences are explored and results in the design of a new file format called *The Conic Benchmark Format (CBF)*. Unlike any other file format for conic optimization, this one is both cross-platform compatible, high performant and future-proof by encompassing other conic extensions. Scripts and tools have moreover been developed to aid parsing (resp. conversion) of the file format in service of software developers (resp. optimization practitioners), and are actively distributed. The functionality of all of this is proven not only by first-class citizenship in the modeling language PICOS [87], but also by *The Conic Benchmark Library (CBLIB)* where the conversion tools have been used to test its more than a thousand instances with MOSEK and CPLEX. This benchmark library was compiled as part of this thesis in support of studies in performance and reliability, but has yet to be used for the theoretical subjects of this thesis.

Danish abstract

Konisk kvadratisk blandet-heltalsoptimering er en stærk matematisk ramme der accepterer både *logiske begrænsninger* og *ikke-lineære relationer* i matematiske modeller af industrielle optimeringsproblemer. Hvad mere er, at løsningsmetoder allerede eksisterer i mange af de største kommercielle løsere i såvel MOSEK [72] som i XPRESS [31], GUROBI [46] og CPLEX [50]. Denne afhandling beskæftiger sig med ydeevnen og pålideligheden af disse løserer og giver to bidrag; et teoretisk og et praktisk.

I den teoretiske del af afhandlingen opdages og behandles et fundamentalt problem vedrørende pålidelighed, der påvirker både kontinuert- og blandet-heltalsoptimering mere generelt end det valgte fokusområde. Denne del af afhandlingen fortsætter studierne af faciale reduktioner der går forud for Borwein og Wolkowicz i 1981, hvor den første algoritmiske kur for denne type pålidelighedsproblemer blev formuleret. En vigtig forskel på kontinuert- og blandet-heltalsoptimering er imidlertid, at de pålidelighedsproblemer der måtte forekomme ikke uden videre kan bebrejdes den praktiserendes matematiske model i blandet-heltalsoptimering, fordi årsagen også kan findes i de ændringer i modellen der foretages af løsningsmetoden. Dette betyder at de kommercielle løserer er nødt til at indbygge understøttelse for faciale reduktionsmetoder for at fungere pålideligt. I jagten på sådanne metoder udvikles mange hurtige og præcise heuristikker der supplerer hovedresultatet; at faciale reduktionsmetoder kan integreres i effektive almenkendte løsningsmetoder. Denne del af afhandlingen sluttes af med at etablere en branch-and-bound metode baseret på disse mekanismer.

I den praktiske del af afhandlingen udforskes den manglende konsensus vedrørende basale definitioner, repræsentationer og filformater, der øger barriererne for benchmarking og derfor nedsætter gennemsigtigheden på markedet for løserer. Dette studie leder frem til udviklingen af et nyt filformat kaldet *The Conic Benchmark Format (CBF)*. I modsætning til andre filformater for konisk optimering, er dette designet til at være både platformsuafhængigt, højtydende og fremtidssikret ved at tillade andre koniske udvidelser. Scripts og værktøjer er desuden udviklet til at afhjælpe parsing (hhv. konvertering) af filformatet i servicering af softwareudvikleren (hhv. den praktitionerende), og distribueres aktivt. Funktionaliteten af alt dette bevises først og fremmest af filformatets fremtrædende rolle i modelleringssproget PICOS [87], men ses også af *The Conic Benchmark Library (CBLIB)* hvor konverteringsværktøjerne er brugt til at teste dets mere end et tusind probleminstanser med MOSEK og CPLEX. Dette koniske benchmark bibliotek blev udarbejdet som en del af denne afhandling til støtte af studier i ydeevne og pålidelighed, men er endnu ikke anvendt på de teoretiske emner i afhandlingen.

Thesis work

Thesis work carried out as part of this PhD project is given below and preprints are attached as appendices of this document. All citations to this work will be given the prefix Friberg, such as [Friberg 36], to make self-citations distinguishable.

[35] H. A. Friberg. Conic Benchmark Format - Version 1. Technical report, DTU Wind Energy, 2013. URL `http://orbit.dtu.dk/files/88492586/Conic_Benchmark_Format.pdf`.

[36] H. A. Friberg. CBLIB 2014: a benchmark library for conic mixed-integer and continuous optimization. *Mathematical Programming Computation*, 2015. URL `http://dx.doi.org/10.1007/s12532-015-0092-4`.

[81] F. Permenter, H. A. Friberg, and E. D. Andersen. Solving conic optimization problems via self-dual embedding and facial reduction: a unified approach. Technical report, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/09/5104.html`.

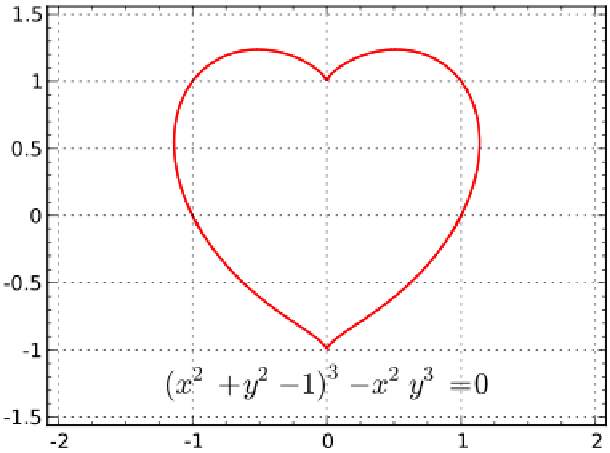
[38] H. A. Friberg. A relaxed-certificate facial reduction algorithm based on subspace intersection. *Optimization Online*, 2016. URL `http://www.optimization-online.org/DB_HTML/2016/01/5283.html`.

[39] H. A. Friberg. Facial reduction heuristics and the motivational example of mixed-integer conic optimization. *Optimization Online*, 2016. URL `http://www.optimization-online.org/DB_HTML/2016/02/5324.html`.

Note that the journal of [Friberg 36] made a mistake and inadvertently missed out an important paragraph during the typesetting process. The attached preprint is correct, however, and all references to [Friberg 36] hence implicitly assumes the following erratum to be in effect:

[37] H. A. Friberg. Erratum to: CBLIB 2014: a benchmark library for conic mixed-integer and continuous optimization. *Mathematical Programming Computation*, 2015. URL `http://dx.doi.org/10.1007/s12532-015-0098-y`.

Dedicated to my wife, kids, friends and family.




```
My mind forever your will share,
but in love of math does not compare.
```

Contents

Preprints of thesis work

1
Mixed-integer conic optimization

Whenever I have to describe what optimization means to me, I usually take the connection to real-world problems as my starting point. This connection is illustrated in Figure 1.1 and can be explained as follows. In many fields there are problems (or situations) involving complicated relationships or large amounts of data for which answers are sought. In order to quantify the problem, practitioners within these fields may attempt to construct a rigorous problem description quantifying the degrees of freedom over which we have control (the variables), how they are limited and how they affect the problem (the constraints), and finally what answers are sought (the objective). In the context of optimization, the objective may be to minimize or maximize any value function such as costs, material properties or emotional happiness as long as it can be quantified. The task of translating this description into a language (or mathematical model) that computers can understand and have solutions methods for, as well as judging and putting the computer output (or mathematical solution) into action, is known as *operations research*. Some may also know it as *prescriptive analytics*. The study of mathematical modeling techniques and development of solution methods for problems where a value function is to be minimized or maximized, is known as *mathematical optimization*.

In this thesis I study mathematical modeling with, and solution methods for, mixed-integer conic optimization problems defined as follows. For coefficients $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, conic sets $\mathcal{K}_i \subseteq \mathbb{R}^{m_i}$ for $i = 1, \ldots, r$, and an index set $\mathcal{I} \subseteq \{1, \ldots, n\}$ of integer variables, consider the mixed-integer conic optimization problem,

$$\begin{array}{lrcll} \underset{x}{\text{minimize}} & c^T x & & & \\ \text{subject to} & (Ax - b)_i & \in & \mathcal{K}_i, & \text{for all } i \in \{1, \ldots, r\}, \\ & x_j & \in & \mathbb{Z} \cap [l_j, u_j], & \text{for all } j \in \mathcal{I}, \end{array} \tag{1.1}$$

where $l_j, u_j \in \mathbb{Z} \cup \{-\infty, +\infty\}$ for all $j \in \mathcal{I}$ represent the possibly unbounded domain of integer variables. The formulation (1.1) generalizes from the computationally hard (i.e., NP-hard [52]) but, as argued in [15], yet highly successful special case of mixed-integer linear optimization.

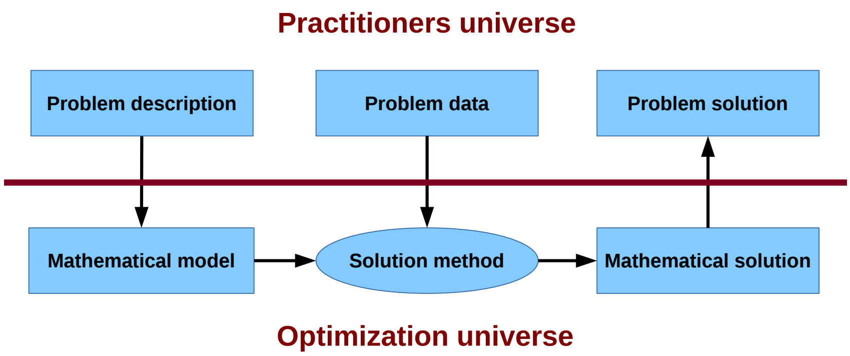


Figure 1.1: The flowchart that I used to disseminate the concept of mathematical optimization at the passed course in teaching and learning.

Here, in this special case, the cones $\mathcal{K}_i$ may be either one of the linear cones $\{0\}^{m_i}$, $\mathbb{R}_+^{m_i}$ or $\mathbb{R}_-^{m_i}$ such that each conic constraint $(Ax-b)_i \in \mathcal{K}_i$ describes a system of equations or inequalities.

Another special case of (1.1) is mixed-integer second-order cone optimization which is the prime target of analysis in this thesis. This case can be regarded as the first step towards nonlinearity in the formulation (1.1), as it allows a selection of quadratic terms in conic forms, i.e., the quadratic cone $\mathcal{Q}^n := \left\{x \in \mathbb{R}_+ \times \mathbb{R}^{n-1} : x_1^2 \geq \sum_{i=2}^n x_{2:n}^2\right\}$ and the rotated quadratic cone $\mathcal{Q}_r^n := \left\{x \in \mathbb{R}_+^2 \times \mathbb{R}^{n-2} : 2x_1x_2 \geq \sum_{i=3}^n x_{2:n}^2\right\}$.

Specializations like the above are important. In particular, while the generality of the conic form (1.1) allows it to capture a wide variety of nonlinear relationships, it is impossible to construct efficient solution methods without limiting the choice of cones. Fortunately, as recently presented in [60], 320 of the 333 problems in the mixed-integer general nonlinear benchmark library MINLPLIB2 [68], can be reformulated to the conic form (1.1) using only three different nonlinear cones. Specifically, in the interest of this thesis, 204 of these problems needed only the quadratic cone. Another interesting but rather technical result, shown under mild conditions in [12, *projective transformation*], is that perspective reformulations of second-order cone representable epigraphs are, themselves, second-order cone representable. This result is used to obtain (sometimes significantly) stronger conic relaxations of mixed-integer problems in [3] and [45], oftentimes without increasing the computational overhead of solving it.

To give the reader an idea of how to reformulate mathematical models into the conic form (1.1), the thesis offers two examples. One is the classical Markowitz portfolio optimization problem [65] whose reformulation is given in the introduction of [Friberg 36]. The other is the production planning problem [108] presented here.

Example 1. *An economic quantity (or production planning) problem with mathematical model as stated below, minimizes the cost of satisfying fixed annual demands d of managed items. All items of volumes a are stored at the same facility with capacity b, from where items are shipped out at a fixed rate. Orders of size x are placed regularly, to ensure that new items ship in when the stored quantity drops to zero. There is a unit cost c, and a per-order cost p, associated with every item. All parameters are members of $\mathbb{R}_+^n$ where n is the number of items.*

$$\begin{array}{ll} \underset{x}{\textit{minimize}} & c^T x + \sum_{j=1}^n p_j \frac{d_j}{x_j} \\ \textit{subject to} & a^T x \;\; \leq \;\; b, \\ & \quad\; x \;\; \in \;\; \mathbb{Z}_+^n. \end{array}$$

The problem appears in CBLIB 2014 [Friberg 36], in its conic formulation,

$$\begin{array}{llcll} \underset{x,y}{\textit{minimize}} & c^T x + q^T y & & & \\ \textit{subject to} & a^T x & \leq & b, & \\ & (x_j, y_j, \sqrt{2}) & \in & \mathcal{Q}_r^3, & \textit{for } j = 1,\ldots,n, \\ & x & \in & \mathbb{Z}_+^n, & \end{array}$$

where q is defined per item as $q_j = p_j d_j$.

The main contributions on the practical side of this thesis, towards handling mixed-integer second-order cone optimization problems and testing solvers, are summarized in the following subsections and nurses for mixed-integer conic optimization in general.

1.1 CBF: The conic benchmark format

CBF is a file format specialized to conic optimization, developed in this thesis with a technical reference manual as found in [Friberg 35]. The idea of such a file format is not new, however, and Imre Polik [82] did take quite a few steps towards designing it. His ideas were ambitious covering dense, sparse and low-rank representations, dictionaries for often used constructions, short-hand notation for repetitive data, and several experimental cones for which the best solvers are still proof of concepts. Although there are good reasons for all of these features, the project never materialized for some reason.

When my supervisors and I picked up the idea at the beginning of my thesis, we focused on making something simple and useful that would rapidly gain widespread acceptance (and on a longer time scale, widespread adoption). This led me to work on the following overall goals.

1. Design a simple structured format allowing people to define experimental extensions on their own in a clear and standardized way, and somehow try to maintain an official standard accepting features as they gained traction. One could imagine the newly started community driven CBLIB project [Friberg 36] to take the lead for such a standard.

2. Populate the format, and implement scripts and tools, to attain the functionality and usability needed to obsolete other file formats in use for conic optimization. Please see [Friberg 36] for an explanation of why these other file formations were dismissed.

The CBF file format started out as a keyword driven variant of the SDPA format [104] to allow for more freedom in the type of problem and cones. Yet, CBF was also a performance driven development with parsers in C, Python and MATLAB written and tested from very early on. Features of the CBF file format can be summarized as follows:

- All text-based (hence human readable and cross-platform compatible).

- Keyword driven.

- Sectioned: 1^{st} Format; 2^{nd} Problem structure; 3^{rd} Problem data.

- Simple but expressive: Affine maps into conic domains; four linear, two second-order and one semidefinite (the ones listed at the end of Section 2.1).

- Versatile: The embedded data-section change list can be used to formulate a sequence of closely related optimization problems where warmstart capabilities may be utilized.

- Compact: 50% smaller files one average than similar MPS [72, 50].

- High performant: Index-based lookup (as opposed to the name-based lookup in MPS) and guarantees that preallocations for efficient memory usage are possible.

In its mathematical nature, the CBF file format is very close to the general conic formulation found in Section 2.4, with a free choice of objective sense (maximize versus minimize). One clarification worth pointing out, however, regards the handling of matrix variable and inequalities. While the conic form (1.1), as well as the general formulation in Section 2.4, suggest that the semidefinite cone of symmetric matrices need to be vectorized, the CBF file format continues practices from the SDPA format. Hence a special notation with (row,column)-indexed matrices are used as elaborated in both the technical reference manual [Friberg 35] and in CBLIB 2014 [Friberg 36]. This indentation seems to be common practice among semidefinite solvers, and avoids unnecessary complications including the otherwise required consensus for whether to use full vectorization, half vectorization or symmetric vectorization (half vectorization where off-diagonal entries are multiplied by $\sqrt{2}$ to preserve dot products).

Finally, a converter tool was developed in supplement of the C, Python and MATLAB parsers, in acceptance of the (at least initial) lack of support for the CBF format. As illustrated in Figure 1.2, this tool features frontends to the CBF parser written in C, and to the MOSEK C API such that all file formats supported by MOSEK can be read. Other frontends can easily

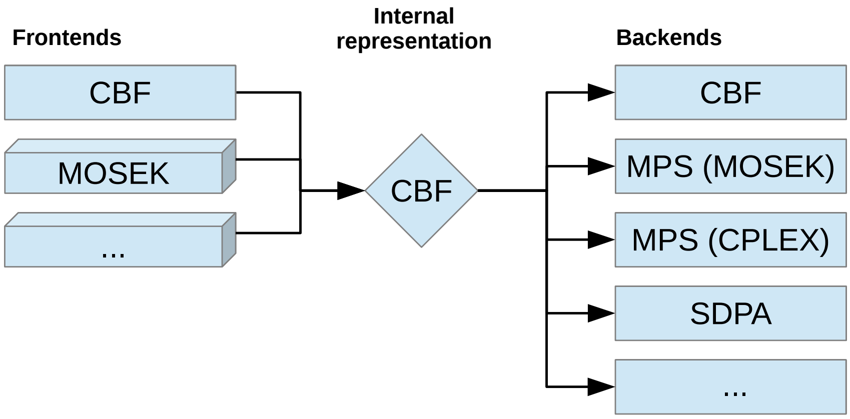


Figure 1.2: Layout of `cbftool`, a converter tool for the CBF file format distributed as part of the CBLIB base of scripts and tools.

be added. Backends have been implemented for the MPS format in the two second-order cone extensions introduced by MOSEK and CPLEX respective, as well as for the SDPA format albeit with no attempts to represent non-semidefinite cones at the moment. A MATLAB-implemented backend for the SEDUMI format [92] is furthermore found alongside the MATLAB parser in the experimental and unfinished cone solver at `https://github.com/HFriberg/micoso-solver`. The other parsers and the converter tool are distributed as *CBLIB tools* at `http://cblib.zib.de`.

1.2 CBLIB: The conic benchmark library

CBLIB is a benchmark library specialized to conic optimization, developed in this thesis with a continuously updated version at `http://cblib.zib.de` and a frozen instance selection called CBLIB 2014 presented in [Friberg 36] (see also the erratum [Friberg 37]).

This library was, and continue to be, an attempt to replicate the momentum previous caused by the benchmark libraries *NETLIB LP* [41] for linear optimization and *MIPLIB* [53] for mixed-integer linear optimization. In particular, from a practical point of view, benchmark libraries like these help to identify bugs, numerical instabilities and performance bottlenecks. By furthermore embracing instances from real or realistic applications, commonly appearing anomalies worth exploiting can be studied as opposed to if the library had been made up of randomly generated instances. An application-driven benchmark library can hence be said to act towards closing the gap between solver capability and industrial demand as captured by Figure 1.3 used on a conference to promote CBLIB.

The unfortunate side-effect of a purely application-driven benchmark library is that subtle edge-cases are rarely represented. Hence, to support theoretical studies of conic optimization it is

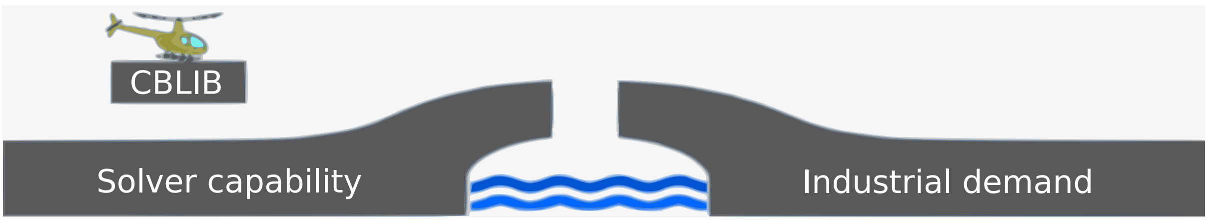


Figure 1.3: Bringing in process. Drawing from the CBLIB poster presented at the 2013 Mixed Integer Programming Workshop, hosted by the University of Wisconsin Madison and the Wisconsin Institute for Discovery.

just as important to embrace instances constructed with the sole purpose of exposing some particular property. These instances are marked by the tag `(academic)`, but CBLIB has not yet been populated actively with instances from this category and only feature one instance pack with this tag which was also part of the CBLIB 2014 selection [Friberg 36]. In contrast, the continuously updated version of CBLIB at `http://cblib.zib.de` currently, as of March 2016, has more than a thousand instances from 17 instance packs representing different applications of mixed-integer and continuous second-order cone optimization. Applications of other conic extensions, notably the semidefinite cone supported by the CBF file format have not yet been gathered actively.

1.3 Reliability in solutions methods of high efficiency

To describe how this thesis has contributed to the subject of reliability in solutions methods of high efficiency, a little background is required. To begin, note that a premise for solving mixed-integer conic optimization problems of the form (1.1) in general, is that we can solve its continuous relaxation. That is, the special case of (1.1) for which there are no integer variables, or in other words, $\mathcal{I} = \emptyset$. There are many possibilities for solving this problem and a few honorable mentions, applicable to the special case of second-order cone optimization, are given as follows.

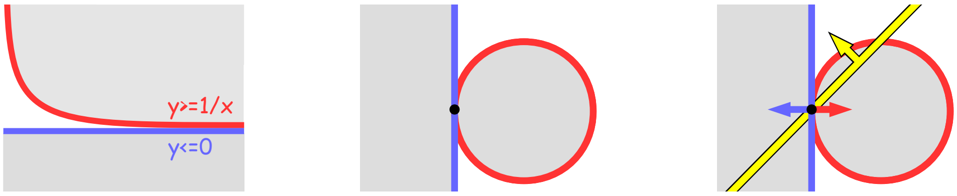
1. Active set methods iterating like the simplex method between vertices of a polyhedron are described in [73] and [29]. The former obtains the polyhedron by restricting attention to formulations in which second-order cones only affect the objective function, and the latter uses Lagrangian relaxation of a single second-order cone. A more general active set approach is found in [78] with preliminary results presented in [55].

2. Cylindrical algebraic decomposition methods are described in [5, 6, 51] and generalizes from Fourier–Motzkin elimination. While the latter eliminates variables one-by-one from a system of linear inequalities, the former does it for a system of polynomial inequalities. The cylindrical algebraic decomposition can hence be specialized to the quadratic inequalities from second-order cones, but has doubly-exponential worst case running time.

Despite many alternatives, the perhaps most successful solution method for second-order cone optimization is given by primal-dual interior-point methods. These methods are efficient in the special case of linear optimization [63], and this efficiency is maintained for so-called symmetric (previously known as self-scaled) cones as shown in [76, 96, 70]. Since second-order cones are a special case of symmetric cones, as elaborated in Section 2.1, these efficiency results apply to second-order cone optimization as well.

A variant of the primal-dual interior-point method, defined on the self-dual homogeneous embedding of the problem to be solved, has the same attractive properties but is known to give more stability in the detection of infeasible problems [103, 8]. As shown in [Friberg 81], the self-dual homogeneous embedding strategy also gives more stability in the detection of so-called *ill-posed problems*.

The term ill-posed comes from Renegar [85, 34] and means that the tiniest perturbation of coefficients will alter the feasibility status. A weakly infeasible problem is thus arbitrarily close to feasibility (Figure 1.4a), while a weakly feasible problem is arbitrarily close to infeasibility (Figure 1.4b). These cases are challenging not only in practice, but also in theory. In particular, a weakly infeasible problem has no dual improving ray [62] (the usual certificate of infeasibility), and a weakly feasible problem has no generalized Slater point [18] (the usual constraint qualification) whereby the KKT conditions may fail to verify optimal solutions depending on the objective function. Specifically, in Figure 1.4c, there is only one feasible point and hence exactly one optimal solution. Nevertheless, is can be confirmed that there is no way to satisfy the KKT



(a) Weak infeasibility. (b) Weak feasibility. (c) Failure of the KKT conditions in the weakly infeasible example.

Figure 1.4: Examples and consequences of ill-posedness.

stationarity condition at this point, as no weighting of normal vectors (blue and red arrow) can counterbalance the gradient direction of objective improvements (yellow arrow). Finally, as shown by [Friberg 39, Example 1], the troublesome blue cutting planes of Figure 1.4 may be added naturally in the process of solving mixed-integer conic problems. This shows that ill-posed problems are a naturally occurring phenomenon for which algorithmic countermeasures should be taken to ensure reliability.

The term *ill-posed* was also used in [8] to describe the situation when the limit point of the primal-dual interior point method, applied to the self-dual embedding, could not be interpreted. With the contributions of [Friberg 81], we now know that these limit points can actually be interpreted as so-called facial reduction certificates proving ill-posedness in the original sense of Renegar [85]. Facial reduction certificates do more than that, however, as they also hold the key to escape the state of ill-posedness. In particular, they reveal reformulations of the problem which act to regularize it such that the optimal value can be found and certified computationally.

The facial reduction certificates interpreted from the limit point of the primal-dual interior point method, applied to the self-dual embedding, has especially attractive properties. These properties, and how they can be used to solve conic optimization problems reliably, are elaborated in Chapter 3 and summarizes the main conclusions to be drawn from [Friberg 81]. This acts both as a recap and need-to-know crash course, to prepare the reader for the unpublished results of Chapter 4. In particular, Chapter 4 shows how the reliable solution method for continuous conic optimization problems from Chapter 3 can be used in a reliable branch-and-bound solution method for mixed-integer conic optimization problems.

Finally, while these reliable solution methods work in theory, the limit points revealing the facial reduction certificates are not computed exactly in practice. In order words, the regularizing reformulations applied to the problem are inaccurate. Looking back at Figure 1.4 and recalling that the tiniest perturbation of coefficients will alter the feasibility status, these inaccuracies can have quite an impact on the final result. In response to this issue, I made two additional studies within this thesis project. One is in [Friberg 38], where the requirements for a facial reduction certificate is relaxed such that all reductions needed can be found analytically. The issue here turned out to be representational. The other study is in [Friberg 39] and surveys existing methods, as well as develops many new ones, able to find facial reductions heuristically by various analytical techniques. Neither of these studies completely solved the problem of inaccurate facial reduction certificates, however, and the issue remains an open research direction.

2
Theoretical introduction

This chapter introduces the concepts and properties with special interest to the work and conclusions of this thesis. Additional care is taken to explain the common differences in definitions and representations of cones and conic constraints respectively. Finally, the chapter ends with two pedagogical remarks. One on dualization by hand, which came as a side-product of designing the conic benchmark format covered in Section 1.1, and one on the naming convention for the entries of conic domains.

2.1 Cones

A *quadratic cone* is defined by

$$\mathcal{Q}^n := \{x \in \mathbb{R}^n : x_1 \geq \|x_{2:n}\|_2\}\,, \tag{2.1}$$

and can be thought of as an infinitely high ice cream cone (see the front page). Geometrically, $x_{2:n}$ holds the set of points in a hyperball with radius x_1. Linguistically, however, I propose calling x_1 the *head* and $x_{2:n}$ the *base* of the quadratic cone, as the concepts of a head and base can be generalized to others cones as shown in Section 2.5. The boundary of the quadratic cone is $\operatorname{bnd}\mathcal{Q}^n = \{x \in \mathbb{R}^n : x_1 = \|x_{2:n}\|_2\}$, and the interior is $\operatorname{int}\mathcal{Q}^n = \{x \in \mathbb{R}^n : x_1 > \|x_{2:n}\|_2\}$. In these terms, and in the important notion of a dual cone, $\mathcal{K}^* := \{y \in \mathbb{R}^n : y^T x \geq 0,\ \ \forall x \in \mathcal{K}\}$, many useful properties can be proven for the quadratic cone.

Proposition 1. *The quadratic cone $\mathcal{Q}^n$ has the following properties:*

1. *Non-empty:* $\mathcal{K} \neq \emptyset$.
2. *Closed:* $\mathcal{K} = \operatorname{cl}\mathcal{K}$.
3. *Convex:* $\lambda x + (1-\lambda)y \in \mathcal{K}$ *for all* $x, y \in \mathcal{K}$ *and* $0 \leq \lambda \leq 1$.
4. *Cone:* $\lambda x \in \mathcal{K}$ *for all* $x \in \mathcal{K}$ *and* $\lambda > 0$.
5. *Solid:* $\operatorname{int}\mathcal{K} \neq \emptyset$.
6. *Pointed:* $\mathcal{K} \cap (-\mathcal{K}) = \{0\}^n$.
7. *Self-dual:* $\mathcal{K} = \mathcal{K}^*$.
8. *Homogeneous: For any* $x, y \in \operatorname{int}\mathcal{K} \subseteq \mathbb{R}^n$, *there exists a mapping* $B \in \mathbb{R}^{n\times n}$ *such that* $B\mathcal{K} = \mathcal{K}$ *and* $Bx = s$.

Proof. Self-duality is shown by $x^T y = x_1 y_1 + x_{2:n}^T y_{2:n} \geq x_1 y_1 - \|x_{2:n}\|_2 \|y_{2:n}\|_2 \geq 0$ for $x, y \in \mathcal{Q}^n$ using the the Cauchy–Schwarz inequality, while $x^T y = \|x_{2:n}\|_2 (x_1 - \|x_{2:n}\|_2) < 0$ for $x \notin \mathcal{Q}^n$ when selecting $y = (\|x_{2:n}\|_2, -x) \in \mathcal{Q}^n$. Hence, as $\mathcal{K} = (\mathcal{K}^*)^*$, the bipolar theorem [86] shows properties 1-4. The point $\hat{x} = (1, 0, \ldots, 0)^T \in \operatorname{int}\mathcal{Q}^n$ shows solidness, and hence pointedness follows by self-duality and [30, Proposition 2.4.3]. Homogeneousity is constructively shown in [76, Theorem 3.1(iii) and Theorem 3.2, using definitions from Section 2.3]. $\square$

Any subset of properties from Proposition 1 define a category of cones with similar properties, and the results of this thesis are generalized, whenever possible, to one of these broader categories.

Non-empty, closed convex cones is the category of cones satisfying the minimal subset of properties considered in this thesis. These cones satisfy $\mathcal{C} = (\mathcal{C}^*)^*$, as shown by the bipolar theorem [86], which is needed to define primal-dual pairs of problems (that is, two conic optimization problems where the Lagrange-dual problem [19] of either gives you the other).

Note that the *the conical hull* of any set $S \subseteq \mathbb{R}^n$, denoted $\operatorname{cone}(S)$, is the union of the origin and all finite conical combinations, $\lambda_1 s_1 + \ldots + \lambda_k s_k$ for $\lambda \in \mathbb{R}^k_{++}$, of points $s_1, \ldots, s_k \in S$. By [86, Corollary 2.6.2], the conical hull $\operatorname{cone}(S)$ is the smallest non-empty, convex cone which contains the set S and the origin. Hence, its closure, $\operatorname{cl}\operatorname{cone}(S)$, belongs to the category above as it is the smallest non-empty, closed, convex cone which contains the set S.

Polyhedral cones are conical hulls, $\operatorname{cone}(S)$, of any finite set S of so-called extreme rays. By definition, these are always non-empty, closed, convex cones [86, Theorem 19.1].

Proper cones are defined to satisfy the properties 1-6 of Proposition 1 (see, e.g. [19]). This category is sometimes also known as regular cones, e.g., in [74, 57], but this name is discouraged as it may erroneously be perceived as the output of a cone regularization procedure (this concept is introduced in Section 2.3.3).

Symmetric cones are defined to be self-dual and homogeneous, and can be shown to satisfy all properties 1-8 of Proposition 1 (see, e.g., [83, 32]).

The important concept of *faces* for polyhedra (extreme points, facets, etc.) can be generalized to cones, or in fact to any convex set $\mathcal{S} \subseteq \mathbb{R}^N$. In particular, faces of convex sets $\mathcal{S}$, denoted $\mathcal{F} \trianglelefteq \mathcal{S}$, are convex subsets $\mathcal{F} \subseteq \mathcal{S}$ for which any line segment in $\mathcal{S}$, with a midpoint in $\mathcal{F}$, has both endpoints in $\mathcal{F}$ [86]. Faces satisfy the following property.

Proposition 2. *Faces of a convex set $\mathcal{F} \trianglelefteq \mathcal{S} \subseteq \mathbb{R}^n$, are convex by definition and satisfy the following properties.*

1. *$\mathcal{F}$ is closed if $\mathcal{S}$ is closed.*
2. *$\mathcal{F}$ is a cone if $\mathcal{S}$ is a cone.*

Proof. Statement 1 is from [86, Corollary 18.1.1]. For statement 2, let $\hat{x} \in \mathcal{F} \setminus \{0\} \subseteq \mathcal{S}$ and suppose $\mathcal{S}$ is a cone. Then any point $\hat{\lambda}\hat{x}$ on the ray $\{\lambda\hat{x} : \lambda > 0\} \subseteq \mathcal{S}$ belongs to $\mathcal{F}$, since $\hat{\lambda}\hat{x}$ for $\hat{\lambda} \neq 1$ is the endpoint of the line segment connecting $\hat{\lambda}\hat{x}$ and $\hat{\lambda}^{-1}\hat{x}$ with $\hat{x} \in \mathcal{F}$ as a midpoint. $\square$

Two types of faces are now elaborated. A *proper face* of a convex set $\mathcal{S}$ is a face which is non-empty and not equal to $\mathcal{S}$. An *exposed face* of $\mathcal{S}$ is a face that appears as the optimal set of $\mathcal{S}$ for some maximized linear functional. These two types of faces can also be characterized in greater detail for non-empty, closed, convex cones $\mathcal{C}$.

Let $z^\perp := \{x \in \mathbb{R}^n : x^T z = 0\}$. For $z \in \mathcal{C}^*$, the intersection $\mathcal{C} \cap z^\perp$ contains the origin and is an exposed face of $\mathcal{C}$ as it maximizes $-z^T x$ over $x \in \mathcal{C}$ [86]. Hence, if $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$, the intersection $\mathcal{C} \cap z^\perp$ cannot equal $\mathcal{C}$ and must represent a proper face of $\mathcal{C}$. The set $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$ can also be represented as the subset of points $z \in \mathcal{C}^*$ having a positive inner product with a relative interior point of $\mathcal{C}$ as shown in [80, Corollary 1]. This technical result is used a few times throughout the thesis and is therefore repeated here under a simpler proof.

Lemma 1. *Let $\mathcal{C}$ be a non-empty, closed, convex cone. Given $z \in \mathcal{C}^*$ and $\hat{p} \in \operatorname{relint}\mathcal{C}$, the following statements hold.*

1. *$\hat{p}^T z = 0 \iff z \in \mathcal{C}^\perp$;*
2. *$\hat{p}^T z > 0 \iff z \notin \mathcal{C}^\perp$.*

Proof. Statement 1. If $z \in \mathcal{C}^{\perp}$, then $p^T z = 0$ by definition. The claim thus follows if $\hat{p}^T z = 0$ implies $z \in \mathcal{C}^{\perp}$, that is, points $\bar{p}$ in $\mathcal{C}$ with $\bar{p}^T z \neq 0$ (or equivalently, $\bar{p}^T z > 0$ by dual cones) are forbidden. This holds since the existence of such points would imply $(\hat{p} - \alpha\bar{p})^T z < 0$ for any $\alpha > 0$, and hence $\hat{p} - \alpha\bar{p} \notin \mathcal{C}$ by dual cones; a contradiction of $\hat{p} \in \operatorname{relint} \mathcal{C}$ by the prolongation lemma, [14, Proposition 1.3.3]. Statement 2 is simply the contrapositive of statement 1. □

Another important concept is that of the *Cartesian product*, $\times$, which is used to glue cones together in satisfaction of

$$x \in \mathcal{K}_x,\ y \in \mathcal{K}_y \quad \iff \quad \begin{bmatrix} x \\ y \end{bmatrix} \in \mathcal{K}_x \times \mathcal{K}_y,$$

for column vectors and

$$X \in \mathcal{K}_X,\ Y \in \mathcal{K}_Y \quad \iff \quad \begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix} \in \mathcal{K}_X \times \mathcal{K}_Y,$$

for matrices. A cone which is not the Cartesian product of smaller cones is said to be *primitive.* A non-primitive cone, such as $\mathbb{R}^2_+ = \mathbb{R}_+ \times \mathbb{R}_+$, has the properties of Proposition 1 shared by its Cartesian factors. Hence, the Cartesian product can, and often is, used to write the conic formulation (1.1) in only one conic constraint without loss of generality.

Finally, I end this section with a formal introduction to the family of cones which appear in the examples of this thesis, with a side note on a common difference to these definitions.

Linear cones This is the subset of polyhedral cones used as constraint and variable domains in general linear optimization. I consider this family to contain the set of zeros $\{0\}^n$, the nonnegative orthant $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x_j \geq 0 \text{ for } j = 1, \ldots, n\}$, the nonpositive orthant $\mathbb{R}^n_- = -\mathbb{R}^n_+$, and the set of reals $\mathbb{R}^n$ used as the domain of free variables. The nonnegative and nonpositive orthants are both symmetric cones [102]. The set of reals and of zeros are both non-empty, closed, convex cones, the former solid, the latter pointed, and they are each others dual cone; $(\mathbb{R}^n)^* = \{0\}^n$.

Second-order cones This family is also known as the quadratic cones, nicknamed the ice cream cones, and often called the Lorentz cones in honor of the physicist Hendrik Lorentz and his work on the similarly defined light cone. I consider this family to contain two members, namely the quadratic cone $\mathcal{Q}^{1+n} = \{(r, x) \in \mathbb{R}^1_+ \times \mathbb{R}^n \mid r^2 \geq x^T x\}$ and the rotated quadratic cone $\mathcal{Q}^{2+n}_r = \{(r, x) \in \mathbb{R}^2_+ \times \mathbb{R}^n \mid 2r_1 r_2 \geq x^T x\}$ related by $\mathcal{Q}_r = T\mathcal{Q}$ and $T\mathcal{Q}_r = \mathcal{Q}$ for the symmetric and orthogonal matrix

$$T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \\ 0 & 0 & \mathrm{I} \end{bmatrix}. \tag{2.2}$$

The matrix T is chosen for its properties but is notably not a rotation matrix. The rotated quadratic cone still is a rotation of $\mathcal{Q}$, however, as shown by the rotation matrix at $\theta = \frac{\pi}{4}$ which satisfy $\mathcal{Q}_r = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & \mathrm{I} \end{bmatrix} \mathcal{Q}$. Specifically, this relation can be shown by swapping the first two rows of T corresponding to a permutation within the (r_1, r_2) variable symmetry in the definition of $\mathcal{Q}_r$. Both cones, $\mathcal{Q}$ and $\mathcal{Q}_r$, are symmetric; the former by Proposition 1 and the latter because neither inner products (related to self-duality) nor the property of being a homogeneous cone are affected by rotations.

Finally, on a side note, the rotated quadratic cone is often encountered without the factor 2 in front of $r_1 r_2$. This cone is sometimes easier for practitioners, with respect to mathematical modeling, and has a long history dating back to 1957 as a special case of the power cone introduced in [54]. The practice of calling it a rotated quadratic cone is far more recent, however, and somewhat confusing. In particular, in contrast to common belief (without naming and shaming), this cone it not a rotation of $\mathcal{Q}$ and is thus, perhaps, better denoted by another name; possibilities include the *quadratic power cone*, the *restricted hyperbolic* as used in [58], or something like *the squeeze-rotated quadratic cone*. When using this cone in mathematical modeling rather than $\mathcal{Q}_r$, it should also be noted that this cone is not self-dual in the standard inner product. Hence, for software developers, this means that abstractions of the inner product space, or an extra cone definition, is needed represent the Lagrange-dual problem, e.g., in modeling languages such as PICOS [87]. This offers at least some explanation for why $\mathcal{Q}_r$, rather than this cone, is the right choice for the CBF file format [Friberg 35].

Semidefinite cones This family refers to the real-valued symmetric positive semidefinite cone (often just denoted the semidefinite cone) $\mathcal{S}^n_+ = \left\{VV^T : V \in \mathbb{R}^{N\times N}\right\} = \{X \in \mathcal{S}^n \mid \lambda(X) \in \mathbb{R}^n_+\}$, where λ is the eigenvalue function and $\mathcal{S}^n$ is the set of n-by-n symmetric matrices. The semidefinite cone is a symmetric cone [102].

2.2 Conic constraints

A linear or affine conic constraint (often just denoted a conic constraint) is an affine map constrained to the domain of a cone, that is

$$Dx - d = \left(D,\, d \right) \left(\begin{smallmatrix} x \\ -1 \end{smallmatrix} \right) \in \mathcal{K}, \tag{2.3}$$

where $D \in \mathbb{R}^{p\times n}$ and $d \in \mathbb{R}^p$ are the coefficients of the affine entries and $\mathcal{K} \subseteq \mathbb{R}^p$ is the conic domain. For the choice $\mathcal{K} = \mathcal{Q}^p$, this is called a *second-order cone constraint*. Just like a system of equations, $\mathcal{K} = \{0\}^p$, it turns out that full row rank of $\left(D,\, d \right)$ can be assumed (at least in theory) for $\mathcal{K} = \mathcal{Q}^p$ without loss of generality.

Proposition 3. *The second-order cone constraint in dimension* $p \geq 1$,

$$Dx - d = \left(D,\, d \right) \left(\begin{smallmatrix} x \\ -1 \end{smallmatrix} \right) \in \mathcal{Q}^p, \tag{2.4}$$

for some $D \in \mathbb{R}^{p\times n}$ *and* $d \in \mathbb{R}^p$*, can be reformulated to a second-order cone constraint in dimension* $r = \operatorname{rank}\left(D,\, d \right)$.

Proof. Omitted with reference to [Friberg 38]. □

Some optimization software does not support conic constraints directly and require slack variables to be added. That would be $Dx - d = s$ and $s \in \mathcal{Q}^p$ to represent (2.3). The second-order cone constraint is sometimes also required in its quadratic-linear representation with slack:

$$s = Dx - d, \quad s_1^2 \geq \sum_{j=2}^{n} s_j^2, \quad s_1 \geq 0. \tag{2.5}$$

Optimization software accepting this quadratic-linear representation needs to take a little extra care to avoid loss of dual information as $s \in \mathcal{Q}^p$ (with p dual variables) becomes a linear and quadratic inequality usually having only one dual variable each. The representation (2.5) leads to an interesting question, however, of whether the slack variable can be dropped. In particular, with $D = \left(\begin{smallmatrix} \alpha^T \\ A \end{smallmatrix} \right)$ and $d = \left(\begin{smallmatrix} \beta \\ b \end{smallmatrix} \right)$, the quadratic part of (2.5) can be expanded to the form of a standard quadratic inequality,

$$x^T Q x + q^T x + p \leq 0, \tag{2.6}$$

where $Q = A^T A - \alpha\alpha^T$. That is, the quadratic coefficient matrix Q of (2.6) is a rank-one updated positive semidefinite matrix which can have up to one negative eigenvalue [95]. Due to dependencies between Q, q and p in this expansion, however, the other direction from (2.6) to (2.5) is not always possible.

Mahajan and Munson [64] clarifies this relationship. In particular, necessary and sufficient conditions are proven under which (2.6) can be reformulated as quadratic part of (2.5) without slack, i.e., $(\alpha^T x - \beta)^2 \geq \|Ax - b\|_2^2$. To match the linear part of (2.5) without slack, i.e., $\alpha^T x - \beta \geq 0$, the paper suggests that the two possible domains of $\alpha^T x - \beta$ are treated separately, that is, $\alpha^T x - \beta \geq 0$ and $\alpha^T x - \beta \leq 0$, respectively. Both cases leads to a second-order cone constraint and this procedure hence works by growing a binary tree of subproblems to consider, doubling in size with each reformulation. The following proposition characterize when $\alpha^T x - \beta \geq 0$, respectively $\alpha^T x - \beta \leq 0$, contains the entire feasible set of the considered optimization problem such that subproblem partitioning can be avoided.

Proposition 4. *Suppose $(\alpha^T x - \beta)^2 \geq \|Ax - b\|_2^2$ is satisfied. Then*

1. $\binom{\alpha^T x - \beta}{Ax - b} \in \mathcal{Q}^p \iff \lambda^T \binom{\alpha^T x - \beta}{Ax - b} \geq 0$ *is a valid inequality for some* $\lambda \in \operatorname{int} \mathcal{Q}^p$.

2. $\binom{-(\alpha^T x - \beta)}{Ax - b} \in \mathcal{Q}^p \iff \lambda^T \binom{\alpha^T x - \beta}{Ax - b} \geq 0$ *is a valid inequality for some* $\lambda \in -\operatorname{int} \mathcal{Q}^p$.

Proof. Let $s = \binom{s_1}{s'} = \binom{\alpha^T x - \beta}{Ax - b}$ and consider statement 1. If $s \in \mathcal{Q}^p$, then $\lambda^T s \geq 0$ for all $\lambda \in \operatorname{int} \mathcal{Q}^p$ by self-duality of the quadratic cone. Conversely, if $\lambda^T s \geq 0$ for some $\lambda = \binom{\lambda_1}{\lambda'} \in \operatorname{int} \mathcal{Q}^p$, we need to show $s_1 \geq 0$ to fulfill the claim via the quadratic-linear form (2.5). If $s_1 = 0$ we are done. Otherwise,

$$0 \leq \lambda^T s = \lambda_1 s_1 + (\lambda')^T (s') \leq \lambda_1 s_1 + \|\lambda'\|_2 \|s'\|_2 < \lambda_1 s_1 + \lambda_1 |s_1|,$$

firstly by the Cauchy-Schwarz inequality and secondly by assumption; $\lambda \in \operatorname{int} \mathcal{Q}^p$ and $s_1^2 \geq \|s'\|_2^2$. Since $\lambda_1 > 0$, we find that $|s_1| + s_1 > 0$ and hence $s_1 > 0$. Statement 2 follows by statement 1. □

Searching for valid inequalities of the type required by Proposition 4 is difficult in practice. Hence, it is generally nontrivial to recognize second-order cone constraints in the quadratic-linear representation when the quadratic part is specified, without slack, as a standard quadratic inequality (2.6). Moreover, by [64], such a recognition would require an eigenvalue decomposition of the quadratic coefficient matrix Q of (2.6) and hence be subject to numerical difficulties.

Finally, the second-order cone constraint is sometimes required as a semidefinite constraint, e.g., to be dealt with in pure semidefinite optimization (e.g., for addition to SDPLIB [16]) or when using SDPA [104] as a backend to SCIP-SDP [67]). All reformulations I have seen follow Alizadeh and Goldfarb [4], and use that $x \in \mathcal{Q}^n$ if and only if $\operatorname{arrow}(x) \succeq 0$, in terms of the arrowhead matrix,

$$\operatorname{arrow}(x) = \begin{pmatrix} x_1 & x_{2:n}^T \\ x_{2:n} & x_1 I \end{pmatrix}.$$

I will now show that the dimension of the semidefinite matrix representation can be reduced by one. This has computational advantages as solving a semidefinite problem requires at least $O(n^3)$ arithmetic operations to compute the Cholesky factorization on each $n \times n$ block [40]. By reducing the dimension n by one, we thus avoid $O(n^2)$ arithmetic operations for each represented second-order cone.

Proposition 5. *$x \in \mathcal{Q}^n$ if and only if $arw(x) \succeq 0$, where*

$$arw(x) = \begin{pmatrix} x_1 + x_2 & x_{3:n}^T \\ x_{3:n} & (x_1 - x_2) I \end{pmatrix}.$$

Proof. The eigenvalues of $\mathrm{arw}(x)$ is given by $x_1 - \|x_{2:n}\|_2$, $x_1 + \|x_{2:n}\|_2$ and $x_1 - x_2$, the latter with multiplicity $n-2$ [77]. Hence, $\mathrm{arw}(x) \succeq 0$ if and only if $x_1 + \|x_{2:n}\|_2 \geq x_1 - x_2 \geq x_1 - \|x_{2:n}\|_2 \geq 0$ showing the claim. □

This lower-dimensional representation can also be realized for the rotated quadratic cone.

Proposition 6. *$x \in \mathcal{Q}_r^n$ if and only if $arw(Tx) \succeq 0$, where*

$$arw(Tx) = \begin{pmatrix} \sqrt{2}x_1 & x_{3:n}^T \\ x_{3:n} & \sqrt{2}x_2 I \end{pmatrix},$$

and T is the symmetric and orthogonal matrix from (2.2)

Proof. Follows by inspection given Proposition 5, noting that $Tx \in \mathcal{Q}$ if and only if $x \in \mathcal{Q}_r$. □

At last, I end this section by mentioning that the adjective *second-order cone representable* is used for nonlinear relationships that can be reformulated using one or more second-order cone constraints. This is studied in [58, 12, 4] and progress is still being made as now exemplified.

Example 2. *The geometric mean is defined by*

$$\{x \in \mathbb{R} \times \mathbb{R}_+^{m-1} \mid |x_1|^{\beta_1} \leq x_2^{\beta_2} \cdots x_m^{\beta_m}\}, \tag{2.7}$$

with integer powers $\beta \in \mathbb{Z}^m$ satisfying $\beta_1 = \sum_{j=2}^m \beta_j$. The geometric mean is second-order cone representable as shown by the "tower of variables"-type of construction in [4]. The number of second-order cones needed in this construction was recently reduced in [71], however, showing an economic formulation for all geometric cones of dimension $m \leq 4$. These results apply to other popularized cones.

1. *The rational power cone is defined by*

$$\{(y, x) \in \mathbb{R} \times \mathbb{R}_+^{m-1} \mid |y| \leq x_1^{\alpha_1} \cdots x_{m-1}^{\alpha_{m-1}}\}, \tag{2.8}$$

for rational powers $\alpha \in \mathbb{Q}^{m-1}$ summing to $e^T\alpha = 1$. Suppose $\alpha_j = p_j/q_j$ and define $P = \gcd(p_1, p_2, \ldots)$ and $Q = \mathrm{lcm}(q_1, q_2, \ldots)$. Then (2.8) *can be rewritten as a geometric mean with integer vector $\beta = \frac{Q}{P}(1, \alpha)$.*

2. *The rational p-order cone is defined by*

$$\{(y, x) \in \mathbb{R}_+ \times \mathbb{R}^{m-1} \mid y \geq \|x\|_\alpha\}, \tag{2.9}$$

for rational $\alpha \geq 1$, and can be formulated using $m-1$ four-dimensional geometric means as shown by the constructions in [4, 71].

2.3 Primal-dual pairs of problems

The continuous special case of (1.1) with no integer variables, $\mathcal{I} = \emptyset$, is given by

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & (Ax - b)_i \;\in\; \mathcal{K}_i, \quad \text{for all } i \in \{1, \ldots, r\}. \end{array} \tag{2.10}$$

The *feasible set* is the common denotation for the (possibly empty) set of points x that satisfy all conic constraints. On the feasible set of (2.10), the *objective* $c^T x$ and the *minimize* statement defines a comparison operator for judging points against each other, as well as an intent to find good feasible solutions with respect to this metric, e.g., using trial and error by hand.

We can improve our understanding of the conic optimization problem (2.10) significantly, however, by considering properties such as the infimum value of $c^T x$ over the feasible set, defined as $-\infty$ if the value can be arbitrarily decreased and $+\infty$ if the feasible set is empty. This value is also commonly known as the (possibly unattainable) *optimal value* of (2.10).

Definition 1. *The optimal value of a problem is defined as the infimum value of the objective function if minimizing, and the supremum value if maximizing. Its value can be*

1. $-\infty$ *(designates a minimization problem as unbounded).*
2. $+\infty$ *(designates a minimization problem as infeasible).*
3. *finite and attained by a feasible point.*
4. *finite and yet unattained by any feasible point; like the infimum of* $f(x) = 1/x$ *for* $x \geq 0$.

The optimal value is subject to study using duality theory [62, 19, 97]. To simplify the matter of duality for the considered conic optimization problem, it is assumed to be defined over a single cone $\mathcal{K}$ (taking advantage of the Cartesian product) which is non-empty, closed and convex (thereby satisfying the bipolar relation $(\mathcal{K}^*)^* = \mathcal{K}$ allowing for primal-dual pairs).

Satisfying these assumptions for the conic problem (2.10), it can be seen to be part of a primal-dual pair as elaborated in Section 2.4. In the remainder of this section, however, I will avoid deviating too much from existing work and consider the following de factor standard primal-dual pair. That is, for coefficients $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and a non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^m$, this section considers the primal-dual pair

$$(P): \quad \theta_P = \inf_x \{c^T x : Ax = b,\, x \in \mathcal{K}\}, \qquad (D): \quad \theta_D = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{K}^*\}, \tag{2.11}$$

where $\theta_P \in \mathbb{R} \cup \{-\infty, +\infty\}$ is the optimal value of the primal problem and $\theta_D \in \mathbb{R} \cup \{-\infty, +\infty\}$ is the optimal value of the dual. Primal-dual pairs have the property that the Lagrange-dual problem of either is the other [19] and furthermore satisfy *weak duality.*

Proposition 7 (Weak duality)**.** *The primal-dual pair* (2.11) *satisfies* $\theta_P \geq \theta_D$.

Proof. For all primal-dual feasible triples (x, s, y) of (2.11) we have $c^T x \geq b^T y$ as

$$0 \leq x^T s = x^T (c - A^T y) = c^T x - b^T y;$$

first inequality by the dual cone variable domains, $x \in \mathcal{K}$ and $s \in \mathcal{K}^*$, and remaining steps by the equations of (2.11). Taking the supremum (i.e., the least-upper-bound) over all feasible pairs (s, y) of (D) and the infimum (i.e., the greatest-lower-bound) over all feasible points x of (P) yields the claim. □

Weak duality can sometimes be strengthened to the notion of *strong duality.*

Definition 2. *Strong duality is said to hold for the primal-dual pair* (2.11) *if* $\theta_P = \theta_D$.

Strong duality does not hold in general and is not even satisfied for linear optimization problems. This is seen, e.g., in [Friberg 81, Example 1], when the feasible sets of the primal and dual problems are both empty. This is nevertheless the only exception to strong duality that can be found in linear optimization [105, Theorem 3] (also implied by the forthcoming Proposition 8), and a far greater variety of strong duality failures can be observed in conic optimization over non-polyhedral cones. One example is as follows.

Example 3 (Lack of strong duality)**.** *The following primal-dual pair of problems, taken from [7] and studied in [Friberg 81], are both feasible but have different optimal values:*

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 + x_2 + x_4 + x_5 = 0 \\ & -x_3 + x_4 = 1 \\ & x \in \mathcal{Q}^3 \times \mathcal{Q}^2, \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_1 \\ -y_2 \\ y_1 + y_2 \\ y_1 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3 \times \mathcal{Q}^2. \end{array} \tag{2.12}$$

Primal feasible points can be seen to satisfy $x_1+x_2 \geq 0$ *and* $x_4+x_5 \geq 0$ *given their membership in* $\mathcal{Q}^3 \times \mathcal{Q}^2$*. From the equation* $x_1 + x_2 + x_4 + x_5 = 0$ *we hence conclude* $x_1 + x_2 = 0$*, which given* $(x_1, x_2, x_3) \in \mathcal{Q}^3$ *implies* $x_3 = 0$ *for all primal feasible points such as* $\hat{x} = (0, 0, 0, 1, -1)$*. Dual feasible points can be seen to satisfy* $s_1 = s_2$*, which given* $(s_1, s_2, s_3) \in \mathcal{Q}^3$ *implies* $s_3 = 0$*. This in turn shows* $y_2 = -1$ *for all dual feasible points such as* $\hat{s} = (0, 0, 0, 1, 0)$ *and* $\hat{y} = (0, -1)$*. Conclusively, the primal optimal value is zero and the dual optimal value is* -1*.*

The following is a sufficient condition for strong duality. In contrast to Slater's original condition [90] in conic form [91], it allows for the use of non-solid cones and proves strong duality for primal or dual feasible problems in linear optimization. The definition of the generalized Slater condition was given already in [17], but the implications of it stated here is from [18].

Proposition 8 (The generalized Slater condition)**.** *Strong duality holds if, in either* (P) *or* (D)*, there exists a feasible point in the relative interior of all non-polyhedral cones. Moreover, in addition to strong duality, if this point is found in* (P) *then* (D) *is either infeasible or has an attained optimal value, and vice versa swapping* (P) *and* (D)*.*

In order to facilitate further analysis of the primal-dual pair, and characterize unboundedness, infeasibility and optimality, the following vocabulary is formalized.

Definition 3. *The following definitions denote the feasible sets of* (P) *and* (D)*, respectively, and in the latter case also the projection of the feasible set onto each set of variables.*

- $\Omega_P^x = \{x \in \mathbb{R}^n : Ax = b,\, x \in \mathcal{K}\}$.
- $\Omega_D^{(s,y)} = \{(s, y) \in \mathbb{R}^n \times \mathbb{R}^m : c - A^T y = s,\, s \in \mathcal{K}^*\}$.
- $\Omega_D^s = \{s \in \mathbb{R}^n : c - A^T y = s,\, s \in \mathcal{K}^*\}$.
- $\Omega_D^y = \{y \in \mathbb{R}^m : c - A^T y = s,\, s \in \mathcal{K}^*\}$.

Definition 4. *The following sets denote the (value-normalized) improving rays for* (P) *and* (D)*, respectively.*

- *The set of improving rays for* (P) *is* $\{c^T x = -1,\, Ax = 0,\, x \in \mathcal{K}\}$.
- *The set of improving rays for* (D) *is* $\{b^T y = +1,\, A^T y + s = 0,\, s \in \mathcal{K}^*\}$.

2.3.1 Unboundedness and infeasibility conditions

Unboundedness denotes the situation when the objective value can be improved indefinitely, while infeasibility (resp. feasibility) denotes that the feasible set is empty (resp. non-empty). These concepts are closely related via weak duality, and hence we begin by formalizing the former in terms of the optimal value.

Definition 5. *Consider the optimal values of* (P) *and* (D)*, respectively.*

- (P) *is said to be unbounded if* $\theta_P = -\infty$.
- (D) *is said to be unbounded if* $\theta_D = \infty$.

A problem obviously has to be feasible in order to be unbounded. Combining this with a non-empty set of improving rays yields a sufficient condition for unboundedness.

Proposition 9. *Problem* (P)*, respectively* (D)*, is unbounded if it has a feasible point and an improving ray.*

Proof. The objective value can be improved indefinitely by starting at the feasible point and moving in the direction of the improving ray. $\square$

A relation between unboundedness and infeasibility is given as follows.

Proposition 10. *Problem (P) is infeasible if (D) is unbounded, and vice versa swapping (P) and (D).*

Proof. By weak duality. □

The proposition above can be used to prove a stronger characterization of infeasibility.

Proposition 11. *Problem (P) is infeasible if (D) has an improving ray, and vice versa swapping (P) and (D).*

Proof. The objective of the considered problem does not affect the feasible set, nor does it affect the set of improving rays in the dual problem. Specifically, these set are unchanged if we fix the objective to zero. Doing so gives the dual problem a feasible point, the origin, making it unbounded if it has an improving ray. The claim hence follows by Proposition 10. □

Proposition 11 allows one to construct an easily verifiable certificate of infeasibility, namely a dual improving ray. Moreover, the proposition represents a both necessary and sufficient condition of infeasibility in linear optimization as shown by Farkas lemma over polyhedral cones [10, Theorem 3.5].

Proposition 12 (Farkas lemma over polyhedral cones)**.** *Let $\mathcal{K}$ and $\mathcal{K}^*$ be polyhedral cones of (P) and (D), respectively. Problem (P) is then infeasible if and only if (D) has an improving ray, and vice versa swapping (P) and (D).*

2.3.2 Optimality conditions

Combining the definition of feasibility with an objective leads to the common denotation of optimal solutions formalized below.

Definition 6. *A feasible point $(\hat{s}, \hat{y})$ of $\Omega_D^{(s,y)}$ is an optimal solution of (D) if and only if*

$$b^T \hat{y} \geq b^T y \quad \text{for all } y \in \Omega_D^y.$$

We can use this definition of optimality to derive the both necessary and sufficient Rockafellar-Pshenichnyi condition [97, Proof of Theorem 4.10].

Proposition 13. *A feasible point $(\hat{s}, \hat{y})$ of $\Omega_D^{(s,y)}$ is an optimal solution of (D) if and only if*

$$b \in -(\Omega_D^y - \hat{y})^*.$$

Proof. Optimality is defined as $b^T(\hat{y} - y) \geq 0$ for all $y \in \Omega_D^y$, or alternatively, $b^T z \geq 0$ for all $z \in \hat{y} - \Omega_D^y$. That is, $b \in (\hat{y} - \Omega_D^y)^*$ by definition of dual cones showing the claim. □

A necessary and sufficient variant of the Rockafellar-Pshenichnyi condition (closely related to [97, Proposition 5.1]) can also be derived in the s-variable.

Proposition 14. *A feasible point $(\hat{s}, \hat{y})$ of $\Omega_D^{(s,y)}$ is an optimal solution of (D) if and only if there exists a solution $x \in \mathbb{R}^n$ to the system*

$$Ax = b, \quad x \in (\Omega_D^s - \hat{s})^*.$$

Proof. Reformulate the objective function of (D) as $b^T y = x^T A^T y = x^T c - x^T s$, for any fixed choice of $x \in \mathbb{R}^n$ satisfying $Ax = b$. If there is no such x, then $Ax = b$ is inconsistent, and by Farkas lemma (Proposition 12) problem (D) has an improving ray which implies that the feasible point $(\hat{s}, \hat{y})$ is not optimal. Otherwise, $x^T c$ is constant after the reformulation. Optimality is thus defined as $x^T(s - \hat{s}) \geq 0$ for all $s \in \Omega_D^s$. That is, $x \in (\Omega_D^s - \hat{s})^*$. □

Consider next the KKT optimality conditions in conic form (e.g., [89, 19]) yielding a symmetric, but only sufficient, characterization of optimality in terms of the primal-dual pair (2.11).

Proposition 15. *The feasible points* $(\hat{s}, \hat{y})$ *of* $\Omega_D^{(s,y)}$ *and* $\hat{x}$ *of* Ω_P^x *are optimal solutions of* (D) *and* (P)*, respectively, if the complementarity condition*

$$\hat{x}^T \hat{s} = 0$$

is satisfied.

Proof. Follows by weak duality since $x^T s = x^T(c - A^T y) = c^T x - b^T y$. The connection to the KKT conditions is shown, e.g., in [42]. In summary, stationarity is given by the equation system of Ω_P^x, Lagrange multiplier feasibility is given the variable domain of Ω_P^x, and the remaining of the two complementarity conditions, $y^T(Ax - b)$, is trivially zero given $Ax = b$. □

The lack of strong duality in Example 3 exemplifies why the KKT conditions gives an inadequate description of optimality. It will now be compared to the necessary and sufficient Rockafellar-Pshenichnyi condition of Proposition 14 using the following technical result.

Lemma 2. *Let* $\mathcal{K} \subseteq \mathbb{R}^n$ *be a non-empty, closed, convex cone, and* $s \in \mathbb{R}$ *an arbitrary point. The following statements hold:*

1. $\text{cone}(\mathcal{K} - s) = \text{cone}(\mathcal{K} - \text{cone}(s))$.
2. $(\mathcal{K} - s)^* = \mathcal{K}^* \cap s^\perp$.

Proof. Statement 1 is different from the similar statement shown in [97, Proposition 3.2] because s is arbitrary. It follows, however, by $\text{cone}(\mathcal{K} - s) = \text{cone}(\lambda(\mathcal{K} - s)) = \text{cone}(\mathcal{K} - \lambda s)$ for $\lambda > 0$. In turn, Statement 2 follows by $(\mathcal{K} - s)^* = (\text{cone}(\mathcal{K} - s))^* = (\mathcal{K} - \text{cone}(s))^*$ and application of [86, Corollary 16.4.2]. □

Using this lemma, the KKT conditions can be restated as a sufficient condition for optimality on the same form as the Rockafellar-Pshenichnyi condition.

Corollary 1. *A feasible point* $(\hat{s}, \hat{y})$ *of* $\Omega_D^{(s,y)}$ *is an optimal solution of* (D) *if there exists a solution* $x \in \mathbb{R}^n$ *to the system*

$$Ax = b, \quad x \in (\mathcal{K}^* - \hat{s})^*.$$

Proof. Given Lemma 2, $(\mathcal{K}^* - \hat{s})^* = \mathcal{K} \cap \hat{s}^\perp = \{x \in \mathbb{R}^n : x \in \mathcal{K}, x^T \hat{s} = 0\}$, whereby the statement is seen equivalent to Proposition 15. □

In order to facilitate a comparison between the restated KKT conditions and the Rockafellar-Pshenichnyi condition, however, another technical result is needed.

Lemma 3. *Let* $S_1, S_2 \subseteq \mathbb{R}^n$ *and* $\hat{s} \in \mathbb{R}^n$ *be arbitrary sets and a point. Then* $S_1 \subseteq S_2$ *implies*

$$(S_1 - \hat{s})^* \supseteq (S_2 - \hat{s})^*.$$

Proof. Note that $(S_i - \hat{s})^* = \{x \in \mathbb{R}^n : x^T(s - \hat{s}) \geq 0, \forall s \in S_i\}$. Hence, $S_1 \subseteq S_2$ means that $(S_2 - \hat{s})^*$ is described by all the inequalities of $(S_1 - \hat{s})^*$ and possibly more. □

Finally, comparing Corollary 1 and Proposition 14, we see that $\Omega_D^s \subseteq \mathcal{K}^*$ is satisfied by definition of Ω_D^s and implies

$$(\Omega_D^s - \hat{s})^* \supseteq (\mathcal{K}^* - \hat{s})^*,$$

by Lemma 3. A strict containment here is thus the only reason for a difference between Corollary 1 and Proposition 14, and inadequateness of the KKT conditions can be explained by the set $(\mathcal{K}^* - \hat{s})^*$ being too small. This motivates reformulations of (D) which causes the set $(\mathcal{K}^* - \hat{s})^*$ to grow; a process called (cone) regularization.

2.3.3 Regularization and facial reduction

The comparison of Corollary 1 and Proposition 14 in the previous Section 2.3.2, concluded that reformulations which grow the set $(\mathcal{K}^* - \hat{s})^*$ acts to regularize problem (D) in the sense of making the KKT conditions a both necessary and sufficient condition for optimality. This growth can be facilitated, as shown in Lemma 3, by reformulations shrinking $\mathcal{K}^*$. The room for these reformulations is specified in the following corollary.

Corollary 2. *For any optimization problem, let Ω^ρ be its feasible set projected onto a variable ρ for which a constraint $\rho \in \hat{\mathcal{K}} \subseteq \mathbb{R}^r$ is given. Any non-empty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$ in the sandwich $\Omega^\rho \subseteq \mathcal{C} \subseteq \hat{\mathcal{K}}$ is a valid replacement of $\hat{\mathcal{K}}$ in the optimization problem.*

In general, we cannot shrink the cone $\mathcal{K}^*$ of (D) all the way down to Ω^s_D as this set may not be a non-empty, closed, convex cone hence violating the problem definition. Fortunately, there is no need for that as the Rockafellar-Pshenichnyi condition of Proposition 14 can be restated in terms of certain relaxations of Ω^s_D. To prove this, consider the smallest non-empty, close, convex cone $\mathcal{C}^*$ containing Ω^s_D, that is, $\mathcal{C}^* = \operatorname{cl}\operatorname{cone}(\Omega^s_D)$. The following proposition shows that the Rockafellar-Pshenichnyi condition of Proposition 14 is a necessary and sufficient condition for optimality even after replacing Ω^s_D by $\operatorname{cl}\operatorname{cone}(\Omega^s_D)$.

Proposition 16. *A feasible point $(\hat{s}, \hat{y})$ to the constraint set of (D), is an optimal solution if and only if there exists a solution $x \in \mathbb{R}^n$ to the system*

$$Ax = b, \quad x \in (\operatorname{cl}\operatorname{cone}(\Omega^s_D) - \hat{s})^*.$$

Proof. Reformulate the objective function of (D) as $b^T y = x^T A^T y = x^T c - x^T s$, for any fixed choice of $x \in \mathbb{R}^n$ satisfying $Ax = b$ and $x^T \hat{s} = 0$. If there is no such x, then $\left(\begin{smallmatrix} A \\ \hat{s}^T \end{smallmatrix}\right) x = \left(\begin{smallmatrix} b \\ 0 \end{smallmatrix}\right)$ is inconsistent, and by Farkas lemma (Proposition 12) there exists a $A^T y + \lambda \hat{s} = 0$ where $b^T y = 1$. For $\lambda \geq 0$, this an improving ray for (D) since $-A^T y = \lambda \hat{s} \in \mathcal{K}^*$. For $\lambda < 0$, this is an improving direction realized by $(\hat{s}, \hat{y}) + \frac{1}{|\lambda|}(\lambda \hat{s}, y)$ which is feasible since $c - A^T(\hat{y} + \frac{y}{|\lambda|}) = \hat{s} + \frac{\lambda}{|\lambda|}\hat{s} = 0 \in \mathcal{K}^*$. Both cases show $(\hat{s}, \hat{y})$ is not optimal. Otherwise, if there is such x to facilitate the reformulation, $x^T c$ is constant afterwards. Optimality is thus defined as $x^T(s - \hat{s}) = x^T s \geq 0$ for all $s \in \Omega^s_D$. That is, $x \in (\Omega^s_D)^*$. This characterization is finally reformulated by enforcing $x^T \hat{s} = 0$ via the cone giving $Ax = b$ and $x \in (\Omega^s_D)^* \cap \hat{s}^\perp = ((\Omega^s_D)^{**} - \hat{s})^*$, by Lemma 2, showing the claim. $\square$

An algorithm reducing the cone $\mathcal{K}^*$ of (D) to $\operatorname{cl}\operatorname{cone}(\Omega^s_D)$ was realized in [Friberg 38], and makes the KKT conditions a necessary and sufficient condition for optimality as argued. The paper concludes that while every iteration of the reformulating procedure is computationally efficient and numerically accurate, representational issues impede its usefulness.

In light of Proposition 16 and its algorithmic realization a relevant question arises. To what degree can the Rockafellar-Pshenichnyi condition of Proposition 14 can be restated in terms of relaxed sets of Ω^s_D? Or in other words, what is the least amount of regularization of (D) needed? The best answer so far seems to be the greatest non-empty, closed, convex cone $\mathcal{C}^* \subseteq \mathbb{R}^n$ in the sandwich $\Omega^s_D \subseteq \mathcal{C}^* \subseteq \mathcal{K}^*$ for which the generalized Slater condition of Proposition 8 is satisfied. To formalize this condition, an introduction of the minimal face [21] is needed.

Definition 7. *Let S be a non-empty subset of a non-empty, closed, convex cone $\mathcal{K}$. Then the minimal face of $\mathcal{K}$, containing S, is defined as the intersection of all faces of $\mathcal{K}$ containing S. That is,*

$$\operatorname{face}(S, \mathcal{K}) := \bigcap \{\mathcal{F} : S \subseteq \mathcal{F} \trianglelefteq \mathcal{K}\},$$

which is itself a face of $\mathcal{K}$.

Corollary 3. *Let S be a non-empty subset of a non-empty, closed, convex cone $\mathcal{K}$. Then* $\operatorname{face}(S, \mathcal{K})$ *is a non-empty, closed, convex cone in the sandwich*

$$S \subseteq \operatorname{face}(S, \mathcal{K}) \subseteq \mathcal{K}.$$

Proof. By Definition 7 and Proposition 2. □

The reason for introducing the notion of a minimal face is that the generalized Slater condition requires existence of a feasible point in the relative interior of all non-polyhedral cones (Proposition 8). It is hence convenient for our quest that the minimal face, $\Omega_D^s \subseteq \operatorname{face}(\Omega_D^s, \mathcal{K}^*) \subseteq \mathcal{K}^*$, is not only a valid replacement of $\mathcal{K}^*$ in (D), by Corollary 2, but also defines a relative interior-maximizing operation of Ω_D^s on $\mathcal{K}^*$.

Lemma 4. *Let S be any non-empty subset of a non-empty, closed, convex cone $\mathcal{K}$. The following statements hold:*

1. $\operatorname{relint} S \subseteq \operatorname{relint} \operatorname{face}(S, \mathcal{K})$.
2. $\operatorname{face}(S, \mathcal{K}) = \operatorname{face}(s, \mathcal{K})$ *for all* $s \in \operatorname{relint} S$.
3. *Any subset* $\mathcal{C} \subseteq \mathcal{K}$ *for which* $\operatorname{relint} S \cap \operatorname{relint} \mathcal{C} \neq \emptyset$*, satisfy* $\mathcal{C} \subseteq \operatorname{face}(S, \mathcal{K})$.

Proof. Statements 1–2 are by [21, Proposition 2.2.5], and the latter represents a reformulation of [86, Corollary 18.1.2]. For statement 3, there exists a $z \in \operatorname{relint} S \cap \operatorname{relint} \mathcal{C} \subseteq \operatorname{relint} \operatorname{face}(S, \mathcal{K}) \cap \operatorname{relint} \operatorname{face}(\mathcal{C}, \mathcal{K})$; containment by statement 1. Hence, with a relative interior point z in common, statement 2 shows $\operatorname{face}(S, \mathcal{K}) = \operatorname{face}(\mathcal{C}, \mathcal{K}) \supseteq \mathcal{C}$; containment by Corollary 3. □

We are now in position to reduce the need of regularization to the greatest extend allowed by the generalized Slater condition. Note that only the proof of the theorem below is novel, as the statement is a well known consequence of facial reduction algorithms; see, e.g., [97, Corollary 4.4] although the detail on polyhedral cones is missing from this reference.

Theorem 1. *A feasible point $(\hat{s}, \hat{y})$ to the constraint set of (D) with $\mathcal{K}^* = \mathcal{K}_1^* \times \cdots \times \mathcal{K}_k^*$, is an optimal solution if and only if there exists a solution $x \in \mathbb{R}^n$ to the system*

$$Ax = b, \quad x \in (\mathcal{C}_1^* \times \cdots \times \mathcal{C}_k^* - \hat{s})^*, \quad \mathcal{C}_j^* = \begin{cases} \mathcal{K}_j^*, & \text{if } \mathcal{K}_j^* \text{ is a polyhedral cone,} \\ \operatorname{face}\left((\Omega_D^s)_j, \mathcal{K}_j^*\right), & \text{otherwise,} \end{cases}$$

in terms of the Cartesian factors $\Omega_D^s = (\Omega_D^s)_1 \times \cdots \times (\Omega_D^s)_k$.

Proof. Let $(D)'$ denote (D) after replacing $\mathcal{K}^*$ by $\mathcal{C}^*$, and let $(P)'$ be its Lagrange-dual problem. Given $\Omega_D^s \subseteq \mathcal{C}^* \subseteq \mathcal{K}^*$, the feasible sets of $(D)'$ and (D) are equal by Corollary 2 and non-empty by existence of $(\hat{s}, \hat{y})$. Hence, there exists a feasible point (s, y) where $s = (s_1, \ldots, s_k)$ satisfies

$$s_j \in \begin{cases} (\Omega_D^s)_j \subseteq \mathcal{K}_j^* = \mathcal{C}_j^*, & \text{if } \mathcal{K}_j^* \text{ is a polyhedral cone,} \\ \operatorname{relint}(\Omega_D^s)_j \subseteq \operatorname{relint} \operatorname{face}\left((\Omega_D^s)_j, \mathcal{K}_j^*\right) = \operatorname{relint} \mathcal{C}_j^*, & \text{otherwise,} \end{cases}$$

which certifies the generalized Slater condition for $(D)'$. By strong duality and Proposition 8, either $(P)'$ is infeasible and $(D)'$ is unbounded showing $(\hat{s}, \hat{y})$ is not optimal, or $(P)'$ is attained and has a non-empty optimal set described by $c^T x = b^T \hat{y}$, $A = b$ and $x \in \mathcal{C}$. This characterization can be reformulated by rewriting the former equation as $0 = c^T x - b^T \hat{y} = x^T (c - A^T \hat{y}) = x^T \hat{s}$, and enforcing it via the cone giving $x \in \mathcal{C} \cap \hat{s}^\perp = (\mathcal{C}^* - \hat{s})^*$, by Lemma 2, showing the claim. □

An important consequence of Theorem 1 is that we can restrict regularization algorithms to only consider *facial reductions*, that is, reductions of $\mathcal{K}^*$ revealing its proper faces. The advantage of such a restriction is that the representational issues that occurred for the realization of Proposition 16 in [Friberg 38] are completely resolved, at least for the class of symmetric cones. The disadvantage of such a restriction, however, is that each reduction takes a greater computational effort to find, suggesting that a combination of the two may be advantageous; a research direction left open. The room for facial reductions allowed by Corollary 2 is materialized by what is here denoted *valid* facial reductions.

Definition 8. *For any optimization problem, let Ω^ρ be its feasible set projected onto a variable ρ for which a constraint $\rho \in \hat{\mathcal{K}} \subseteq \mathbb{R}^r$ is given. Any proper face $\mathcal{F} \trianglelefteq \hat{\mathcal{K}}$ in the sandwich $\Omega^\rho \subseteq \mathcal{F} \subseteq \hat{\mathcal{K}}$ is called a* valid *facial reduction of $\hat{\mathcal{K}}$.*

Corollary 4. *A valid facial reduction $\mathcal{F}$ of a non-empty, closed, convex cone $\hat{\mathcal{K}}$ is itself a non-empty, closed, convex cone by Proposition 2, and may thus replace $\hat{\mathcal{K}}$ in the optimization problem by Corollary 2.*

As can be verified, repeated use of valid facial reductions on non-polyhedral cones of (D) will eventual reduce $\mathcal{K}^*$ to $\mathcal{C}^*$ in Theorem 1, such that the KKT conditions becomes a necessary and sufficient condition for optimality as argued. Regularization algorithms utilizing this property, independently of whether polyhedral and non-polyhedral cones are distinguished, are commonly known as *facial reduction algorithms* and were first established algorithmically in [17]. An important and common special case of valid facial reductions, often utilized by these algorithms, are those that are exposed by so-called *facial reduction certificates*.

Definition 9. *For a non-empty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$, and the problem data $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ of (2.11), define facial reduction certificates as follows:*

- *Call $s \in \mathcal{K}^*$ a* facial reduction certificate for (P) *if the hyperplane $s^\perp$ contains the affine set $\{x \in \mathbb{R}^n : Ax = b\}$ and $\mathcal{K} \cap s^\perp \subseteq \mathcal{K}$ holds strictly.*
- *Call $x \in \mathcal{K}$ a* facial reduction certificate for (D) *if the hyperplane $x^\perp$ contains the affine set $\{c - A^T y : y \in \mathbb{R}^m\}$ and $\mathcal{K}^* \cap x^\perp \subseteq \mathcal{K}^*$ holds strictly.*

Facial reduction certificates are closely related to the subset of *non-improving rays* in the respective dual problem that cannot be extended to lines (or one-sided level directions in the vocabulary of [62]). This subset is now formalized.

Definition 10. *The subset non-improving rays that cannot be extended to lines (or one-sided level directions) are given by:*

1. $\mathring{\Omega}_P = \{x \in \mathbb{R}^n : c^T x = 0,\ Ax = 0,\ x \in \mathcal{K} \setminus (\mathcal{K}^*)^\perp\}$ *for problem (P);*
2. $\mathring{\Omega}_D = \{(s, y) \in \mathbb{R}^n \times \mathbb{R}^m : b^T y = 0,\ A^T y + s = 0,\ s \in \mathcal{K}^* \setminus \mathcal{K}^\perp\}$ *for problem (D),*

where $\mathcal{C}^\perp = \mathcal{C}^* \cap (-\mathcal{C}^*)$ is the largest linear subspace contained in $\mathcal{K}^*$.

The connection between the facial reduction certificates of Definition 9 and the non-improving rays of Definition 10 can now be established. Note that this result is slightly more elaborative than the one presented in [Friberg 81, Proposition 2].

Proposition 17. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. The following statements hold.*

1. *$s \in \mathbb{R}^n$ is a facial reduction certificate for the primal problem (P) if and only if there exists $y \in \mathbb{R}^m$ for which either*
$$(s, y) \in \mathring{\Omega}_D,$$
or $s \in \mathcal{K}^* \setminus \mathcal{K}^\perp$ and $(0, y)$ is an improving ray of (D).
2. *$x \in \mathbb{R}^n$ is a facial reduction certificate for the dual problem (D) if and only if*
$$x \in \mathring{\Omega}_P.$$

Proof. The two conditions of Definition 9 are restated to show the two claims as follows:

1. $\mathcal{K} \cap s^\perp \subseteq \mathcal{K}$ holds strictly if and only if $s \notin \mathcal{K}^\perp$ by the characterization of proper faces in Section 2.1. Hence, s is a facial reduction certificate if and only if $s^\perp$ contains the affine set $\{x \in \mathbb{R}^n : Ax = b\}$. If the affine set is empty this is trivially satisfied, and there exists an improving ray $(0, y)$ of (D) by Farkas lemma (Proposition 12). Otherwise, $s^T x = 0$ needs to be a valid equation for the system $Ax = b$ as shown by a row weighting $y \in \mathbb{R}^m$ such that $y^T A = s^T$ and $y^T b = 0$. This corresponds to $(s, -y) \in \mathring{\Omega}_D$ showing the claim.

2. $\mathcal{K}^* \cap x^\perp \subseteq \mathcal{K}^*$ holds strictly if and only if $x \notin (\mathcal{K}^*)^\perp$ as above. Hence, x is a facial reduction certificate if and only if $x^\perp$ contains the non-empty affine set $\{c - A^T y : y \in \mathbb{R}^m\}$. That is, $x^T(c - A^T y) = 0$ for all $y \in \mathbb{R}^m$ as solved by $x^T c = 0$ and $x^T A^T = 0$ showing the claim.

$\square$

The facial reduction certificates of $\mathring{\Omega}_P$ and $\mathring{\Omega}_D$ can easily be seen equivalent to the reducing certificates of [80, Lemma 2], the exposing vectors of [79, Lemma 1, statement 2] (as seen by Lemma 1) and, of course, the one-sided level directions of [62, Theorem 3]. Moreover, under appropriate assumptions of non-empty interiors, it is equivalent to the exposing vectors of [100, Lemma 2.3(ii)] and of [22, Lemma 12.6] (which specializes [17, Theorem 7.1]).

Complete regularization methods So far regularization has only been discussed with respect to optimality conditions, concerning only the special case of attained optimal values, but complete regularization methods should also address all other cases.

Recent work in this direction includes the simplified facial reduction algorithms of [79, 100] which, in contrast to [17], do not require a feasible point of the considered problem. These are therefore able to deal with infeasible problems in general. In particular, in case of infeasibility, these algorithms may keep on facially reducing the cone of the considered problem until it becomes polyhedral (e.g., equal to $\{0\}^n$), at which point Farkas lemma (Proposition 12) gives a necessary and sufficient condition for this infeasibility. This ability was made explicit in the facial reduction algorithm of [100].

Other cases includes unattained optimal values (for which all optimality conditions presented so far are inadequate) and unboundedness without improving rays (for which the sufficient condition of Proposition 9 is inadequate). These remaining cases were dealt with simultaneously in [Friberg 81] and [59]. The approach of [Friberg 81] is special, however, in the sense that it interleaves regularization and optimization by exploiting properties of the so-called self-dual embedding – more details given in Chapter 3. In practice, this allows one to regularize using common solutions methods of high efficiency, and avoid the computational burden and possible loss of dual information (briefly mentioning in Section 2.3.4) associated with unnecessary facial reduction. From a theoretical point of view it is also interesting that this method allows one to distinguish necessary and unnecessary facial reductions. To give an example, note first that unnecessary facial reductions are shown to appear in [80, Section 5.1.1] for semidefinite problems, but are also present in second-order cone optimization.

Example 4. *Consider the primal-dual pair,*

$$\begin{array}{ll} \text{minimize} & 2x_1 + x_2 + x_3 \\ \text{subject to} & x_1 - x_2 = 0 \\ & x \in \mathcal{Q}^3, \end{array} \qquad \begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} y \\ -y \\ 0 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3. \end{array}$$

The minimization problem fails the generalized Slater condition (Proposition 8) as verified by any valid facial reduction of a nonlinear cone, noting that the relative interior of different faces are non-overlapping (Lemma 4). In particular, enforcing $x_1 - x_2 = 0$ via the cone exposes the proper face $x \in \mathcal{Q}^3 \cap \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}^\perp = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \mathbb{R}_+$ (see [Friberg 38]). Nevertheless, this valid facial *reduction is unnecessary as seen by the KKT conditions of Proposition 15 which are satisfied by*

$$\hat{x} = (0, 0, 0), \quad \hat{s} = (2, 1, 1), \quad \hat{y} = 0.$$

Finally, take note of the fact that Example 4 and [80, Section 5.1.1] prove existence of an even stronger characterization of optimality than the one presented in Theorem 1, that is, one for which even less facial reduction is needed. Quantifying this in a computationally effective manner is open to further research.

2.3.4 On the unnecessary loss of dual information

Proactive use of regularization has a downside. In particular, regularization of the considered problem changes the feasible set of its dual problem thereby resulting in a possible loss of dual information at the optimal solution. This part should come as no surprise, though, as the whole purpose of regularization is to reformulate the considered problem such that the dual problem changes to satisfy the desired properties, e.g., strong duality. The problem, however, lies in the fact that active use of regularization may also cause unnecessary reformulations to be applied. These reformulations will also change the dual problem and unnecessarily throw away the dual information.

The challenge of recovering dual information after unnecessary facial reduction is considered in [80], where a simple procedure based on line search is established. In [80, Section 5.1.1], however, it is also shown that this procedure is only heuristic and may fail even when dual recovery is possible. While this is not a concern for the interleaved regularization and optimization method of Chapter 3 as argued in the previous section, it does affect the facial reduction heuristics explored in [Friberg 39]. This thesis has not made progress on the topic, however, and dual recovery hence remain open to further research.

2.4 MIN-VAR: A mnemonic for dualization by hand

The considered primal-dual pair (2.11) is useful from a theoretical point of view, but cumbersome when working with duality in practice. In particular, real problems rarely fit into these rigid definitions. A more general primal-dual pair was hence presented in [Friberg 36] and helped shape the Conic Benchmark Format [Friberg 35]. That is,

$$\begin{array}{llcccl} \underset{x}{\text{minimize}} & c^T x & & & & \\ \text{subject to} & Ax & - & b & \in & \mathcal{K}_g, \\ & & & x & \in & \mathcal{K}_x, \end{array}$$

and

$$\begin{array}{llcccl} \underset{y}{\text{maximize}} & b^T y & & & & \\ \text{subject to} & A^T y & - & c & \in & -\mathcal{K}_x^*, \\ & & & y & \in & \mathcal{K}_g^*, \end{array}$$

for coefficients $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and a non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^m$. This dualization result is easily confirmed via reformulation to and from the primal-dual pair (2.11). Moreover, assuming knowledge of dual cones, it may even be simpler to memorize than the SOB (Sensible-Odd-Bizarre) mnemonic introduced in [13] and limited to linear programs. Decide for yourself: Table 2.1 is reconstructed with slight modifications from the text book [49] to illustrate the rules of the SOB mnemonic. Now compare this to Table 2.2 which encodes the exact same information.

Note first that while Table 2.2 is based on the general primal-dual pair above, the table does not assume that conic constraints are combined using the Cartesian product and hence can be used directly on conic forms such as (2.10).

Note second that the only difficulty in memorizing Table 2.2 regards the placement of the negation and this is where the MIN-VAR mnemonic finds its use. The mnemonic can be stated as follows: You dualize the cone, and whenever you transform to or from the *variables* of the *minimization* problem (the MIN-VAR mnemonic) you also negate the cone. That's it!

	Minimize		Maximize
	Constraint i:		Variable i:
Sensible	$\geq$ form	$\longleftrightarrow$	$\geq$ form
Odd	$=$ form	$\longleftrightarrow$	Free
Bizarre	$\leq$ form	$\longleftrightarrow$	$\leq$ form
	Variable i:		Constraint i:
Sensible	$\geq$ form	$\longleftrightarrow$	$\leq$ form
Odd	Free	$\longleftrightarrow$	$=$ form
Bizarre	$\leq$ form	$\longleftrightarrow$	$\geq$ form

Table 2.1: Dualization rules based on the SOB mnemonic.

Minimize		Maximize
Constraint i:		Variable i:
$\mathcal{K}$ form	$\longleftrightarrow$	$\mathcal{K}^*$ form
Variable i:		Constraint i:
$\mathcal{K}$ form	$\longleftrightarrow$	$-\mathcal{K}^*$ form

Table 2.2: Dualization rules based on the MIN-VAR mnemonic.

2.5 Head and base: A naming convention for conic entries

The word *cone head* appears sporadically on online forums and conference discussions to denote the first entry of the quadratic cone. To my knowledge, however, it has never been given a formal definition and has never appeared in a paper. I propose to define the head and corresponding base of a cone as follows.

Definition 11. *Let $\mathcal{K} \subseteq \mathcal{X}$ be any non-empty, closed, convex cone in a vector space $\mathcal{X}$ with standard basis $\{e_j\}_{j\in\{1,\ldots,n\}}$. Then a head of $\mathcal{K}$ is any minimal subset $\{e_j\}_{j\in\mathcal{J}}$ such that*

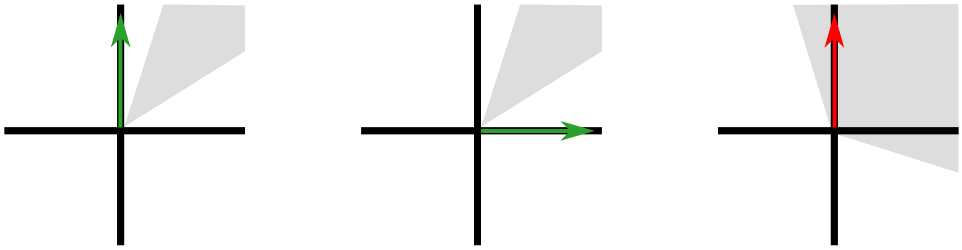
$$s = \sum_{j\in\mathcal{J}} \lambda_j e_j \in \operatorname{relint} \mathcal{K}^*,$$

for some $\lambda \in \mathbb{R}^n$. The complement of a head, $\{e_j\}_{j\in\{1,\ldots,n\}\setminus\mathcal{J}}$, is the corresponding base.

This definition leads directly to the notion of head and base entries of a cone $x \in \mathcal{K}$, since there is a unique correspondence between steps $\lambda_{\mathcal{J}}$ in the basis directions $\{e_j\}_{j\in\mathcal{J}}$ and changes $\lambda_{\mathcal{J}}$ in the entries $x_{\mathcal{J}}$. The following are realizations of this definition.

Corollary 5. *Let $x \in \mathcal{Q}^n \subseteq \mathbb{R}^n$. Then x_1 is the head and $x_{2:n}$ the base of the quadratic cone.*

Corollary 6. *Let $X \in \mathcal{S}^n_+$ in the vector space of symmetric matrices. Then the diagonal entries of X is the head and the off-diagonal entries is the base of the semidefinite cone.*



(a) Head direction which is not a relative interior direction.

(b) Head direction which is not a relative interior direction.

(c) Relative interior direction which is not a head direction.

Figure 2.1: Examples of the head definition.

In clarification of Definition 11 and its relation to the proposed names, head and base, note that the cone can be thought of as heading in direction s. In particular, $s^T x_{ray} > 0$ for all one-sided directions $x_{ray} \in \mathcal{K} \setminus (\mathcal{K}^*)^\perp$ by Lemma 1. Moreover, bounding the direction s in $\mathcal{K}$ leaves a generating base (a bounded subset plus the largest linear subspace $(\mathcal{K}^*)^\perp = \mathcal{K} \cap (-\mathcal{K})$ of $\mathcal{K}$ [62]) generating the entire cone $\mathcal{K}$ via scaling.

Theorem 2. *Consider a non-empty, closed, convex cone with a head defining some direction $s = \sum_{j \in \mathcal{J}} \lambda_j e_j \in \operatorname{relint} \mathcal{K}^*$ by Definition 11. Bounding $\mathcal{K}$ in direction s leaves a generating base of the form $B = S + (\mathcal{K} \cap (-\mathcal{K}))$, for some bounded set $S \subseteq \mathcal{K}$, which satisfy $\mathcal{K} = \cup_{\lambda \geq 0}(\lambda B)$.*

Proof. Bounding $\mathcal{K}$ in direction s gives $\mathcal{K} \cap \{x : s^T x \leq a\} = \{x \in \mathcal{K} : s^T x \leq a\}$ for some constant $a \in \mathbb{R}_+$. By dual cones, this is equal to $\{x \in \mathcal{K} : 0 \leq s^T x \leq a\}$ and all directions are hence given by $\mathcal{K} \cap \{x : s^T x = 0\} = \mathcal{K} \cap (\mathcal{K}^*)^\perp = \mathcal{K} \cap (-\mathcal{K})$; the first equality by statement 1 of Lemma 1 with $\mathcal{C} = \mathcal{K}^*$, the second by definitions. This shows $\mathcal{K} \cap \{x : s^T x \leq a\} = S + (\mathcal{K} \cap (-\mathcal{K}))$, for some bounded set $S \subseteq \mathcal{K}$. To complete the claim, note that this set is a generating base for the entire cone since $\cup_{\lambda > 0}\big(\lambda(\mathcal{K} \cap \{x : s^T x \leq a\})\big) = \cup_{\lambda > 0}\big(\mathcal{K} \cap \{x : s^T x \leq \lambda a\}\big) = \mathcal{K}$. □

For pointed cones where $\mathcal{K} \cap (-\mathcal{K}) = \{0\}^n$, a simple consequence of Theorem 2 is a follows.

Corollary 7. *Let $\mathcal{K}$ be a pointed, non-empty, closed, convex cone. Fixing the value of all entries of some head of $\mathcal{K}$ the resulting set, $\mathcal{K} \cap \{x : x_{\mathcal{J}} = \hat{x}_{\mathcal{J}}\}$, is bounded.*

Finally I would like to stress, as illustrated in Figure 2.1, that the head definition may not be unique nor correspond to a relative interior direction for non-symmetric cones. The latter part is intentionally, however, as the properties of Theorem 2 and Corollary 7 does not hold for relative interior directions in general as shown by Figure 2.1c.

3
Interleaved regularization and optimization

For coefficients $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and a non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^m$, consider again the primal-dual pair from (2.11),

$$(P): \quad \theta_P = \inf_x \{c^T x : Ax = b,\, x \in \mathcal{K}\}, \qquad (D): \quad \theta_D = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{K}^*\},$$

This problem pair was treated in [Friberg 81] by means of the *self-dual embedding* technique, originally due to Goldman and Tucker [44] and generalized in [61, 25, 75, 84], which solves (2.11) by finding solutions to the following *self-dual homogeneous model*:

$$\begin{array}{rcl} Ax - b\tau & = & 0, \\ -A^T y - s + c\tau & = & 0, \\ b^T y - c^T x - \kappa & = & 0, \\ (x, s, y, \tau, \kappa) \in \mathcal{K} \times \mathcal{K}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+. & & \end{array} \tag{3.1}$$

The homogeneous model (3.1) contains the feasible set of (P) and (D) with constants homogenized by $\tau \geq 0$. Moreover, while weak duality, $c^T x \geq b^T y$ as in Proposition 7, implicitly holds for all feasible triples (x, s, y) of (2.11), the opposite relation, $c^T x \leq b^T y$, is enforced explicitly by the homogeneous model with slack $\kappa \geq 0$. This causes all solutions of the homogeneous model to satisfy the following two complementarity relations.

Proposition 18. *All feasible points (x, s, y, τ, κ) of (3.1) satisfy $x^T s = \tau\kappa = 0$.*

Proof. Follows by $0 \leq x^T s = x^T(c\tau - A^T y) = \tau(c^T x - b^T y) = -\tau\kappa \leq 0$, the former inequality by the definition of dual cones. □

The complementarity relation, $\tau\kappa = 0$, gives a tight coupling between the homogeneous model (3.1) and the primal-dual pair (2.11).

Proposition 19. *The following solutions to the homogeneous model (3.1) can be interpreted in terms of the primal-dual pair (2.11):*

1. *If $\tau > 0$: Primal-dual optimum of (2.11) given by $(x, y, s)/\tau$ since $b^T y - c^T x = \kappa = 0$.*
2. *If $\tau = 0$:*

   From the primal side:

   (a) *If $c^T x < 0$: Dual infeasibility certified by the improving ray x (Proposition 11).*

   (b) *If $c^T x = 0$ and $x \notin (\mathcal{K}^*)^\perp$: Reduction from $\mathcal{K}^*$ to its proper face $\mathcal{K}^* \cap x^\perp$ in (D), certified by the facial reduction certificate x (Proposition 17).*

   From the dual side:

   (c) *If $b^T y > 0$: Primal infeasibility certified by the improving ray (s, y) (Proposition 11).*

   (d) *If $b^T y = 0$ and $s \notin \mathcal{K}^\perp$: Reduction from $\mathcal{K}$ to its proper face $\mathcal{K} \cap s^\perp$ in (P), certified by the facial reduction certificate (s, y) (Proposition 17).*

Note that the list of interpretations in Proposition 19 is not exhaustive as, for instance, $b^T y = s = \tau = 0$ and $y \neq 0$ certifies linear dependency between the rows of A in (2.11). This does not matter, however, as the list given above is enough for the interleaved regularization and optimization procedure. In particular, the following condition guarantees that one of these interpretations exist.

Corollary 8. *There is an interpretation for the primal-dual pair* (2.11) *for all solutions of the homogeneous model* (3.1)*, additionally satisfying the logical condition*

$$[\tau \notin (\mathbb{R}_+^*)^\perp = \{0\}] \text{ or } [\kappa \notin (\mathbb{R}_+^*)^\perp = \{0\}] \text{ or } [x \notin (K^*)^\perp] \text{ or } [s \notin K^\perp].$$

Proof. If $\tau > 0$, statement 1 of Proposition 19 holds. Suppose $\tau = 0$. If $\kappa > 0$, then either statement 2a or 2c of Proposition 19 holds. Suppose also $\kappa = 0$. If $x \notin (K^*)^\perp$, then either statement 2a, 2b or 2c holds, and if $s \notin K^\perp$, then either statement 2c, 2d or 2a holds. $\square$

The logical condition of Corollary 8 can also be represented by a solution normalizing equation.

Corollary 9. *There is an interpretation for the primal-dual pair* (2.11) *for all solutions of the homogeneous model* (3.1)*, additionally satisfying any equation of the form*

$$\tau\hat{\kappa} + \hat{\tau}\kappa + x^T\hat{s} + \hat{x}^T s = \mu,$$

for $\hat{\kappa}, \hat{\tau} \in \operatorname{relint} \mathbb{R}_+^* = \mathbb{R}_{++}$*,* $\hat{s} \in \operatorname{relint} \mathcal{K}^*$*,* $\hat{x} \in \operatorname{relint} \mathcal{K}$*, and some* $\mu > 0$*.*

Proof. By dual cones the strict inequality, $\tau\hat{\kappa} + \hat{\tau}\kappa + x^T\hat{s} + \hat{x}^T s > 0$, holds if and only if

$$[\tau\hat{\kappa} > 0] \text{ or } [\hat{\tau}\kappa > 0] \text{ or } [x^T\hat{s} > 0] \text{ or } [\hat{x}^T s > 0],$$

showing the claim by Corollary 8 after application of Lemma 1. $\square$

Finally, the solution normalizing equation of Corollary 9 can be seen satisfied by all optimal solutions of the so-called *extended embedding* of Ye et al. [105].

Corollary 10. *There is an interpretation for the primal-dual pair* (2.11) *for all optimal solutions of the extended embedding of* (3.1) *given by*

$$\begin{array}{ll} \underset{x,s,y,\tau,\kappa,\theta}{\textit{minimize}} & \mu\theta \\ \textit{subject to} & \begin{array}{rcl} Ax - b\tau & = & r_p\theta, \\ -A^T y - s + c\tau & = & r_d\theta, \\ b^T y - c^T x - \kappa & = & r_g\theta, \\ r_p^T y + r_d^T x + r_g\tau & = & -\mu, \\ (x,s,y,\tau,\kappa,\theta) \in \mathcal{K} \times \mathcal{K}^* \times \mathbb{R}^m & \times & \mathbb{R}_+ \times \mathbb{R}_+ \times \mathbb{R}, \end{array} \end{array}$$

where $(\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa})$ *are chosen in the relative interior of* $\mathcal{K} \times \mathcal{K}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+$ *to define*

$$r_p = A\hat{x} - b\hat{\tau}, \qquad r_d = -A^T\hat{y} - \hat{s} + c\hat{\tau}, \qquad r_g = b^T\hat{y} - c^T\hat{x} - \hat{\kappa}, \qquad \mu = \hat{x}^T\hat{s} + \hat{\tau}\hat{\kappa}.$$

Proof. Weighting the rows of the equation system by $(\hat{y}^T, \hat{x}^T, \hat{\tau}, 1)$, one gets

$$-\tau\hat{\kappa} - \hat{\tau}\kappa - x^T\hat{s} - \hat{x}^T s = -\theta\mu - \mu.$$

The claim hence follows by Corollary 9, since optimal solutions of the extended embedding satisfy $\theta = 0$ (see [105] or [Friberg 81]) and $\mu > 0$ by definition. $\square$

3.1 Optimal values

Existence of some interpretation from some optimal solution to the extended embedding of (3.1), as established in Corollary 10, is not always enough to determining the optimal value of the considered problem, (P) or (D). The beauty of [Friberg 81] is that the interpretations from *relative interior* solutions of the self-dual homogeneous embedding (3.1) are shown sufficient for this purpose, and that these can be obtained via relative interior (also known as maximally complementary) optimal solutions to the extended embedding in Corollary 10. Hence, if the central path of the extended embedding converges to the relative interior of the optimal set, any central-path following algorithm will, in principle, yield the information needed to work out the optimal value although practical issues of numerical accuracy may interfere. This requirement is now formalized.

Requirement 1. *The considered primal-dual pair gives rise to an extended embedding (as in Corollary 10) with a central path whose limit point belongs to the relative interior of the extended embedding's optimal set.*

Given that some of the interpretations from Proposition 19 are facial reduction certificates, this requirement has to be satisfied recursively, as formalized next, for an optimal value computation to succeed.

Requirement 2. *The considered primal-dual pair satisfy Requirement 1 after any number of primal and dual facial reductions.*

In terms of these requirements, the computability of optimal values using the interleaved regularization and optimization algorithm proposed in [Friberg 81], can be boiled down to the following statement.

Theorem 3. *The interleaved regularization and optimization algorithm can compute the optimal value for any problem,* (P) *or* (D)*, given that the primal-dual pair* (2.11) *satisfy Requirement 2. The computation will also show whether the optimal value is attained.*

Proof. By [Friberg 81, Theorem 5] and [Friberg 81, Theorem 3] for (P), which can be extended to (D) as discussed in [Friberg 81, Section 4.2]. □

The requirements of this theorem can be satisfied as shown next, using existing results on the limiting behavior of central paths for semidefinite optimization problems.

Proposition 20. *The interleaved regularization and optimization algorithm can compute the optimal value for any semidefinite optimization problem. The computation will also show whether the optimal value is attained.*

Proof. The claim follows by Theorem 3 since Requirement 1 is satisfied in semidefinite optimization [47] and faces of a semidefinite cone are semidefinite-representable (see, e.g., [80]) hence implying Requirement 2. □

Finally, of special interest to this thesis, the following corollary is obtained as a consequence of Proposition 20 and the definition of the central path, given equivalence of Jordan products in the standard semidefinite representations of linear and second-order cones.

Corollary 11. *The interleaved regularization and optimization algorithm can compute the optimal value for any conic optimization problem over a cone composed as a Cartesian product of linear, second-order and semidefinite cones. The computation will also show whether the optimal value is attained.*

3.2 Solutions and other certificates

The interleaved regularization and optimization algorithm proposed in [Friberg 81] is able to produce *certificates* for all computed optimal values. Although rarely formalized, the common usage of the term certificate seems to comply with the rigorous definition used here.

Definition 12. *A* certificate *is defined with respect to some predicate* $P(x) : \Omega \to \{\texttt{false}, \texttt{true}\}$ *which, if true for any* $\hat{x} \in \Omega$*, implies the statement* Q*, that is*

$$[\exists \hat{x} \in \Omega : P(\hat{x})] \implies Q. \tag{3.2}$$

Specifically, a point $\hat{x} \in \Omega$ *is called a* certificate of Q*, with respect to* (3.2)*, if* $P(\hat{x})$ *is true.*

Most of the time, and in this thesis, it is moreover implicitly understood by a certificate that the predicate $P(x)$ of Definition 12 is trivial to evaluate on its domain. In the interleaved regularization and optimization algorithm of [Friberg 81], three basic certificates lay the basis for certifying optimal values; one based on the optimality condition of Proposition 15, one based on the unboundedness condition of Proposition 9, and one for based on the infeasibility condition of Proposition 11. As already seen, however, these propositions represent only sufficient conditions that can not always be satisfied. The algorithm works around this issue by not promising one of these certificates for the primal-dual pair (2.11), but only for a member of the cone-parametrized family of primal-dual pairs given by

$$\begin{aligned} P(\mathcal{C}) : \quad & \theta_P(\mathcal{C}) = \inf_{x} \{c^T x : Ax = b,\, x \in \mathcal{C}\}, \\ D(\mathcal{C}) : \quad & \theta_D(\mathcal{C}) = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{C}^*\}, \end{aligned} \tag{3.3}$$

for non-empty, closed, convex cones $\mathcal{C} \subseteq \mathbb{R}^m$. Specifically, the algorithm finds a cone $\mathcal{C}$ for which the optimal value of the considered problem, say $P(\mathcal{K})$, and the regularized problem, say $P(\mathcal{C})$, are equal, a so-called *regularization certificate* for this equality [Friberg 81, Definition 8], and one of the three before-mentioned basic certificates for $P(\mathcal{C})$. This behavior is now formalized, following a proper write-up of the basic certificates.

Definition 13. *Let* $\mathcal{C} \subseteq \mathbb{R}^n$ *be a non-empty, closed, convex cone. For the conic optimization problem* $P(\mathcal{C})$*, define the following basic certificates:*

- *A* complementary solution *is a triple* $(x, s, y) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ *satisfying*

$$Ax = b, \qquad s = c - A^T y, \qquad x^T s = 0.$$

- *An* unboundedness certificates *is a tuple* $(x, x_{ray}) \in \mathcal{C} \times \mathcal{C}$ *satisfying*

$$Ax = b, \qquad Ax_{ray} = 0, \qquad c^T x_{ray} < 0.$$

- *An* infeasibility certificates *is a tuple* $(s_{ray}, y_{ray}) \in \mathcal{C}^* \times \mathbb{R}^m$ *satisfying*

$$s_{ray} = -A^T y_{ray}, \qquad b^T y_{ray} > 0.$$

Theorem 4. *Suppose Requirement 2 is satisfied for the primal-dual pair* (2.11)*. Given* $P(\mathcal{K})$*, the interleaved regularization and optimization algorithm finds a non-empty, closed, convex cone* $\mathcal{C} \subseteq \mathbb{R}^n$*, a regularization certificate [Friberg 81, Definition 8] showing that the optimal values of* $P(\mathcal{K})$ *and* $P(\mathcal{C})$ *are equal, and one of the basic certificates from Definition 13 for* $P(\mathcal{C})$*. The equivalent statement in terms of* $D(\mathcal{K})$ *and* $D(\mathcal{C})$ *holds as well.*

Proof. By [Friberg 81, Theorem 5] and [Friberg 81, Theorem 3] for (P), which can be extended to (D) as discussed in [Friberg 81, Section 4.2]. The definition and construction of the regularization certificate is found in [Friberg 81, Section 4.1.3]. □

Algorithms that satisfy Theorem 4 can be said to *weakly solve* a conic optimization problem because, while they do find the optimal value, there is potentially very little information to be extracted from the certificate. Specifically, a weakly solving algorithm cannot be expected to yield optimal solutions for the original primal-dual pair (2.11) even if they exist. This scenario can happen when using facial reduction algorithms in the traditional way [17, 79, 100, 22, 59] or when using the facial reduction heuristics of [Friberg 39], given the existence of unnecessary facial reductions, illustrated by Example 4, changing the dual problem. Algorithms that satisfy Theorem 4 with guarantees of cone equivalence $\mathcal{K} = \mathcal{C}$ if possible, and feasible set equivalence for the considered and regularized problem if possible, can be said to *strongly solve* a conic optimization problem. This property holds for the interleaved regularization and optimization algorithm of [Friberg 81] as will be elaborated once a basic understanding of the algorithm has been established.

The interleaved regularization and optimization algorithm is sketched by Algorithm 2. The algorithm consist of simple logic based on the output from at most two calls to the real workhorse, namely Algorithm 1. In particular, Algorithm 1 satisfy all aspects of Theorem 4 by itself, except that it is only able to return the first half of an unboundedness certificate (the improving ray). Two calls are hence needed to finish the unboundedness certificate, but may also be needed if both the primal and dual problems are infeasible and only the improving ray certifying infeasibility of the Lagrange-dual problem was identified. The first call is with respect to the optimization problem itself, and the potential second call is with respect to its *feasibility problem* where the objective function is fixed to zero. These algorithmic descriptions are based on [Friberg 81, Algorithm 1 and Algorithm 2] where more details can be found. The properties that makes the interleaved regularization and optimization algorithm special are summarized in the following proposition.

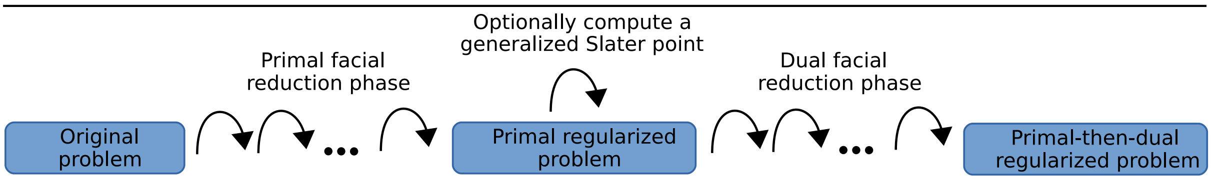
Proposition 21. *The following statements hold for the interleaved regularization and optimization algorithm:*

1. *Facial reduction is used only when needed. In particular, if a complementary solution, infeasibility certificate or improving ray exists in any iteration of Algorithm 1, it will be returned immediately without further reductions.*

2. *When facial reduction is needed, only optimal facial reductions [Friberg 81, Definition 3] are found and used in Algorithm 1, causing the greatest possible reduction of all facial reduction certificates in each iteration.*

3. *When dual facial reduction is needed, the algorithm will know exactly when to switch to the dual facial reduction phase with no need to compute a generalized Slater point. This point exists and is required, however, for construction of the regularization certificate.*

Algorithm 1: Regularize as shown until one of the following items come into existence: a complementary solution, infeasibility certificate or improving ray.



Algorithm 2: Construct one of the items from Definition 13.

Execute Algorithm 1 on the *optimization problem* to find:

1. **Complementary solution**.
2. **Infeasibility certificate**.
3. Improving ray. In this case, execute Algorithm 1 on the *feasibility problem* to find:
   (a) Complementary solution, that is, an **unboundedness certificate** when combined with the improving ray.
   (b) **Infeasibility certificate**.
   (c) ~~Improving ray~~. This case cannot occur for the feasibility problem as shown by [Friberg 81, Theorem 4, statement 3].

Proof. Statement 1 and 2 follows by [Friberg 81, Theorem 1]. Statement 3 is a consequence of [Friberg 81, Corollary 2] proven explicitly in [Friberg 81, Theorem 4, statement 4a]. □

Statement 1 of Proposition 21 implies, together with previous results on linear optimization, that the algorithm never uses facial reduction when all cones of the primal-dual pair (3.3) are polyhedral because this will always be unnecessary.

Proposition 22. *The interleaved regularization and optimization algorithm does not use facial reduction when given a problem where all cones are polyhedral.*

Proof. By the generalized Slater's condition (Proposition 8), Farkas lemma (Proposition 12), and statement 1 of Proposition 21. □

Statement 1 of Proposition 21 should not be overinterpreted, however, as it notably does not imply that unnecessary facial reductions are completely avoided. In particular, since optimal facial reductions are used in each iteration of Algorithm 1, as noted in statement 2 of Proposition 21, a cone composed of multiple Cartesian factors may have some of its factors reduced unnecessarily. The algorithm is nevertheless capable of distinguishing three fundamental cases apart: when no regularization is needed, when a primal regularization phase is needed, and when a dual regularization phase is needed (compare to Algorithm 1). Hence, by this distinction, the algorithm can be seen to *strongly solve* a conic optimization problem by guaranteeing cone equivalence $\mathcal{K} = \mathcal{C}$ if possible, and feasible set equivalence for the considered and regularized problem if possible. This is elaborated and formalized in the following subsections.

3.2.1 When regularization is not needed

When regularization is not needed, the following corollary holds.

Corollary 12. *Suppose regularization is not needed to construct a certificate from Definition 13 for the considered problem, (P) or (D). Then the interleaved regularization and optimization algorithm will not use facial reduction (Proposition 21, statement 1) and hence satisfy Theorem 4 with $\mathcal{C} = \mathcal{K}$. Hence, primal and dual feasible points, optimal solutions and improving rays from the identified certificate of Definition 13 are also valid for the original primal-dual pair* (2.11).

In this case where regularization is not needed, it can also be note that primal-dual interior-point methods with no built-in regularization mechanism, such as [8], will also succeed as a substitute for Algorithm 1. For some reason, however, the additional logic of Algorithm 2 needed to guarantee that the considered problem is solved, seems to be left out of open source and commercial implementations. Admittedly, this common incompleteness may be explained by a fail-first principle given that improving rays often indicates a mistake in the practitioner's problem formulation.

3.2.2 When only primal regularization is needed

When primal regularization is needed, but dual regularization is not, the following corollary holds.

Corollary 13. *Suppose $P(\mathcal{K})$ is the problem of interest, and that no dual regularization is needed to construct a certificate from Definition 13. Then the interleaved regularization and optimization algorithm will not use dual facial reduction (Proposition 21, statement 1) and hence satisfy Theorem 4 for a cone $\mathcal{C} \subseteq \mathcal{K}$ maintaining equivalence between the feasible set of $P(\mathcal{C})$ and $P(\mathcal{K})$. Hence, primal feasible points, optimal solutions and improving rays from identified certificate of Definition 13 are also valid for the original primal problem $P(\mathcal{K})$.*

3.2.3 When also dual regularization is needed

The interleaved regularization and optimization algorithm shows that there only exists two subtle cases for which dual regularization can be needed.

Proposition 23. *Given $P(\mathcal{K})$, the interleaved regularization and optimization algorithm uses dual regularization if and only if $P(\mathcal{K})$ has a finite but unattained optimal value or is unbounded and has an empty set of improving rays.*

Proof. By [81, Theorem 4, statement 3]. □

When dual regularization is needed the following corollary holds.

Corollary 14. *Suppose $P(\mathcal{K})$ is the problem of interest, and that dual regularization is needed to construct a certificate from Definition 13. Then the interleaved regularization and optimization algorithm will satisfy Theorem 4 for a cone $\mathcal{C}$ maintaining that the feasible set of $P(\mathcal{C})$ is greater than or equal to that of $P(\mathcal{K})$. In other words, $P(\mathcal{C})$ is a relaxation of $P(\mathcal{K})$.*

Corollary 14 surprisingly states that when attempting to solve a problem that requires dual regularization, we actually end up solving a relaxation of that problem instead. With respect to mixed-integer optimization, as elaborated in Chapter 4, this is just fine as all relaxations (even the unintentional ones) are useful. With respect to continuous optimization, however, one may desire more intuition behind the identified certificate for $P(\mathcal{C})$. This issue was addressed for unattained optimal values in the slightly different context of [59, Section 6], in terms of the computable generalized Slater point (see Algorithm 1) and points from the certificate for $P(\mathcal{C})$. In particular, the arguments of [59] show that convex combinations of these points are capable of generating a sequence of feasible points in $P(\mathcal{K})$, whose objective value approached the unattained optimal value.

4
Branch-and-bound

For coefficients $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, a *Cartesian product of linear, second-order and semidefinite cones* $\mathcal{K} \subseteq \mathbb{R}^m$, and the index set $\mathcal{I} \subseteq \{1, \ldots, n\}$ of integer-constrained variables, consider the mixed-integer conic optimization problem,

$$\begin{array}{lrcll} \underset{x}{\text{minimize}} & c^T x & & & \\ \text{subject to} & Ax - b & \in & \mathcal{K}, & \\ & x_j & \in & \mathbb{Z} \cap [l_j, u_j], & \text{for all } j \in \mathcal{I}, \end{array} \tag{4.1}$$

where $l_j, u_j \in \mathbb{Z}$ for all $j \in \mathcal{I}$ represent a *bounded* integer variable domain.

The formulation in (4.1) may be recognized as a restricted form of the mixed-integer formulation in (1.1), where the choice of cones and variable bounds have been limited. These limitations enable us to design finitely converging algorithms for solving (4.1). In particular, under these two limitations, it is possible to enumerate all integer assignments in finite amount of time, compute the optimal value for each integer assignment (by Corollary 11), and compare them to identify the best. The only additional requirement needed for this simple strategy to succeed is a specification of the comparison operator, considering that optimal values can be unattained.

Definition 14 (Optimal value pairs). *Let optimal value pairs be members,*

$$\psi \in (\mathbb{R} \cup \{-\infty, +\infty\}) \times \{\texttt{false}, \texttt{true}\},$$

where $\psi = (\theta, \mathit{attained})$ *holds the optimal value, and whether it is attained, for an optimization problem. Associated with these pairs are the comparison operators* ($\prec_{\min}$. $\preceq_{\min}$, $\succeq_{\min}$, $\succ_{\min}$) *from the lexicographical order for which* $\texttt{true} < \texttt{false}$, *respectively the comparison operators* ($\prec_{\max}$. $\preceq_{\max}$, $\succ_{\max}$, $\succeq_{\max}$) *for which* $\texttt{true} > \texttt{false}$.

With this definition, the left-hand-side of a satisfied relation $\psi_1 \preceq_{\min} \psi_2$ is always preferred when minimizing, respectively the right-hand-side of a satisfied relation $\psi_1 \preceq_{\max} \psi_2$ when maximizing. Optimal value pairs, and the lexicographical comparison operators of Definition 14, is on abstraction to differentiate between attained and unattained optimal values. Another one would be to redefine optimal values in terms of suprema and infima in the hyperreal numbering system which would allow infinitesimal differences. In both cases, a so-called *standard part* function to "round off" these abstractions to the standard numbering system used till now, i.e., the extended reals, will be useful in latter derivations. In the abstraction of optimal value pairs, this function is defined as follows.

Definition 15. *The standard part of an optimal value pair,* $\psi = (\theta, \mathit{attained})$, *is defined as*

$$\operatorname{st} \psi := \begin{cases} (\theta, \texttt{true}), & \textit{if } \theta \textit{ is finite}, \\ (\theta, \texttt{false}), & \textit{otherwise}. \end{cases}$$

Corollary 15. *The standard part function acts to reduce lexicographical comparison on optimal value pairs, to the corresponding comparison on optimal values. Taking* $\psi' = (\theta', \mathit{attained}')$ *and* $\psi'' = (\theta'', \mathit{attained}'')$ *as example,*

$$[\operatorname{st} \psi' \preceq_{\min} \operatorname{st} \psi''] \iff [\operatorname{st} \psi' \preceq_{\max} \operatorname{st} \psi''] \iff [\theta' \leq \theta''].$$

Branch-and-bound The *branch-and-bound* algorithm, usually attributed to [26], is a divide-and-conquer strategy which can be deployed to implicitly enumerate all integer assignments. It iteratively partitions the integer variable domain, defined by $l_j, u_j \in \mathbb{Z}$ for all $j \in \mathcal{I}$, and solves a relaxation of (4.1) on this smaller domain. This approach can be graphically represented by a *search tree* as illustrated in Figure 4.1. The so-called *root node* of this search tree holds the original problem and is partitioned into subproblems, which may themselves be partitioned further as needed.

The major advantage of branch-and-bound over explicit enumeration is that the relaxations provides a bound on how good the optimal value pair can be on any given partition, as well as a qualified guess at what the optimal solution could look like. The following list represent examples of how this information can be used.

1. Entire partitions can be compared based on the bounding value pair enabling us to guide the search towards the best-looking partitions.

2. Entire partitions can be disregarded (i.e., pruned from the search tree) if the bound encodes infeasibility or simply represents no improvement over the value pair of an already known solution.

3. The qualified guesses for integer assignments $x_{\mathcal{I}}$ may reveal the optimal solution very early in the search process, although further processing may be needed before its optimality can be established.

A basic general purpose branch-and-bound algorithm for the mixed-integer conic optimization problem (4.1) is given by Algorithm 3, followed by a proof of its correctness under the following natural assumption.

Assumption 1. *The considered relaxations allow line 5 and 7 in Algorithm 3 to be executed. That is, allow one to*

1. *Compute optimal value pairs for the relaxation.*

2. *Extract a (possibly fractional) assignment of integer variables that attains the optimal value pair of the relaxation whenever it is feasible.*

This assumption is elaborated in Section 4.1 and Section 4.2 where it is shown satisfiable for both the continuous relaxation and the considered linear relaxations of (4.1). Note that Algorithm 3, for simplicity, only finds the optimal value pair even though the corresponding integer assignment and other certificates from [Friberg 81] (such as a primal-dual optimal solution for (4.1) at this integer assignment) can easily be registered as well.

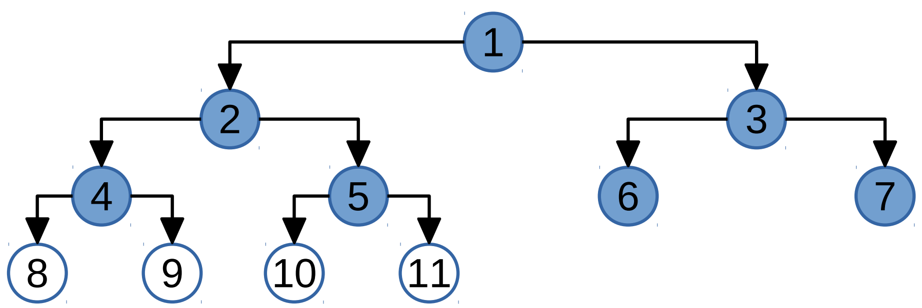


Figure 4.1: The search tree in midst of the branch-and-bound algorithm. The root node with index 1 holds the original problem. Nodes 8–11 are in the queue waiting to be processed, while the subproblems of nodes 6–7 needed no additional processing as inferred from the solved relaxations.

Algorithm 3: A general branch-and-bound algorithm to solve the mixed-integer conic minimization problem (4.1).

1 Let $(\theta^\star, \mathit{attained}^\star) \leftarrow (\infty, \texttt{false})$.
2 Create and queue node holding the minimization problem (4.1).
3 **repeat**
4   Take out node from the queue for processing.
5   Compute bounding value pair $(\theta', \mathit{attained}')$ for node based on some relaxation.
6   **if** $(\theta', \mathit{attained}') \prec_{\min} (\theta^\star, \mathit{attained}^\star)$ **then**
7   Extract attaining integer-variable assignment $x'_{\mathcal{I}}$ for the relaxation.
8   **if** $x'_{\mathcal{I}} \notin \mathbb{Z}^{|\mathcal{I}|}$ **then**
9   Create and queue nodes holding the subproblems obtained by partitioning the integer variable domain to exclude the fractional assignment $x'_{\mathcal{I}}$.
10   **else**
11   Compute optimal value pair $(\theta'', \mathit{attained}'')$ for (4.1) with $x_{\mathcal{I}} = x'_{\mathcal{I}}$.
12   **if** $(\theta'', \mathit{attained}'') \prec_{\min} (\theta^\star, \mathit{attained}^\star)$ **then**
13   Let $(\theta^\star, \mathit{attained}^\star) \leftarrow (\theta'', \mathit{attained}'')$.
14   **if** $(\theta', \mathit{attained}') \prec_{\min} (\theta'', \mathit{attained}'')$ **then**
15   Create and queue nodes holding the subproblems obtained by partitioning the integer variable domain to either exclude the assignment $x'_{\mathcal{I}}$ or locally improve the relaxation such that $x'_{\mathcal{I}}$ is valued as $(\theta'', \mathit{attained}'')$.
16 **until** *Until there are no more nodes in the queue*;

Theorem 5. *Under Assumption 1, Algorithm 3 terminates in finitely many iterations with the optimal value pair of the mixed-integer conic optimization problem* (4.1).

Proof. Finite termination. If the condition on line 6 is not satisfied, a node—and hence a partition of integer assignments—is pruned from the search tree. Otherwise, line 7 is executed to retrieve an integer-assignment $x'_{\mathcal{I}}$ and finite terminate follows from a bounded integer variable domain in (4.1) by showing that any such integer-assignment $x'_{\mathcal{I}}$ can be visited at most twice. Note that there are only three possible execution paths for any $x'_{\mathcal{I}}$. If line 9 is executed or the condition on line 14 is not satisfied, then $x'_{\mathcal{I}}$ is excluded from all remaining nodes on the queue. Otherwise line 15 is executed to either exclude $x'_{\mathcal{I}}$ directly or make sure that all relaxations used on the partition holding $x'_{\mathcal{I}}$ value this assignment as $(\theta'', \mathit{attained}'')$, which is a value pair that will be pruned on line 6 if ever revisited as guaranteed by line 12-13.

Optimal value pair. The value pair $(\theta^\star, \mathit{attained}^\star)$ is initialized on line 1 to indicate infeasibility, and otherwise only modified on line 13 if an integer assignment with a better value pair (with respect to Definition 14) has been established, based on Corollary 11, on line 11. Optimality of the value pair $(\theta^\star, \mathit{attained}^\star)$ at termination thus follows by showing that all integer assignments for which line 13 is not executed have value pairs inferior to $(\theta^\star, \mathit{attained}^\star)$. This is the case whenever nodes—and hence a partition of integer assignment—are pruned from the search tree by failing to satisfy the condition on line 6, as the bounding value pair is better than all integer assignments in a partition by definition. This is also the case whenever nodes are pruned from the search tree by failing to satisfy the condition on line 14 by the same argument, noting that $(\theta^\star, \mathit{attained}^\star) \preceq_{\min} (\theta'', \mathit{attained}'') = (\theta', \mathit{attained}')$, the former by line 12-13, the latter by relative complement of the condition on line 14, noting that $(\theta', \mathit{attained}') \preceq_{\min} (\theta'', \mathit{attained}'')$ holds by definition of the bounding value pair. Finally, an integer assignment may be pruned by exclusion on line 15, but only after updating $(\theta^\star, \mathit{attained}^\star)$ based on its value on lines line 11-13. The claim hence follows as these are the only lines that may cause integer assignments to be pruned. $\square$

There is a wealth of additional techniques that can be added to Algorithm 3 such as presolving, cut generation and primal heuristics (see, e.g. [2] for a survey), as well as search tree restructuring [106], advanced pruning (e.g., symmetry detection [23], conflict analysis [2] and pruning moves [33]), and more. These techniques have an important impact on solving problems in practice [88], but also significantly complicate the implementation of the algorithm. In particular, Algorithm 3 already possess many degrees of freedom regarding node selection (line 4) and branching decisions (line 9) (as witnessed, e.g., in [2]) as well as in the choice of relaxation (line 5) and whether (or how) to improve it on line 15.

The last point here, namely the choice and usage of relaxations left open by the specification of Algorithm 3, is elaborated in the following sections.

4.1 Continuous relaxation bounds

The continuous relaxation of (4.1) is given by

$$\begin{array}{lrcll} \underset{x}{\text{minimize}} & c^T x & & & \\ \text{subject to} & Ax - b & \in & \mathcal{K}, & \\ & x_j & \in & \mathbb{R} \cap [l_j, u_j], & \text{for all } j \in \mathcal{I}, \end{array} \tag{4.2}$$

but can easily be rewritten to conform with the specification of the primal-dual pair (2.11), such that the results of of the previous chapters apply. This demonstration is omitted. Instead, it is now argued that continuous relaxations can be solved in satisfaction of Assumption 1.

Proposition 24. *Let $P(\mathcal{K})$ represent the continuous relaxation* (4.2)*. Given $P(\mathcal{K})$, the interleaved regularization and optimization algorithm of Chapter 3 finds a relaxation $P(\mathcal{C})$, and*

1. *Compute the optimal value pair for $P(\mathcal{C})$.*

2. *Extract a (possibly fractional) assignment of integer variables that attains the optimal value pair of $P(\mathcal{C})$ whenever it is feasible.*

Proof. That it finds a relaxation $P(\mathcal{C})$ of $P(\mathcal{K})$ is shown by Corollary 12, Corollary 13 and Corollary 14. The statements follows by Theorem 4 noting that feasible points of $P(\mathcal{C})$ in all certificates of Definition 13 except the infeasibility certificate. $\square$

Note that the optimal value of $P(\mathcal{C})$ and $P(\mathcal{K})$ are identical in Proposition 24, and that their feasible sets actually only differ in the special case of Corollary 14. This special case was examined in Proposition 23, and appears whenever the continuous relaxation has a finite but unattained optimal value, or is unbounded without improving ray. Beware that the difference in feasible sets for $P(\mathcal{C})$ and $P(\mathcal{K})$ in this special case, represents a potential pitfall that was avoided in the following proposed enhancements of the branch-and-bound method.

The first family of possible enhancements to Algorithm 3 comes from the fact that additional restrictions of a problem does not invalidate previously valid facial reductions.

Proposition 25. *Consider a conic constraint* (2.3) *with conic domain $\mathcal{K} \subseteq \mathbb{R}^r$, present in two optimization problems P_1 and P_2 with feasible domains $\Omega_1 \subseteq \Omega_2 \subseteq \mathbb{R}^n$. Then a valid facial reduction of $\mathcal{K}$ in P_2 is also a valid facial reduction of $\mathcal{K}$ in P_1.*

Proof. Let ρ equal the affine map of the conic constraint (2.3) which is then expressed by $\rho \in \mathcal{K}$, and let Ω_i^ρ denote the feasible set of P_i projected onto ρ for $i \in \{1, 2\}$. A valid facial reduction in P_i from $\mathcal{K}$ to $\mathcal{C}$ is defined in Definition 8 to satisfy $\mathcal{C} \trianglelefteq \mathcal{K}$ and $\Omega_i^\rho \subseteq \mathcal{C} \subseteq \mathcal{K}$. Hence, the claim follows because $\Omega_1^\rho \subseteq \Omega_2^\rho$ is implied by $\Omega_1 \subseteq \Omega_2$. $\square$

First of all, this means that we can apply all primal facial reductions identified for the continuous relaxation (4.2), to the corresponding mixed-integer problem it relaxes and to all subsequent subproblems constructed from it by branching.

Corollary 16. *All primal facial reductions identified for the continuous relaxation on line 5 of Algorithm 3, may be kept in the subproblems constructed by partitioning the integer variable domain on line 9 and 15 of Algorithm 3.*

Secondly, it leads to the fact that several interleaved regularization and optimization solves can be skipped entirely to significantly improve expected performance.

Proposition 26. *If the continuous relaxation on line 5 of Algorithm 3 can be solved with no— or only primal—facial reduction, the cone, certificate and optimal value identified to satisfy Theorem 4, also satisfy Theorem 4 for the corresponding problem on line 11 of Algorithm 3.*

Proof. Let $P_1(\mathcal{K})$ be the continuous relaxation (the problem from line 5) and let $P_2(\mathcal{K})$ be $P_1(\mathcal{K})$ after adding equations of the form $x_{\mathcal{I}} = x'_{\mathcal{I}}$ (the problem from line 11). Solving $P_1(\mathcal{K})$ yields a certificate of $P_1(\mathcal{C})$ for some $\mathcal{C}$ as in Theorem 4. Adding $x_{\mathcal{I}} = x'_{\mathcal{I}}$ to $P_1(\mathcal{C})$ does not change its feasible set by definition of $x'_{\mathcal{I}}$ on line 7. Moreover, it does not change the set of rays (because all integer variables are bounded from below and above by definition of (4.1)), and only adds variables in the dual problem which can be fixed to zero. Hence, the certificate also satisfies Theorem 4 for $P_2(\mathcal{C})$ and the claim follows because $P_2(\mathcal{C})$ can be reached from $P_2(\mathcal{K})$ through primal facial reduction following Proposition 25. $\square$

While Proposition 26 showed a class of problems for which line 11 of Algorithm 3 could be skipped, thereby showing the condition on line 14 to be false, there is also a class of problems for which line 11 and the condition on line 14 are needed. This is the continuous relaxations for which dual regularization is needed, as illustrated by the following example.

Example 5 (Triggering the condition on line 14)**.** *Consider appending $x_6 \in \mathbb{Z}_+$ to Example 3 such that its continuous relaxation becomes the primal problem of the primal-dual pair,*

$$\begin{array}{ll} \text{minimize} & x_3 + x_6 \\ \text{subject to} & x_1 + x_2 + x_4 + x_5 - x_6 = 0 \\ & -x_3 + x_4 = 1 \\ & x \in \mathcal{Q}^3 \times \mathcal{Q}^2 \times \mathbb{R}_+, \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \begin{pmatrix} 0\\0\\1\\0\\0\\1 \end{pmatrix} - \begin{pmatrix} y_1\\y_1\\-y_2\\y_1+y_2\\y_1\\-y_1 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3 \times \mathcal{Q}^2 \times \mathbb{R}_+. \end{array} \tag{4.3}$$

This primal-dual pair can be solved analytically. For $x_6 = 0$, we have $x_3 = 0$ by Example 3. For $x_6 > 0$, we have the family of solutions given by $x_1 = \sqrt{x_2^2 + x_3^2}$, $x_2 = (x_6^2 - 1)/(2x_6)$ and $x_3 = -1$ (such that $x_1 + x_2 = x_6$) as well as $x_4 = x_5 = 0$. Note, in this family, that x_3 attains its lower bound induced by $x_3 = x_4 - 1 \geq -1$ from $x_4 \geq 0$ The primal problem thus has a finite but unattained infimum value of -1, approached by positive $x_6 \to 0$, and strong duality holds by feasibility of $s = (0, 0, 0, 1, 0, 1)$, $y = (0, -1)$ in the dual problem.

Despite of strong duality, dual regularization is needed by Algorithm 2 to solve the primal problem as shown by Proposition 23. Taking the valid equality $s_1 - s_2 = 0$ as basis, it is possible to reformulate the conic constraint $\left(\begin{smallmatrix} s_1\\s_2\\s_3 \end{smallmatrix}\right) \in \mathcal{Q}^3$ as

$$\begin{pmatrix} s_1\\s_2\\s_3 \end{pmatrix} \in \mathcal{Q}^3 \cap \begin{pmatrix} 1\\-1\\0 \end{pmatrix}^{\perp}, \tag{4.4}$$

because $\left(\begin{smallmatrix} s_1\\s_2\\s_3 \end{smallmatrix}\right) \in \left(\begin{smallmatrix} 1\\-1\\0 \end{smallmatrix}\right)^{\perp}$ is just another way of writing $\left(\begin{smallmatrix} 1\\-1\\0 \end{smallmatrix}\right)^T \left(\begin{smallmatrix} s_1\\s_2\\s_3 \end{smallmatrix}\right) = s_1 - s_2 = 0$. In turn, given $\mathcal{Q}^3 \cap \left(\begin{smallmatrix} 1\\-1\\0 \end{smallmatrix}\right)^{\perp} = \left(\begin{smallmatrix} 1\\1\\0 \end{smallmatrix}\right) \mathbb{R}_+$ (see [Friberg 38]) with dual cone $\left\{\left(\begin{smallmatrix} 1\\1\\0 \end{smallmatrix}\right)\right\}^*$, this reformulation *changes the primal conic constraint* $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in \mathcal{Q}^3$ *into* $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in \left\{ \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \right\}^*$*, equivalent the linear constraint* $x_1 + x_2 \geq 0$*. Hence, the dual facial reduction enlarges the primal feasible set such that the optimal value of* -1 *becomes attained, e.g., by* $x = (0, 0, -1, 0, 0, 0)$*.*

Recall that $x_6 \in \mathbb{Z}_+$*. When this mixed-integer conic optimization problem is solved by Algorithm 3, the continuous relaxation is solved as above to yield the bounding value pair* $(-1, \texttt{false})$ *on line 5 and the integer assignment* $x_6 = 0$ *on line 7. On line 11, we hence obtain and solve the problem of Example 3 yielding the inferior value pair* $(0, \texttt{true})$ *whereby the condition on line 14 is triggered.*

When the condition on line 14 is triggered, line 15 of Algorithm 3 is executed. As stated in Algorithm 3, one possibility on line 15 is to partition the integer variable domain to exclude the currently considered integer assignment. An expected improvement over this generic exclusion approach can also be achieved, however, focusing the partitioning effort to facilitate local changes to the continuous relaxation as indicated by the following theorem.

Theorem 6. *If the condition on line 14 of Algorithm 3 is satisfied, the evaluation problem on line 11 has duality gap-closing primal facial reductions which were not valid primal facial reductions for the continuous relaxation on line 5.*

Proof. Let $P_1(\mathcal{K})$ be the continuous relaxation (the problem from line 5) and let $P_2(\mathcal{K})$ be $P_1(\mathcal{K})$ after adding equations of the form $x_\mathcal{I} = x'_\mathcal{I}$ (the problem from line 11). By satisfaction of condition line 14 and Proposition 26, solving $P_1(\mathcal{K})$ first goes through a primal facial reduction phase ending with some other cone $\mathcal{C}'$, and then through a dual facial reduction phase ending with some cone $\mathcal{C}''$. We will show that the optimal value pair of $P_2(\mathcal{K})$ and $P_2(\mathcal{C}')$ is the same, say $(\theta_2, \mathit{attained}_2)$, and that the optimal value pair of its dual problem, denoted $D_2(\mathcal{C}')$, and $P_1(\mathcal{K})$ is the same, say $(\theta_1, \mathit{attained}_1)$. Hence, if the condition on line 14 is satisfied and there is a difference between the optimal value pairs of $P_1(\mathcal{K})$ and $P_2(\mathcal{K})$, that is

$$(\theta_1, \mathit{attained}_1) \prec_{\min} (\theta_2, \mathit{attained}_2),$$

then $\theta_1 < \theta_2$ is the only option since $\mathit{attained}_1 = \texttt{false}$ by a need of dual facial reduction (see Proposition 23). That is, there is a positive duality gap between $P_2(\mathcal{C}')$ and $D_2(\mathcal{C}')$. This shows existence of duality gap-closing primal facial reductions for $P_2(\mathcal{C}')$, which did not exist for $P_1(\mathcal{C}')$ since the dual facial reduction phase initiated (see Proposition 21, statement 3).

First, the optimal value pair of $P_2(\mathcal{K})$ and $P_2(\mathcal{C}')$ is the same because, by Proposition 25, the primal facial reductions used on $P_1(\mathcal{K})$ can also be used on $P_2(\mathcal{K})$.

Secondly, note that the optimal value pair for $P_1(\mathcal{K})$ is the same as for the pair $P_1(\mathcal{C}'')$ and $D_1(\mathcal{C}'')$. Exactly as in Proposition 26, adding $x_\mathcal{I} = x'_\mathcal{I}$ does not violate any part of the certificate returned for $P_1(\mathcal{C}'')$. Hence, this certificate is also valid for the pair $P_2(\mathcal{C}'')$ and $D_2(\mathcal{C}'')$, and equivalence of the optimal value pair for $P_1(\mathcal{K})$ and $D_2(\mathcal{C}'')$ have now been shown. Finally, to see that the optimal value pair of $D_2(\mathcal{C}'')$ and $D_2(\mathcal{C}')$ is the same, note that facial reduction certificates used on $D_1(\mathcal{C}')$ to obtain $D_1(\mathcal{C}'')$ can also be used on $D_2(\mathcal{C}')$ to obtain $D_2(\mathcal{C}'')$. This holds because the equations $x_\mathcal{I} = x'_\mathcal{I}$ are redundant in the description of primal non-improving rays (Definition 10), and hence redundant to the availability of dual facial reductions, since the integer variable domain is bounded (i.e., $x_\mathcal{I} = 0$ already holds for all rays). $\square$

Theorem 6 states that if the condition on line 14 of Algorithm 3 is satisfied, the addition of the equations, $x_\mathcal{I} = x'_\mathcal{I}$, to the continuous relaxation enlarges its set of valid and necessary primal facial reductions. Hence, it also suggests the existence of a minimal set of equations, say $x_{\mathcal{I}^\star} = x'_{\mathcal{I}^\star}$, for which these primal facial reductions become valid and necessary in the continuous relaxation. This leads to the strategy of partitioning the integer variable domain to obtain a subproblem for which $x_{\mathcal{I}^\star} = x'_{\mathcal{I}^\star}$ holds, and for which the continuous relaxation and the evaluation problem value the integer assignment $x_\mathcal{I} = x'_\mathcal{I}$ equivalently. Formally:

Corollary 17. *Suppose the condition on line 14 of Algorithm 3 is satisfied, and let $\mathcal{I}^\star$ be the subset of integer variables whose equation in the integer assignment $x_{\mathcal{I}} = x'_{\mathcal{I}}$ is used in the primal facial reduction phase when solving the evaluation problem on line 11 with Algorithm 2. Let further $(\theta'', attained'')$ be the optimal value pair of this evaluation problem. Then the continuous relaxation from line 5, restricted to a smaller partition where $x_{\mathcal{I}^\star} = x'_{\mathcal{I}^\star}$ holds, will give the bounding value pair $(\theta'', attained'')$ if ever proposing the integer assignment $x_{\mathcal{I}} = x'_{\mathcal{I}}$ again.*

Proof. All primal facial reductions needed to solve the continuous relaxation with $x_{\mathcal{I}} = x'_{\mathcal{I}}$ (the evaluation problem on the smaller partition), are also valid primal facial reductions for the continuous relaxation with $x_{\mathcal{I}^\star} = x'_{\mathcal{I}^\star}$ (the continuous relaxation on the smaller partition). Hence, by Theorem 6, the condition on line 14 cannot be satisfied and the optimal value pairs of the two problems must be identical. □

The partitioning strategy of Corollary 17, needed to make $x_{\mathcal{I}^\star} = x'_{\mathcal{I}^\star}$ explicit in one of the subproblems, is expected to generate significantly fewer subproblems than the partitioning needed to exclude $x_{\mathcal{I}} = x'_{\mathcal{I}}$. Moreover, the subset $\mathcal{I}^\star$ can be assembled by looking at the primal facial certificates generated by Algorithm 2, which are all members of $\mathring{\Omega}_D$ (as defined in Definition 10) for an iteratively changing cone. In particular, for any $(s, y) \in \mathring{\Omega}_D$, the nonzero values in the vector y identifies the equations used, or in other words, the equations needed for the facial reduction to be valid.

4.2 Linear relaxation bounds

Applying linear outer approximation to all nonlinear Cartesian factors of $\mathcal{K}$ to define $\mathcal{C} \supseteq \mathcal{K}$, e.g., using subgradients as described in [Friberg 39], gives life to a family of linear relaxations of (4.1) on the form

$$\begin{array}{lll} \underset{x}{\text{minimize}} & c^T x & \\ \text{subject to} & Ax - b \;\in\; \mathcal{C}, & \\ & \quad\;\; x_j \;\in\; \mathbb{R} \cap [l_j, u_j], & \text{for all } j \in \mathcal{I}. \end{array} \tag{4.5}$$

In order to solve mixed-integer conic optimization problems with the branch-and-bound method of Algorithm 3, using relaxations of the form (4.5), the algorithm considered for solving these linear optimization problems has to satisfy Assumption 1. Rather than going into the details of the simplex method [24] or its variants (e.g., [66]), which are often associated with branch-and-bound algorithms, we will simply note that this assumption is satisfied by the interleaved regularization and optimization method of Chapter 3. This is seen by the arguments of the previous Section 4.1, noting that the linear relaxation (4.5) is a special case of (4.2). A notable difference between (4.5) and (4.2), however, is that regularization is never used to solve the linear relaxations as established by Proposition 22. Regarding how to build and refine the linear relaxation, the following results can be used.

Proposition 27 ([Friberg 39, Corollary 2])**.** *A subgradient-based outer approximation of a non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^n$ is given by $\mathcal{C} = \hat{\Omega}^* = \{x : \xi^T x \geq 0 \text{ for } \xi \in \hat{\Omega}\}$, that is $\mathcal{C} = \hat{\Omega}^* \supseteq \mathcal{K}$, for any finite subset $\hat{\Omega} \subseteq \mathcal{K}^*$.*

One simple way to build or refine the linear relaxation is thus to fill $\hat{\Omega}$ with points and rays found for the dual problem of the continuous relaxation or the evaluation problem. The following corollary concerns this practice using the continuous relaxation.

Corollary 18. *Let* (CRP) *denote the continuous relaxation* (4.2) *with dual problem* (CRD)*. By relaxation, all feasible points and rays of* (CRP) *are also feasible in the linear relaxation. Hence, the following statement holds when using $\mathcal{C} = \hat{\Omega}^*$ from Proposition 27:*

- *If primal regularization is unnecessary to solve* (CRP)*, then any certificate for Theorem 4 returned by Algorithm 2 for* (CRP)*, can be made valid for the linear relaxation by adding the returned dual point and/or ray to* $\hat{\Omega}$*.*

Another way to build or refine the linear relaxation is to fill $\hat{\Omega}$ with generators of subgradient inequalities supporting $\mathcal{K}$ at boundary points $\hat{x} \in \operatorname{bnd} \mathcal{K}$. These points can be found either by projection of points in $\mathcal{C} \setminus \mathcal{K}$ from the linear relaxation itself, or from points and rays for the primal problem of the continuous relaxation or the evaluation problem.

Proposition 28. *The subgradient inequalities supporting* $\mathcal{K}$ *at boundary points* $\hat{x} \in \operatorname{bnd} \mathcal{K}$*, is given by* $\xi^T x \geq 0$ *for* $\xi \in \mathcal{K}^* \cap \hat{x}^{\perp}$*.*

Proof. By the derivation of [Friberg 39, Theorem 1] based on [86, Corollary 23.5.4]. □

The latter way is used in [67] for mixed-integer semidefinite optimization, and a mixture of the two are used in [27] for mixed-integer second-order cone optimization. The construction and refinement of linear relaxations has not been the focus of this thesis, but a few pointers from existing literature can be given follows.

In [1, Section 3.3] it is stated that points from the continuous relaxation yields more valuable subgradient inequalities in Proposition 28 than points from the evaluation problem, but that they typically also require a higher computational effort to obtain.

In [43], two principles leading to size-effective outer approximations are listed. Decomposition, or disaggregation, of nonlinearities (see also [93]), and projections of extended formulations. Beginning with the tower of variables decomposition and symmetry-exploiting extended formulation constructed in [11], many alternatives based on these principles have appeared [11, 43, 98, 99].

Considering Algorithm 3, one important enhancement when using linear relaxations is on line 15 where the default behavior, as stated, is to partition the integer variable domain to exclude the currently considered integer assignment. The alternative of improving the linear relaxation is not a simple task, however, as there are several cases that must be considered when line 15 is executed. This characterization of cases most notably depends on the continuous relaxation, and uses the following property of the standard function from Definition 15.

Lemma 5. *Let* ψ *be the optimal value pair of a linear optimization problem. Then the standard part function makes no transformation,*

$$\operatorname{st} \psi = \psi.$$

Proof. By inspection of Definition 15 given attainment of all finite optimal values (Proposition 22) and unattainment of all infinite optimal values. □

Theorem 7. *Let* (LRP) *and* (CRP) *be the linear and continuous relaxation, respectively, on line 5 of Algorithm 3, with dual problems* (LRD) *and* (CRD)*, and let* (EVAL) *be the evaluation problem on line 11 obtained by the integer assignment from* (LRP)*. If the condition on line 14 of Algorithm 3 detects* $\psi_{(\mathrm{LRP})} \prec_{\min} \psi_{(\mathrm{EVAL})}$*, then*

$$\psi_{(\mathrm{LRP})} = \operatorname{st} \psi_{(\mathrm{LRD})} \preceq_{\min} \operatorname{st} \psi_{(\mathrm{CRD})} \preceq_{\min} \operatorname{st} \psi_{(\mathrm{CRP})} \preceq_{\min} \psi_{(\mathrm{CRP})} \preceq_{\min} \psi_{(\mathrm{EVAL})} \tag{4.6}$$

holds with strict inequality in one or more places. In particular, all four inequalities represent independent reasons for triggering the execution of line 15 of Algorithm 3.

Proof. Strong duality holds for (LRP), as failure of strong duality would imply infeasibility by arguments following Definition 2 leading to inconsistency: $\psi_{(\mathrm{LRP})} = (\infty, \texttt{false}) \prec_{\min} \psi_{(\mathrm{EVAL})}$. Hence, by Corollary 15 and [Friberg 39, Proposition 1] the following relation holds

$$\psi_{(\mathrm{LRP})} = \operatorname{st} \psi_{(\mathrm{LRP})} = \operatorname{st} \psi_{(\mathrm{LRD})} \preceq_{\min} \operatorname{st} \psi_{(\mathrm{CRD})} \preceq_{\min} \operatorname{st} \psi_{(\mathrm{CRP})},$$

first step by Lemma 5. Finally, $\operatorname{st}\psi_{(\mathrm{CRP})} \preceq_{\min} \psi_{(\mathrm{CRP})}$ holds for any optimal value pair, and $\psi_{(\mathrm{CRP})} \preceq_{\min} \psi_{(\mathrm{EVAL})}$ follows by relaxation of the feasible set from right to left.

A difference $\psi_{(\mathrm{CRP})} \prec_{\min} \psi_{(\mathrm{EVAL})}$ may be caused by inferiority of the integer assignment identified by (LRP), compared to that which would have been found by (CRP). Otherwise, the difference is caused by necessary duality-gap closing primal facial reductions which are not valid on the current partition of the integer variable domain as formalized in Theorem 6.

A difference $\operatorname{st}\psi_{(\mathrm{CRP})} \prec_{\min} \psi_{(\mathrm{CRP})}$ is caused by (CRP) having a finite but unattained optimal value which cannot be represented by linear relaxations according to Proposition 22.

A difference $\operatorname{st}\psi_{(\mathrm{CRD})} \prec_{\min} \operatorname{st}\psi_{(\mathrm{CRP})}$ is caused by lack of strong duality in the continuous relaxation by Corollary 15 and Definition 2.

A difference $\operatorname{st}\psi_{(\mathrm{LRD})} \prec_{\min} \operatorname{st}\psi_{(\mathrm{CRD})}$ may be caused insufficient refinement of the linear relaxation by Proposition 27. Otherwise, primal facial reduction is necessary to solve (CRP) and dual facial reduction is necessary to solve (CRD). □

As already stated, the construction and refinement of linear relaxations has not been the focus of this thesis, but for completion, one option on line 15 of Algorithm 3 is mentioned, namely to jump to line 5 and solve the continuous relaxation. This step can also be taken already when the integer variable assignment is found not to be fractional on line 8. The key to avoid getting stuck with the continuous relaxation is to refine the linear relaxation. This is described by the following steps needed after finished processing of the node with the continuous relaxation.

1. Refine the linear relaxation, using Corollary 18, around the continuous relaxation as it appears after having applied all the primal facial reductions that was used to solve it. This ensures
$$\psi_{(\mathrm{LRP})} = \operatorname{st}\psi_{(\mathrm{LRD})} = \operatorname{st}\psi_{(\mathrm{CRD})} = \operatorname{st}\psi_{(\mathrm{CRP})}$$
on the current node, but may not hold on subproblems however.

2. If line 15 of Algorithm 3 was executed using the continuous relaxation, the steps of Corollary 17 constructed a partition containing the integer assignment for which all primal facial reductions used to solve the evaluation problem are valid. Hence, the linear relaxation on this partition can be further refined, using Corollary 18, around the continuous relaxation as it appears after having applied all the primal facial reductions that was used to solve the evaluation problem.

Finally, it should be mentioned that differences in the first part of Theorem 7,
$$\psi_{(\mathrm{LRP})} = \operatorname{st}\psi_{(\mathrm{LRD})} \preceq_{\min} \operatorname{st}\psi_{(\mathrm{CRD})} \preceq_{\min} \operatorname{st}\psi_{(\mathrm{CRP})},$$
should preferably be kept small throughout the branch-and-bound algorithm. Notably, for branch-and-bound algorithms guided solely by the optimal value of linear relaxations, differences in the above relation may cause a far greater number of nodes to be explored than actually needed. Hence, such algorithms may benefit from the facial reduction heuristics developed and discussed in [Friberg 39], which are based on simple analysis and do not require conic optimization problems to be solved.

5
Conclusions and future work

The thesis makes several pedagogical, practical and theoretical contributions extending well beyond the main topic and scope given by mixed-integer second-order cone optimization.

(a) The introduced MIN-VAR mnemonic for dualization by hand (Section 2.4) and the proposed head and base terminology for conic entries (Section 2.5) are small pedagogical contributions that makes it easier to work with, talk about and educate in conic optimization. The former adds to the list of simple results in conic optimization which may help in making the jump from linear optimization less intimidating; other simple results includes the proof of weak duality (Proposition 7) and the conclusions on strong duality in linear optimization to be drawn from the generalized Slater condition (Proposition 8). The latter contribution, regarding the head and base terminology, not only increase expressibility in spoken and written language but also remove ambiguity. For instances, if someone referred to the first or last entry of the quadratic cone you would first have to determine whether they define the quadratic cone as $\{x \in \mathbb{R}^n : x_1 \geq \|x_{2:n}\|_2\}$ or $\{x \in \mathbb{R}^n : \|x_{1:n-1}\|_2 \leq x_n\}$ because both definitions are common.

(b) The developed CBF file format (Section 1.1) and CBLIB library (Section 1.2) have generally been well received by the optimization community, and have found several usages already. One usage worth highlighting is the online second-order cone benchmark tests carried out by Mittelmann [69]. Before this thesis, he would maintain different sets of files for different sets of solvers to cope with their differences, making it unclear whether the solvers were actually solving the same problem. Now he only needs to maintain one version of each problem instance in the CBF file format, and use the developed conversion tools to obtain the respective file formats compatibility with each solver. This approach is transparent, reproducible and unambiguous, and gives an example of how the CBF file format can be used to make cross-solver benchmarking easier.

(c) The theoretically established interleaved regularization and optimization method (Chapter 3) adds the element of performance to previous facial reduction algorithms—and maximizes the amount primal and dual information returned—by being able to distinguish whether primal (resp. dual) regularization is necessary or not. It is moreover a complete regularization method in the sense that it handles all subtleties such as unattainment, unboundedness without improving ray and infeasibility without dual improving ray. This completeness is reflected in the fact that it was possible to establish a reliable and provably converging branch-and-bound method based on the outputs of the interleaved regularization and optimization method. This my knowledge, this is the first theoretical study of ill-posedness in mixed-integer optimization.

(d) The new facial reduction heuristics ([Friberg 39]) were developed with the main motivation of breaking dependencies in regularization on conic systems and the limiting behavior of numerically converging algorithms. This allows these heuristics to be used in combination with branch-and-bound methods based solely on linear relaxations, which are notably also affected by ill-posedness as proven. Moreover, as these heuristics rely only on linear algebra, this allows some or most facial reductions to be carried out in higher numerical accuracy than otherwise possible, e.g, using interleaved regularization and optimization. Worth highlighting are the two heuristics based, respectively, on the basic principles of forcing constraints and linear dependencies. Subgradient matching integrates with domain propagation to not only detect facial reductions in forcing constraints, but also strengthen the performed bound analysis. Despite its simplicity, it was shown useful on various examples of ill-posedness and goes beyond second-order cones. Single-cone analysis, on the other hand, was only explored on second-order cones. In its original formulation it used the general subspace intersections of [Friberg 38] to eliminate linear dependencies as in Proposition 3, but was later revised to solve a least-norm problem pinpointing the particular dependency (if any) of relevance to facial reduction. Notably, while subgradient matching focus on variable bound information and ignores interdependencies, single-cone analysis oppositely focus on interdependencies and ignores variable bound information. Hence, these heuristics may complement each other well.

The two most profound questions not answered in this thesis and therefore subject to further research is, perhaps, to what extend the reliability issues of ill-posedness occurs in real or realistic applications of mixed-integer second-order cone optimization problems, and how the theoretic establishments of this thesis behave in practice.

To elaborate on the first question, note that the property of being ill-posed is unstable as, by definition, even the tiniest perturbation of coefficients may alter the feasibility status. Hence, we would not expect a randomly generated problem to have this property but yet, several applications of semidefinite optimization have been identified where it occurs in practice [28, 56, 20, 9, 107, 101]. Likewise, preliminary results produced in collaboration with Alper Atamtürk on the series of relaxations solved by the branch-and-bound algorithm of MOSEK, have also begun to indicate that ill-posedness might occur within some of the mixed-integer second-order cone instances of CBLIB 2014. More research is needed, however, before any conclusions can be drawn. An interesting angle on this question is the possible causality dilemma caused the practitioners ability to use strict inequalities ($R_{++} = \{x \in \mathbb{R} : (x, y, 1) \in \mathcal{Q}_r^3,\ y \in \mathbb{R}\}$) and unbounded disjunctions [48] (usually formulated by what is known as bigM-type inequalities) if only ill-posedness could be handled reliably.

The second question can be divided into two subquestions regarding whether we can recognize, and whether we can use, approximative facial reduction certificates. Taking the extended embedding of Corollary 10 as example, a facial reduction certificate will most likely appear with $\tau \approx 0$ and $\kappa \approx 0$ in practice. Hence, there is the possibility of interpreting it as $\tau > 0$ and $\kappa = 0$ (complementary solutions of high norm) or $\tau = 0$ and $\kappa > 0$ (weak improving rays). At least, in this case, you have the alternatives to choose between. This is notably not the case in branch-and-bound methods augmented with functionality that may heuristically come across these hard to assess solutions of high norm and weak improving rays. To handle these cases, it would seem that acceptance thresholds of some sort are necessary. Aside from this, suppose now that the approximative facial reduction certificate of the extended embedding is correctly interpreted as $\tau = 0$ and $\kappa = 0$. Since interior point methods are characterized by maintaining $x \in \operatorname{int} \mathcal{K}$ and $s \in \operatorname{int} \mathcal{K}^*$ for proper cones throughout all iterations, it is not unreasonable of them to also return such an interior point pair for the approximative facial reduction certificate. If interpreted literally, this causes all cones to be facially reduced to the origin by Lemma 1, and since this is not always the correct behavior, mechanisms of some sort are also necessary to ensure that interior point pairs are projected to a smaller face if one is nearby.

On a final comment, these two questions and the brief examination given of them probably only scratched the surface of the issues remaining before ill-posedness can be coped with in practice. Hence, I hope other researchers will find inspiration in my work and carry on the research needed to close the gap between theoretic results and software implementations. Thank you for your attention!

6 Bibliography

[1] K. Abhishek, S. Leyffer, and J. Linderoth. FilMINT: An outer approximation-based solver for convex mixed-integer nonlinear programs. *INFORMS Journal on Computing*, 22(4):555–567, 2010.

[2] T. Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2008. URL `http://dx.doi.org/10.14279/depositonce-1634`.

[3] M. S. Aktürk, A. Atamtürk, and S. Gürel. A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Operations Research Letters*, 37(3):187–191, 2009.

[4] F. Alizadeh and G. Goldfarb. Second-order cone programming. *Mathematical programming*, 95(1):3–51, 2003.

[5] H. Anai. On solving semidefinite programming by quantifier elimination. In *Proceedings of American Control Conference, Philadelphia*, volume 5, pages 2814–2818, 1998.

[6] H. Anai and P. A. Parrilo. Convex quantifier elimination for semidefinite programming. In *Proceedings of the 6th International Workshop on Computer Algebra in Scientific Computing*, 2003.

[7] E. Andersen, C. Roos, and T. Terlaky. Notes on Duality in Second Order and p-Order Cone Optimization. *Optimization*, 51(4):627–643, 2002.

[8] E. D. Andersen, C. Roos, and T. Terlaky. On implementing a primal-dual interior-point method for conic quadratic optimization. *Mathematical Programming, Series B*, 95(2):249–277, 2003.

[9] M. F. Anjos and H. Wolkowicz. Strengthened semidefinite relaxations via a second lifting for the Max-Cut problem. *Discrete Applied Mathematics*, 119(1-2):79–106, 2002.

[10] A. Ben-Israel. Linear equations and inequalities on finite dimensional, real or complex, vector spaces: A unified theory. *Journal of Mathematical Analysis and Applications*, 27(2):367–389, 1969.

[11] A. Ben-tal and A. Nemirovski. On polyhedral approximations of the second-order cone. *Mathematics of Operations Research*, 26(2):193–205, 2001.

[12] A. Ben-tal and A. Nemirovski. *Lectures on modern convex optimization: Analysis, algorithms, and engineering applications*. Society for Industrial and Applied Mathematics, 2001. ISBN 978-0898714913.

[13] A. T. Benjamin. Sensible Rules for Remembering Duals - the SOB Method. *SIAM review*, 37(1):85–87, 1995.

[14] D. P. Bertsekas. *Convex optimization theory*. Athena Scientific, 2009. ISBN 978-1886529311.

[15] R. E. Bixby and E. Rothberg. Progress in computational mixed integer programming - A look back from the other side of the tipping point. *Annals of Operations Research*, 149(1):37–41, 2007.

[16] B. Borchers. SDPLIB 1.2, A Library of Semidefinite Programming Test Problems. *Optimization Methods and Software*, 11(1):683–690, 1999.

[17] J. Borwein and H. Wolkowicz. Regularizing the abstract convex program. *Journal of Mathematical Analysis and Applications*, 83(2):495–530, 1981.

[18] J. M. Borwein and A. S. Lewis. Partially finite convex programming, Part II: Explicit lattice models. *Mathematical Programming*, 57(1):49–83, 1992.

[19] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004. ISBN 978-0521833783.

[20] F. Burkowski, Y.-L. Cheung, and H. Wolkowicz. Efficient Use of Semidefinite Programming for Selection of Rotamers in Protein Conformations. *INFORMS Journal on Computing*, 26(4):748–766, 2014.

[21] Y.-L. Cheung. *Preprocessing and Reduction for Semidefinite Programming via Facial Reduction: Theory and Practice*. PhD thesis, University of Waterloo, 2013. URL `http://orion.uwaterloo.ca/~hwolkowi/henry/reports/thesisvrisnov13.pdf`.

[22] Y.-L. Cheung, S. Schurr, and H. Wolkowicz. Preprocessing and Regularization for Degenerate Semidefinite Programs. In *Computational and Analytical Mathematics*, volume 50 of *Springer Proceedings in Mathematics & Statistics*, pages 251–303. Springer New York, 2013.

[23] J. J. Cochran, L. A. Cox Jr., P. Keskinocak, J. P. Kharoufeh, and J. C. Smith. Symmetry handling in mixed-integer programming. In *Wiley Encyclopedia of Operations Research and Management Science*. 2011. ISBN 978-0470400630.

[24] G. B. Dantzig. Maximization of a linear function of variables subject to linear inequalities. In *Activity Analysis of Production and Allocation*, pages 339–347. John Wiley and Sons, New York, 1951.

[25] E. de Klerk, C. Roos, and T. Terlaky. Infeasible-start semidefinite programming algorithms via self-dual embeddings. In *Topics in semidefinite and interior point methods*, volume 18 of *Fields Institute Communications*, pages 215–236, 1998.

[26] A. G. Doig and A. H. Land. An Automatic Method of Solving Discrete Programming Problems. *Econometrica*, 28: 497–520, 1960.

[27] S. Drewes and S. Ulbrich. Subgradient based Outer Approximation for Mixed Integer Second Order Cone Programming. In *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 41–59. Springer New York, 2012. ISBN 978-1461419273.

[28] D. Drusvyatskiy, G. Pataki, and H. Wolkowicz. Coordinate shadows of semi-definite and euclidean distance matrices. *arXiv*, 2015. URL `http://arxiv.org/abs/1405.2037`.

[29] E. Erdougan and G. Iyengar. An active set method for single-cone second-order cone programs. *SIAM Journal on Optimization*, 17(2):459–484, 2006.

[30] F. Facchinei and J.-S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Springer Series in Operations Research and Financial Engineering. Springer-Verlag New York, 2003. ISBN 978-0387955803.

[31] Fair Isaac Corporation. Xpress-Optimizer Reference manual, Release 20.00. Technical report, 2009. `http://www.fico.com/en/wp-content/secure_upload/Xpress-Optimizer-Refernce-Manual.pdf`.

[32] J. Faraut and A. Korányi. *Analysis on symmetric cones*. Oxford University Press, 1994. ISBN 978-0198534778.

[33] M. Fischetti and D. Salvagnin. Pruning Moves. *INFORMS Journal on Computing*, 22(1):108–119, 2010.

[34] R. M. Freund and J. R. Vera. Some characterizations and properties of the "distance to ill-posedness" and the condition measure of a conic linear system. *Mathematical Programming*, 86(2):225–260, 1999.

[35] H. A. Friberg. Conic Benchmark Format - Version 1. Technical report, DTU Wind Energy, 2013. URL `http://orbit.dtu.dk/files/88492586/Conic_Benchmark_Format.pdf`.

[36] H. A. Friberg. CBLIB 2014: a benchmark library for conic mixed-integer and continuous optimization. *Mathematical Programming Computation*, 2015. URL `http://dx.doi.org/10.1007/s12532-015-0092-4`.

[37] H. A. Friberg. Erratum to: CBLIB 2014: a benchmark library for conic mixed-integer and continuous optimization. *Mathematical Programming Computation*, 2015. URL `http://dx.doi.org/10.1007/s12532-015-0098-y`.

[38] H. A. Friberg. A relaxed-certificate facial reduction algorithm based on subspace intersection. *Optimization Online*, 2016. URL `http://www.optimization-online.org/DB_HTML/2016/01/5283.html`.

[39] H. A. Friberg. Facial reduction heuristics and the motivational example of mixed-integer conic optimization. *Optimization Online*, 2016. URL `http://www.optimization-online.org/DB_HTML/2016/02/5324.html`.

[40] B. Gärtner and J. Matousek. Semidefinite Programming. In *Approximation Algorithms and Semidefinite Programming*, pages 15–25. Springer Berlin Heidelberg, 2012. ISBN 978-3642220159.

[41] D. M. Gay. Electronic mail distribution of linear programming test problems. *Mathematical Programming Society COAL Newsletter*, 13:10–12, 1985.

[42] L. E. Ghaoui. Convex Optimization. Lecture Notes for EE 227BT, EECS Department, UC Berkeley, 2013. URL `http://www.eecs.berkeley.edu/~elghaoui/Teaching/EE227BT/LectureNotes_EE227BT.pdf`.

[43] F. Glineur. Computational experiments with a linear approximation of second-order cone optimization. Image technical report 0001, Faculté Polytechnique de Mons, 2000.

[44] A. J. Goldman and A. W. Tucker. Theory of Linear Programming. In *Linear inequalities and related systems*, volume 38 of *Annals of Mathematical Studies*, pages 53–97. Princeton University Press, 1956.

[45] O. Günlük and J. Linderoth. Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Mathematical programming*, 124(1):183–206, 2010.

[46] Gurobi Optimization, Inc. Gurobi Optimizer Reference Manual - Version 5.6. Technical report, 2013. `http://www.gurobi.com/documentation/5.6/reference-manual/refman.pdf`.

[47] M. Halická, E. de Klerk, and C. Roos. On the Convergence of the Central Path in Semidefinite Optimization. *SIAM Journal on Optimization*, 12(4):1090–1099, 2002.

[48] H. Hijazi and L. Liberti. Constraint Qualification Failure in Second-Order Cone Formulations of Unbounded Disjunctions. Technical report, NICTA, Canberra ACT Australia, 2014. URL `http://www.optimization-online.org/DB_HTML/2014/07/4456.html`.

[49] F. S. Hillier and G. J. Lieberman. *Introduction to Operations Research*. McGraw-Hill, 2005. ISBN 0077298349.

[50] IBM Corporation. IBM ILOG CPLEX Optimization Studio V12.6.0 documentation. Technical report, 2014. `www-01.ibm.com/support/knowledgecenter/SSSA5P_12.6.0`.

[51] H. Iwane, H. Yanami, H. Anai, and K. Yokoyama. An effective implementation of symbolic–numeric cylindrical algebraic decomposition for quantifier elimination. *Theoretical Computer Science*, 479:43–69, 2013.

[52] R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computation*, The IBM Research Symposia Series, pages 85–103. Plenum Press, 1972. ISBN 978-3540682745.

[53] T. Koch, T. Achterberg, E. D. Andersen, O. Bastert, T. Berthold, R. E. Bixby, E. Danna, G. Gamrath, A. M. Gleixner, S. Heinz, A. Lodi, H. D. Mittelmann, T. Ralphs, D. Salvagnin, D. E. Steffy, and K. Wolter. MIPLIB 2010. *Mathematical Programming Computation*, 3(2):103–163, 2011.

[54] M. Koecher. Positivitätsbereiche im $\mathbb{R}^n$. *American Journal of Mathematics*, 79(3):575–596, 1957.

[55] K. Krishnan. Towards a simplex-like method for conic programming. Conference presentation, International Conference on Continuous Optimization (ICCOPT), 2004.

[56] N. Krislock and H. Wolkowicz. Explicit sensor network localization using semidefinite representations and clique reductions. *SIAM Journal on Optimization*, 20(5):2679–2708, 2010.

[57] F. Kılınç-Karzan and S. Yıldız. Two-Term Disjunctions on the Second-Order Cone. In *Integer Programming and Combinatorial Optimization*, volume 8494 of *Lecture Notes in Computer Science*, pages 345–356. Springer International Publishing, 2014. ISBN 978-3319075563.

[58] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

[59] B. F. Lourenço, M. Muramatsu, and T. Tsuchiya. Solving SDP Completely with an Interior Point Oracle. *Optimization Online*, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/06/4982.html`.

[60] M. Lubin, E. Yamangil, R. Bent, and J. P. Vielma. Extended Formulations in Mixed-integer Convex Programming. *ArXiv*, page 7, 2015. URL `http://arxiv.org/abs/1511.06710`.

[61] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Duality and self-duality for conic convex programming. Technical Report 9620/A, Econometric Institute, Erasmus University Rotterdam, 1996.

[62] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Duality results for conic convex programming. Technical Report 9719/A, Econometric Institute, Erasmus University Rotterdam, 1997.

[63] I. J. Lustig, R. E. Marsden, and D. F. Shanno. Interior point methods for linear programming: computational state of the art with discussion. *ORSA Journal on Computing*, 6:1–36, 1994.

[64] A. Mahajan and T. Munson. Exploiting second-order cone structure for global optimization. Technical Report ANL/MCS-P1801-1010, Argonne National Laboratory, 2010. URL `http://www.mcs.anl.gov/~tmunson/papers/socp.pdf`.

[65] H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.

[66] I. Maros. A generalized dual phase-2 simplex algorithm. *European Journal Of Operational Research*, 149(1):1–16, 2003.

[67] S. Mars and L. Schewe. An SDP-Package for SCIP. Technical Report August, TU Darmstadt, 2012. URL `http://www.opt.tu-darmstadt.de/~smars/scip_sdp.html`.

[68] MINLP WORLD. MINLPLIB2 library. URL `http://www.gamsworld.org/minlp/minlplib2/html/`.

[69] H. D. Mittelmann. MISOCP and large SOCP Benchmark, 2016. URL `http://plato.asu.edu/ftp/socp.html`.

[70] R. D. Monteiro and T. Tsuchiya. Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions. *Mathematical Programming*, 88(1):61–83, 2000.

[71] Y. Morenko, A. Vinel, Z. Yu, and P. Krokhmal. On p-norm linear discrimination. *European Journal of Operational Research*, 231(3):784–789, 2013.

[72] MOSEK ApS. The MOSEK C Optimizer API manual, Version 7.0. Technical report, 2013. `http://docs.mosek.com/7.0/capi.pdf`.

[73] M. Muramatsu. A pivoting procedure for a class of second-order cone programming. *Optimization Methods and Software*, 21(2):295–315, 2006.

[74] A. Nemirovski. Advances in Convex Optimization: Conic Programming. In *Proceedings of the International Congress of Mathematicians*, pages 413–444, 2006.

[75] Y. Nesterov, M. J. Todd, and Y. Ye. Infeasible-start interior-point primal-dual methods in nonlinear programming. CORE Discussion Papers 1995067, Université catholique de Louvain, Center for Operations Research and Econometrics, 1996.

[76] Y. E. Nesterov and M. J. Todd. Self-Scaled Barriers and Interior-Point Methods for Convex Programming. *Mathematics of Operations Research*, 22(1):1–42, 1997.

[77] D. P. O'Leary and G. W. Stewart. Computing the eigenvalues and eigenvectors of arrowhead matrices. *Journal of Computational Physics*, 90(2):497–505, 1990.

[78] G. Pataki. Cone-LP's and semidefinite programs: Geometry and a simplex-type method. In *Integer programming and combinatorial optimization*, volume 1084 of *Lecture Notes in Computer Science*, pages 162–174. 2005.

[79] G. Pataki. Strong duality in conic linear programming: facial reduction and extended duals. In *Computational and Analytical Mathematics*, volume 50 of *Springer Proceedings in Mathematics and Statistics*, pages 613–634. 2013. ISBN 978-1461476207.

[80] F. Permenter and P. A. Parrilo. Partial facial reduction: simplified, equivalent SDPs via approximations of the PSD cone. *arXiv*, 2014. URL `http://arxiv.org/abs/1408.4685`.

[81] F. Permenter, H. A. Friberg, and E. D. Andersen. Solving conic optimization problems via self-dual embedding and facial reduction: a unified approach. Technical report, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/09/5104.html`.

[82] I. Pólik. A (new) input format for conic optimization. Conference resentation, Applied mathematical programming and Modelling (APMOD), 2008.

[83] I. Pólik and T. Terlaky. Exact duality for optimization over symmetric cones. AdvOL Report 2007/10, Advanced Optimization Lab, McMaster University, 2007.

[84] F. A. Potra and R. Sheng. On homogeneous interior-point algorithms for semidefinite programming. *Optimization Methods and Software*, 9(1-3):161–184, 1998.

[85] J. Renegar. Incorporating Condition Measures into the Complexity Theory of Linear Programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995.

[86] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970. ISBN 978-0691015866.

[87] G. Sagnol. PICOS: A Python Interface for Conic Optimization Solvers, 2015. URL `http://picos.zib.de`.

[88] D. Salvagnin. *Constraint programming techniques for mixed integer linear programs*. PhD thesis, University of Padova, Italy, 2009. URL `http://paduaresearch.cab.unipd.it/1482/`.

[89] A. Skajaa, P. C. Hansen, and J. B. Jørgensen. *The Homogeneous Interior-Point Algorithm: Nonsymmetric Cones, Warmstarting, and Applications*. PhD thesis, Technical University of Denmark, 2013.

[90] M. Slater. Lagrange Multipliers Revisited: A Contribution to Nonlinear Programming. Cowles Foundation Discussion Papers 80, Cowles Foundation for Research in Economics, Yale University, 1950.

[91] V. A. Sposito and H. T. David. Saddle-Point Optimality Criteria Of Nonlinear Programming Problems Over Cones without Differentiability. *SIAM Journal on Applied Mathematics*, 20(4):698–702, 1971.

[92] J. F. Sturm. Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

[93] M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103(2):225–249, 2005.

[94] T. Terlaky and Z. Wang. On the Identification of the Optimal Partition of Second Order Cone Optimization Problems. *SIAM Journal on Optimization*, 24(1):385–414, 2014.

[95] R. C. Thompson. The Behavior of Eigenvalues and Singular Values under Perturbations of Restricted Rank. *Linear Algebra and its Applications*, 13(1):69–78, 1976.

[96] T. Tsuchiya. A convergence analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming. *Optimization Methods and Software*, 11(1):141–182, 1999.

[97] L. Tunçel and H. Wolkowicz. Strong duality and minimal representations for cone optimization. *Computational Optimization and Applications*, 53(2):619–648, 2012.

[98] J. P. Vielma, S. Ahmed, and G. L. Nemhauser. A Lifted Linear Programming Branch-and-Bound Algorithm for Mixed Integer Conic Quadratic Programs. *INFORMS Journal on Computing*, 20:438–450, 2008.

[99] J. P. Vielma, I. Dunning, J. Huchette, and M. Lubin. Extended Formulations in Mixed Integer Conic Quadratic Programming. *ArXiv*, 2015. URL `http://arxiv.org/abs/1505.07857`.

[100] H. Waki and M. Muramatsu. Facial reduction algorithms for conic optimization problems. *Journal of Optimization Theory and Applications*, 158(1):188–215, 2013.

[101] H. Wolkowicz and Q. Zhao. Semidefinite programming relaxations for the graph partitioning problem. *Discrete Applied Mathematics*, 96-97:461–479, 1999.

[102] H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors. *Handbook of semidefinite programming: theory, algorithms, and applications*, volume 27 of *International Series in Operations Research & Management Science*. Springer US, 2000. ISBN 978-0792377719.

[103] 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.

[104] M. Yamashita, K. Fujisawa, K. Nakata, M. Nakata, M. Fukuda, K. Kobayashi, and K. Goto. A high-performance software package for semidefinite programs: SDPA 7. Research Report B-460, Dept. of Mathematical and Computing Science, Tokyo Institute of Technology, 2010. URL `http://www.optimization-online.org/DB_HTML/2010/01/2531.html`.

[105] Y. Ye, M. J. Todd, and S. Mizuno. An $\mathcal{O}(\sqrt{n}L)$-Iteration Homogeneous and Self-Dual Linear Programming Algorithm. *Mathematics of Operations Research*, 19(1):53–67, 1994.

[106] E. Zawadzki and T. Sandholm. Search Tree Restructuring. Technical Report CMU-CS-10-102, Carnegie Mellon University, 2010. URL `http://www.cs.cmu.edu/~epz/Papers/Search_Tree_Restructuring_Tech_epz.pdf`.

[107] Q. Zhao, S. E. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite Programming Relaxations for the Quadratic Assignment Problem. *Journal of Combinatorial Optimization*, 2(1):71–109, 1998.

[108] H. Ziegler. Solving certain singly constrained convex optimization problems in production planning. *Operations Research Letters*, 1(6):246–252, 1982.

Appendix A

[Friberg 35]
The Conic Benchmark Format: Version 1 - Technical Reference Manual

The Conic Benchmark Format

Version 1 – Technical Reference Manual

Department of
Wind Energy
E Report 2014

Henrik A. Friberg
PhD Candidate at the Technical University of Denmark,
Department of Wind Energy, and employee at MOSEK ApS.
E-mail: haf@mosek.com.

DTU Wind Energy – Report E-0047

ISBN 978-87-92896-84-1

February 2014

DTU Wind Energy
Department of Wind Energy

Abstract

This document constitutes the technical reference manual of the Conic Benchmark Format with file extension: `.cbf` or `.CBF`. It unifies linear, second-order cone (also known as conic quadratic) and semidefinite optimization with mixed-integer variables. The format has been designed with benchmark libraries in mind, and therefore focuses on compact and easily parsable representations. The problem structure is separated from the problem data, and the format moreover facilitate benchmarking of hotstart capability through sequences of changes.

Acknowledgements

Henrik A. Friberg is the creator of the CBF format and the author of this document. His work would never have been accomplished to the same degree of quality without the in-depth review and helpful comments from Erling D. Andersen and Mathias Stolpe. The ideas for a conic input format presented by Imre Pólik has been a great source of inspiration.

Funding

Henrik A. Friberg was funded by MOSEK ApS, and the Danish Ministry of Science, Innovation and Higher Education through the Industrial PhD project "Combinatorial Optimization over Second-Order Cones and Industrial Applications".

Contents

1 Minimal working example

The conic optimization problem (1), has three variables in a quadratic cone Q^3 - first one is integer - and an affine expression in domain $\{0\}$ (equality constraint).

$$\begin{array}{ll} \text{minimize} & 5.1\,x_0 \\ \text{subject to} & 6.2\,x_1 + 7.3\,x_2 - 8.4 \in \{0\} \\ & x \in Q^3,\ x_0 \in \mathbb{Z}. \end{array} \tag{1}$$

Its formulation in the Conic Benchmark Format begins with the version of the CBF format used, to safeguard against later revisions.

```
VER
1
```

Next follows the problem structure, consisting of the objective sense, the number and domain of variables, the indices of integer variables, and the number and domain of scalar-valued affine expressions (i.e., the equality constraint).

```
OBJSENSE
MIN

VAR
3 1
Q 3

INT
1
0

CON
1 1
L= 1
```

Finally follows the problem data, consisting of the coefficients of the objective, the coefficients of the constraints, and the constant terms of the constraints. All data is specified on a sparse coordinate form.

```
OBJACOORD
1
0 5.1

ACOORD
2
0 1 6.2
0 2 7.3

BCOORD
1
0 -8.4
```

This concludes the example! Please see Section 2 and Section 3 for details about the document structure and use of keywords.

2 The structure of CBF files

This section defines how information is written in the CBF format, without being specific about the type of information being communicated.

2.1 Information items

The format is composed as a list of information items. The first line of an information item is the `KEYWORD`, revealing the type of information provided. The second line - of some keywords only - is the `HEADER`, typically revealing the size of information that follows. The remaining lines are the `BODY` holding the actual information to be specified.

```
KEYWORD
BODY

KEYWORD
HEADER
BODY
```

The `KEYWORD` specifies how of each line in the `HEADER` and `BODY` is structured. Moreover, the number of lines in the `BODY` is decidable either from the `KEYWORD`, the `HEADER`, or from another information item required to precede it.

2.2 Information groups and their ordering

All information items belong to exactly one of the three groups of information. These information groups, and the order they must appear in, is:

1. File format.
2. Problem structure.
3. Problem data.

The first group, file format, provides information on how to interpret the file. It is currently limited to the keyword `VER`, specifying the version of the CBF format in use. The second group, problem structure, provides the information needed to deduce the type and size of the problem instance. Finally, the third group, problem data, specifies the coefficients and constants of the problem instance.

2.3 Embedded hotstart-sequences

A sequence of problem instances, based on the same problem structure, is allowed within a single file. This is facilitated via the `CHANGE` keyword used within the problem data information group, as a separator between the information items of each instance. The information items following a `CHANGE` keyword is appending to, or changing (e.g., setting coefficients back to their default value of zero), the problem data of the preceding instance.

The sequence is intended for benchmarking of hotstart capability, where the solvers can reuse their internal state and solution (subject to the achieved accuracy) as warmpoint for the succeeding instance. Whenever this feature is unsupported or undesired, the keyword `CHANGE` should be interpreted as the end of file.

2.4 File encoding and line width restrictions

The format is based on the US-ASCII printable character set with two extensions as listed below. Note, by definition, that none of these extensions can be misinterpreted as printable US-ASCII characters:

- A line feed marks the end of a line, carriage returns are ignored.
- Comment-lines may contain unicode characters in UTF-8 encoding.

The line width is restricted to 512 bytes, with 3 bytes reserved for the potential carriage return, line feed and null-terminator.

Integers and floating point numbers must follow the ISO C decimal string representation in the standard "C" locale. The format does not impose restrictions on the magnitude of, or number of significant digits in, numeric data, but the use of 64-bit integers and 64-bit IEEE 754 floating point numbers should be sufficient to avoid loss of precision.

2.5 Comment-line and whitespace rules

The format allows single-line comments respecting the following rule:

- Lines having first byte equal to '`#`' (US-ASCII 35) are comments, and should be ignored. Comments are only allowed between information items.

Given that a line is not a comment-line, whitespace characters should be handled according to the following rules:

- Leading and trailing whitespace characters should be ignored.
- The seperator between multiple pieces of information on one line, is either one or more whitespace characters.
- Lines containing only whitespace characters are empty, and should be ignored. Empty lines are only allowed between information items.

3 How instances are specified

This section defines the spectrum of conic optimization problems that can be formulated in terms of the keywords of the CBF format.

3.1 Problem structure

Conic optimization problems consist of variables, constraints and one objective function. In the CBF format, these are defined as follows.

- Variables are either scalar-valued (part of a vector restricted to a cone), or matrix-valued (restricted to be symmetric positive semidefinite). These are refered to as the scalar variables, x_j for $j \in \mathcal{J}$, and PSD variables, X_j for $j \in \mathcal{J}^{PSD}$. Only scalar variables can be integer.
- Constraints are affine expressions of the variables, either scalar-valued (part of a vector restricted to a cone), or matrix-valued (restricted to be symmetric positive semidefinite). These are thus refered to as the scalar constraints, with affine expressions g_i for $i \in \mathcal{I}$, and PSD constraints, with affine expressions G_i for $i \in \mathcal{I}^{PSD}$.
- The objective is a scalar-valued affine expression of the variables, either to be minimized or maximized. We refer to this expression as g^{obj}.

The problem structure defines the objective sense, whether it is minimization and maximization, using the keyword `OBJSENSE` (follow the hyperlink or see Appendix B). It also defines the index sets, $\mathcal{J}$, $\mathcal{J}^{PSD}$, $\mathcal{I}$ and $\mathcal{I}^{PSD}$, which are all numbered from zero, $\{0, 1, \ldots\}$, and empty until explicitly constructed.

- Scalar variables are constructed in vectors restricted to a conic domain, such as $(x_0, x_1) \in \mathbb{R}^2_+$, $(x_2, x_3, x_4) \in Q^3$, etc. In terms of the Cartesian product, this generalizes to $x \in K_1^{n_1} \times K_2^{n_2} \times \cdots \times K_k^{n_k}$, which in the CBF format becomes

  `VAR`
  $n \;\; k$
  $K_1 \;\; n_1$
  $K_2 \;\; n_2$
  $\ldots$
  $K_k \;\; n_k$

  where n is the total number of scalar variables. The list of supported cones is found in Appendix A. Integrality of scalar variables can be specified afterwards, as in the minimal working example of Section 1, using the keyword `INT` (follow the hyperlink or see Appendix B).
- PSD variables are constructed one-by-one. That is, $X_j \succeq \mathbf{0}^{n_j \times n_j}$ for $j \in \mathcal{J}^{PSD}$, construct matrix-valued variables of size $n_j \times n_j$ restricted

to be symmetric positive semidefinite. In the CBF format, this list of constructions becomes

```
PSDVAR
```
N

n_1

n_2

$\dots$

n_N

where N is the total number of PSD variables.

- Scalar constraints are constructed in vectors restricted to a conic domain, such as $(g_0, g_1) \in \mathbb{R}^2_+$, $(g_2, g_3, g_4) \in Q^3$, etc. In terms of the Cartesian product, this generalizes to $g \in K_1^{m_1} \times K_2^{m_2} \times \cdots \times K_k^{m_k}$, which in the CBF format becomes

  ```
  CON
  ```
  $m\ \ k$

  $K_1\ \ m_1$

  $K_2\ \ m_2$

  $\dots$

  $K_k\ \ m_k$

  where m is the total number of scalar constraints. The list of supported cones is found in Appendix A.

- PSD constraints are constructed one-by-one. That is, $G_i \succeq \mathbf{0}^{m_i \times m_i}$ for $i \in \mathcal{I}^{PSD}$, construct matrix-valued affine expressions of size $m_i \times m_i$ restricted to be symmetric positive semidefinite. In the CBF format, this list of constructions becomes

  ```
  PSDCON
  ```
  M

  m_1

  m_2

  $\dots$

  m_M

  where M is the total number of PSD constraints.

With the objective sense, variables (with integer indications) and constraints, the definitions of the many affine expressions follow in problem data.

Keywords covered in this section:

`OBJSENSE`	- *Define the objective sense.*
`VAR`	- *Construct the scalar variables.*
`INT`	- *Put integer requirements on a selected subset of scalar variables.*
`PSDVAR`	- *Construct the PSD variables.*
`CON`	- *Construct the scalar constraints.*
`PSDCON`	- *Construct the PSD constraints.*

3.2 Problem data

The problem data defines the coefficients and constants of the affine expressions of the problem instance. These are considered zero until explicitly defined, implying that instances with no keywords from this information group are, in fact, valid. Duplicated or conflicting information is a failure to comply with the standard. Consequently, two coefficients written to the same position in a matrix (or to transposed positions in a symmetric matrix) is an error.

The affine expressions of the objective, g^{obj}, of the scalar constraints, g_i, and of the PSD constraints, G_i, are defined separately. The following notation uses the standard trace inner product for matrices, $\langle X, Y \rangle = \sum_{i,j} X_{ij} Y_{ij}$.

- The affine expression of the objective is defined as

$$g^{obj} = \sum_{j \in \mathcal{J}^{PSD}} \langle F_j^{obj}, X_j \rangle + \sum_{j \in \mathcal{J}} a_j^{obj} x_j + b^{obj},$$

  in terms of the symmetric matrices, F_j^{obj}, and scalars, a_j^{obj} and b^{obj}. These are specified on a sparse coordinate form using keywords `OBJFCOORD`, `OBJACOORD`, `OBJBCOORD` (follow the hyperlinks or see Appendix B).

- The affine expressions of the scalar constraints are defined, for $i \in \mathcal{I}$, as

$$g_i = \sum_{j \in \mathcal{J}^{PSD}} \langle F_{ij}, X_j \rangle + \sum_{j \in \mathcal{J}} a_{ij} x_j + b_i,$$

  in terms of the symmetric matrices, F_{ij}, and scalars, a_{ij} and b_i. These are specified on a sparse coordinate form using keywords `FCOORD`, `ACOORD`, and `BCOORD` (follow the hyperlinks or see Appendix B).

- The affine expressions of the PSD constraints are defined, for $i \in \mathcal{I}^{PSD}$, as

$$G_i = \sum_{j \in \mathcal{J}} x_j H_{ij} + D_i,$$

  in terms of the symmetric matrices, H_{ij} and D_i. These are specified on a sparse coordinate form using keywords `HCOORD` and `DCOORD` (follow the hyperlinks or see Appendix B).

Keywords covered in this section:

`OBJFCOORD`
`OBJACOORD`
`OBJBCOORD`
	- *Define the affine expression of the objective.*

`FCOORD`
`ACOORD`
`BCOORD`
	- *Define the affine expressions of the scalar constraints.*

`HCOORD` `DCOORD`	- *Define the affine expressions of the PSD constraints.*

A List of cones

The format uses an explicit syntax for symmetric positive semidefinite cones as shown in Section 3.1. For scalar variables and constraints, constructed in vectors, the supported conic domains and their minimum sizes are given as follows.

- **Free domain**
  CBF name `F`: A cone in the linear family defined by
  $$\{x \in \mathbb{R}^n\}, \text{ for } n \geq 1.$$
- **Positive orthant**
  CBF name `L+`: A cone in the linear family defined by
  $$\{x \in \mathbb{R}^n \mid x_j \geq 0 \text{ for } j = 1, \ldots, n\}, \text{ for } n \geq 1.$$
- **Negative orthant**
  CBF name `L-`: A cone in the linear family defined by
  $$\{x \in \mathbb{R}^n \mid x_j \leq 0 \text{ for } j = 1, \ldots, n\}, \text{ for } n \geq 1.$$
- **Fixpoint zero**
  CBF name `L=`: A cone in the linear family defined by
  $$\{x \in \mathbb{R}^n \mid x_j = 0 \text{ for } j = 1, \ldots, n\}, \text{ for } n \geq 1.$$
- **Quadratic cone**
  CBF name `Q`: A cone in the second-order cone family defined by
  $$\left\{ \begin{pmatrix} p \\ x \end{pmatrix} \in \mathbb{R} \times \mathbb{R}^{n-1},\ p^2 \geq x^T x,\ p \geq 0 \right\}, \text{ for } n \geq 2.$$
- **Rotated quadratic cone**
  CBF name `QR`: A cone in the second-order cone family defined by
  $$\left\{ \begin{pmatrix} p \\ q \\ x \end{pmatrix} \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{n-2},\ 2pq \geq x^T x,\ p \geq 0,\ q \geq 0 \right\}, \text{ for } n \geq 2.$$

B List of keywords

All keywords are case sensitive and may not appear more than once in any instance specification. In summary, by information group, they are given as:

- **File format**

  `VER`

- **Problem structure**

  `OBJSENSE`, `PSDVAR`, `VAR`, `INT`, `PSDCON`, `CON`

- **Problem data**

  `OBJFCOORD`, `OBJACOORD`, `OBJBCOORD`
  `FCOORD`, `ACOORD`, `BCOORD`
  `HCOORD`, `DCOORD`
  `CHANGE`

The information groups must be ordered as specified in Section 2.2. All keywords, and their ordering within their information group, are optional unless explicitly stated as a remark to the respective keyword.

VER

The version of the Conic Benchmark Format used to write the file.

HEADER	None.
BODY	One line formatted as: `INT` This is the version number.

Remarks:
Must appear exactly once in a file, as the first keyword.

OBJSENSE

Define the objective sense.

HEADER	None.
BODY	One line formatted as: `STR` Having `MIN` indicates minimize, and `MAX` indicates maximize. Capital letters are required.

Remarks:
Must appear exactly once in a file.

PSDVAR

Construct the PSD variables.

HEADER — One line formatted as:
`INT`
This is the number of PSD variables in the problem.

BODY — A list of lines formatted as
`INT`
This indicates the number of rows (equal to the number of columns) in the matrix-valued PSD variable. The number of lines should match the number stated in the header.

VAR

Construct the scalar variables.

HEADER — One line formatted as:
`INT INT`
This is the number of scalar variables, followed by the number of conic domains they are restricted to.

BODY — A list of lines formatted as
`STR INT`
This indicates the cone name (see Appendix A), and the number of scalar variables restricted to this cone. These numbers should accumulate to the number of scalar variables stated first in the header. The number of lines should match the second number stated in the header.

INT

Put integer requirements on a selected subset of scalar variables.

HEADER — One line formatted as:
`INT`
This is the number of integer scalar variables in the problem.

BODY — A list of lines formatted as
`INT`
This indicates the scalar variable index $j \in \mathcal{J}$. The number of lines should match the number stated in the header.

Remarks:
Can only be used after these keywords: `VAR`

PSDCON

Construct the PSD constraints.

HEADER One line formatted as:

`INT`

This is the number of PSD constraints in the problem.

BODY A list of lines formatted as

`INT`

This indicates the number of rows (equal to the number of columns) in the matrix-valued affine expression of the PSD constraint. The number of lines should match the number stated in the header.

Remarks:

Can only be used after these keywords: `PSDVAR`, `VAR`

CON

Construct the scalar constraints.

HEADER One line formatted as:

`INT INT`

This is the number of scalar constraints, followed by the number of conic domains they restrict to.

BODY A list of lines formatted as

`STR INT`

This indicates the cone name (see Appendix A), and the number of affine expressions restricted to this cone. These numbers should accumulate to the number of scalar constraints stated first in the header. The number of lines should match the second number stated in the header.

Remarks:

Can only be used after these keywords: `PSDVAR`, `VAR`

OBJFCOORD

Input sparse coordinates (quadruplets) to define the symmetric matrices, F_j^{obj}, as used in the objective.

HEADER One line formatted as:

`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as

`INT INT INT REAL`

This indicates the PSD variable index $j \in \mathcal{J}^{PSD}$, the row index, the column index and the coefficient value. The number of lines should match the number stated in the header.

OBJACOORD

Input sparse coordinates (pairs) to define the scalars, a_j^{obj}, as used in the objective.

HEADER One line formatted as:
`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as
`INT REAL`

This indicates the scalar variable index $j \in \mathcal{J}$ and the coefficient value. The number of lines should match the number stated in the header.

OBJBCOORD

Input the scalar, b^{obj}, as used in the objective.

HEADER None.

BODY One line formatted as:
`REAL`

This indicates the coefficient value.

FCOORD

Input sparse coordinates (quintuplets) to define the symmetric matrices, F_{ij}, as used in the scalar constraints.

HEADER One line formatted as:
`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as
`INT INT INT INT REAL`

This indicates the scalar constraint index $i \in \mathcal{I}$, the PSD variable index $j \in \mathcal{J}^{PSD}$, the row index, the column index and the coefficient value. The number of lines should match the number stated in the header.

ACOORD

Input sparse coordinates (triplets) to define the scalars, a_{ij}, as used in the scalar constraints.

HEADER One line formatted as:
`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as
`INT INT REAL`

This indicates the scalar constraint index $i \in \mathcal{I}$, the scalar variable index $j \in \mathcal{J}$ and the coefficient value. The number of lines should match the number stated in the header.

BCOORD

Input sparse coordinates (pairs) to define the scalars, b_i, as used in the scalar constraints.

HEADER One line formatted as:

`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as

`INT REAL`

This indicates the scalar constraint index $i \in \mathcal{I}$ and the coefficient value. The number of lines should match the number stated in the header.

HCOORD

Input sparse coordinates (quintuplets) to define the symmetric matrices, H_{ij}, as used in the PSD constraints.

HEADER One line formatted as:

`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as

`INT INT INT INT REAL`

This indicates the PSD constraint index $i \in \mathcal{I}^{PSD}$, the scalar variable index $j \in \mathcal{J}$, the row index, the column index and the coefficient value. The number of lines should match the number stated in the header.

DCOORD

Input sparse coordinates (quadruplets) to define the symmetric matrices, D_i, as used in the PSD constraints.

HEADER One line formatted as:

`INT`

This is the number of coordinates to be specified.

BODY A list of lines formatted as

`INT INT INT REAL`

This indicates the PSD constraint index $i \in \mathcal{I}^{PSD}$, the row index, the column index and the coefficient value. The number of lines should match the number stated in the header.

CHANGE

Start of a new instance specification based on changes to the previous.

HEADER None.

BODY None.

Remarks:
Can be interpreted as the end of file when the hotstart-sequence is unsupported or undesired.

C Examples

C.1 Mixing linear, second-order and semidefinite cones

The conic optimization problem (2), has a semidefinite cone, a quadratic cone over unordered subindices, and two equality constraints.

$$
\begin{array}{lll}
\text{minimize} & \left\langle \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}, X_1 \right\rangle + x_1 & \\
\text{subject to} & \left\langle \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, X_1 \right\rangle + x_1 & = \quad 1.0\,, \\
 & \left\langle \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, X_1 \right\rangle + x_0 + x_2 & = \quad 0.5\,, \\
 & x_1 \geq \sqrt{x_0^2 + x_2^2}\,, & \\
 & X_1 \succeq \mathbf{0}\,. &
\end{array} \tag{2}
$$

The equality constraints are easily rewritten to the conic form, $(g_0, g_1) \in \{0\}^2$, by moving constants such that the right-hand-side becomes zero. The quadratic cone does not fit under the `VAR` keyword in this variable permutation, however, as opposed to the minimal working example (1). Instead, it takes a scalar constraint $(g_2, g_3, g_4) = (x_1, x_0, x_2) \in Q^3$, with scalar variables constructed as $(x_0, x_1, x_2) \in \mathbb{R}^3$. Its formulation in the CBF format is written in verbatim.

```
# File written using this version of the Conic Benchmark Format:
#     | Version 1.
VER
1

# The sense of the objective is:
#     | Minimize.
OBJSENSE
MIN

# One PSD variable of this size:
#     | Three times three.
PSDVAR
1
3

# Three scalar variables in this one conic domain:
#     | Three are free.
VAR
3 1
F 3

# Five scalar constraints with affine expressions in two conic domains:
#     | Two are fixed to zero.
```

```
#     | Three are in conic quadratic domain.
CON
5 2
L= 2
Q 3

# Five coordinates in F^{obj}_j coefficients:
#     | F^{obj}[0][0,0] = 2.0
#     | F^{obj}[0][1,0] = 1.0
#     | and more...
OBJFCOORD
5
0 0 0 2.0
0 1 0 1.0
0 1 1 2.0
0 2 1 1.0
0 2 2 2.0

# One coordinate in a^{obj}_j coefficients:
#     | a^{obj}[1] = 1.0
OBJACOORD
1
1 1.0

# Nine coordinates in F_ij coefficients:
#     | F[0,0][0,0] = 1.0
#     | F[0,0][1,1] = 1.0
#     | and more...
FCOORD
9
0 0 0 0 1.0
0 0 1 1 1.0
0 0 2 2 1.0
1 0 0 0 1.0
1 0 1 0 1.0
1 0 2 0 1.0
1 0 1 1 1.0
1 0 2 1 1.0
1 0 2 2 1.0

# Six coordinates in a_ij coefficients:
#     | a[0,1] = 1.0
#     | a[1,0] = 1.0
#     | and more...
ACOORD
6
0 1 1.0
1 0 1.0
1 2 1.0
2 1 1.0
3 0 1.0
4 2 1.0

# Two coordinates in b_i coefficients:
#     | b[0] = -1.0
#     | b[1] = -0.5
BCOORD
```

```
2
0 -1.0
1 -0.5
```

C.2 Mixing semidefinite variables and linear matrix inequalities

The standard forms in semidefinite optimization are usually based either on semidefinite variables or linear matrix inequalities. In the CBF format, both forms are supported and can even be mixed as shown in (3).

$$
\begin{array}{lll}
\text{minimize} & \left\langle \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, X_1 \right\rangle + x_1 + x_2 + 1 & \\
\text{subject to} & \left\langle \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, X_1 \right\rangle - x_1 - x_2 & \geq \ 0.0\,, \\
& x_1 \begin{bmatrix} 0 & 1 \\ 1 & 3 \end{bmatrix} + x_2 \begin{bmatrix} 3 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & \succeq \ \mathbf{0}\,, \\
& X_1 \succeq \mathbf{0}\,. &
\end{array}
\tag{3}
$$

Its formulation in the CBF format is written in verbatim.

```
# File written using this version of the Conic Benchmark Format:
#     | Version 1.
VER
1

# The sense of the objective is:
#     | Minimize.
OBJSENSE
MIN

# One PSD variable of this size:
#     | Two times two.
PSDVAR
1
2

# Two scalar variables in this one conic domain:
#     | Two are free.
VAR
2 1
F 2

# One PSD constraint of this size:
#     | Two times two.
PSDCON
1
2
```

```
# One scalar constraint with an affine expression in this one conic domain:
#     | One is greater than or equal to zero.
CON
1 1
L+ 1

# Two coordinates in F^{obj}_j coefficients:
#     | F^{obj}[0][0,0] = 1.0
#     | F^{obj}[0][1,1] = 1.0
OBJFCOORD
2
0 0 0 1.0
0 1 1 1.0

# Two coordinates in a^{obj}_j coefficients:
#     | a^{obj}[0] = 1.0
#     | a^{obj}[1] = 1.0
OBJACOORD
2
0 1.0
1 1.0

# One coordinate in b^{obj} coefficient:
#     | b^{obj} = 1.0
OBJBCOORD
1.0

# One coordinate in F_ij coefficients:
#     | F[0,0][1,0] = 1.0
FCOORD
1
0 0 1 0 1.0

# Two coordinates in a_ij coefficients:
#     | a[0,0] = -1.0
#     | a[0,1] = -1.0
ACOORD
2
0 0 -1.0
0 1 -1.0

# Four coordinates in H_ij coefficients:
#     | H[0,0][1,0] = 1.0
#     | H[0,0][1,1] = 3.0
#     | and more...
HCOORD
4
0 0 1 0 1.0
0 0 1 1 3.0
0 1 0 0 3.0
0 1 1 0 1.0

# Two coordinates in D_i coefficients:
#     | D[0][0,0] = -1.0
#     | D[0][1,1] = -1.0
DCOORD
```

```
2
0 0 0 -1.0
0 1 1 -1.0
```

C.3 Optimization over a sequence of objectives

The linear optimization problem (4), is defined for a sequence of objectives such that hotstarting from one to the next might be advantages.

$$
\begin{array}{lrcl}
\text{maximize}_k & g_k^{obj} & & \\
\text{subject to} & 50\,x_0 + 31 & \leq & 250\,, \\
 & 3\,x_0 - 2x_1 & \geq & -4\,, \\
 & x \in \mathbb{R}_+^2, & &
\end{array} \tag{4}
$$

given,

$g_0^{obj} = x_0 + 0.64x_1.$

$g_1^{obj} = 1.11x_0 + 0.76x_1.$

$g_2^{obj} = 1.11x_0 + 0.85x_1.$

Its formulation in the CBF format is written in verbatim.

```
# File written using this version of the Conic Benchmark Format:
#     | Version 1.
VER
1

# The sense of the objective is:
#     | Maximize.
OBJSENSE
MAX

# Two scalar variables in this one conic domain:
#     | Two are nonnegative.
VAR
2 1
L+ 2

# Two scalar constraints with affine expressions in these two conic domains:
#     | One is in the nonpositive domain.
#     | One is in the nonnegative domain.
CON
2 2
L- 1
L+ 1

# Two coordinates in a^{obj}_j coefficients:
#     | a^{obj}[0] = 1.0
```

```
#     | a^{obj}[1] = 0.64
OBJACOORD
2
0 1.0
1 0.64

# Four coordinates in a_ij coefficients:
#     | a[0,0] = 50.0
#     | a[1,0] = 3.0
#     | and more...
ACOORD
4
0 0 50.0
1 0 3.0
0 1 31.0
1 1 -2.0

# Two coordinates in b_i coefficients:
#     | b[0] = -250.0
#     | b[1] = 4.0
BCOORD
2
0 -250.0
1 4.0

# New problem instance defined in terms of changes.
CHANGE

# Two coordinate changes in a^{obj}_j coefficients. Now it is:
#     | a^{obj}[0] = 1.11
#     | a^{obj}[1] = 0.76
OBJACOORD
2
0 1.11
1 0.76

# New problem instance defined in terms of changes.
CHANGE

# One coordinate change in a^{obj}_j coefficients. Now it is:
#     | a^{obj}[0] = 1.11
#     | a^{obj}[1] = 0.85
OBJACOORD
1
1 0.85
```

Appendix B

[Friberg 36]
CBLIB 2014: A benchmark library for conic mixed-integer and continuous optimization

CBLIB 2014: A benchmark library for conic mixed-integer and continuous optimization

Henrik A. Friberg*

February 4, 2016

Abstract

The Conic Benchmark Library is an ongoing community-driven project aiming to challenge commercial and open source solvers on mainstream cone support. In this paper, 121 mixed-integer and continuous second-order cone problem instances have been selected from 11 categories as representative for the instances available online. As current file formats were found incapable, we embrace the new Conic Benchmark Format as standard for conic optimization. Tools are provided to aid integration of this format with other software packages.

1 Introduction

A conic optimization problem is the problem of minimizing (or maximizing) a linear objective over a feasible region specified in terms of affine expressions, convex cones, and, if any, integer constraints. It may be formulated as

$$\begin{array}{lrcll} \underset{x}{\text{minimize}} & c^T x & & & \\ \text{subject to} & A_i x - b_i & \in & \mathcal{K}_i, & \text{for } i = 1, \ldots, k, \\ & x_j & \in & \mathbb{Z}, & \text{for } j \in \mathcal{I}. \end{array} \tag{1}$$

The conic form (1) allows us to express all convex mixed-integer and continuous optimization problems without loss of generality [21], but this generality offers no advantages from a computational point of view. Instead, only three types of cones (nonnegative orthant, quadratic cone and semidefinite cone) are typically used to solve a broad range of applications [4]. These three cone types are called the real-valued symmetric cones, and are usually accompanied by equality constraints for convenience, which in (1) would be the cone of zeros $\{0\}^n$. As example, we compare the traditional and the conic form of the classical Markowitz portfolio optimization problem [34]. The portfolio problem (2), maximizes expected return subject to the accepted risk γ and investable wealth

*PhD candidate at the Technical University of Denmark, Department of Wind Energy, and employee at MOSEK ApS. E-mail: haf@mosek.com.

ω. The return vector μ and covariance matrix $\Sigma = U^T U$, characterize the investments in consideration.

$$\begin{array}{llcll} \underset{x}{\text{maximize}} & \mu^T x & & & \\ \text{subject to} & x^T \Sigma x & \leq & \gamma & \leftrightarrow \\ & e^T x & = & \omega & \leftrightarrow \\ & x & \geq & 0 & \leftrightarrow \end{array} \quad \begin{array}{llcl} \underset{x}{\text{maximize}} & \mu^T x & & \\ \text{subject to} & (\gamma^{1/2}, Ux) & \in & \mathcal{Q}^{1+n}, \\ & e^T x - \omega & \in & \{0\}, \\ & x & \in & \mathbb{R}^n_+. \end{array} \tag{2}$$

The traditional form on the left use a convex functional, $f(x) \leq 0$, to express the nonlinearity of the risk constraint. The conic form on the right achieves the same using the quadratic cone, $\mathcal{Q}^{1+n} = \{(r, x) \in \mathbb{R}^1_+ \times \mathbb{R}^n \mid r^2 \geq x^T x\}$. The equation and variable nonnegativity of the traditional form are formulated using two linear cones, the set of zero and the nonnegative orthant. More advanced examples of conic reformulations are found in [1] and [4].

One advantage of the conic form is that convexity does not have to be investigated, since it follows from convexity of the cones involved. In contrast, the convexity of a nonlinear problem in the traditional form cannot be established based on structural information, but has to be verified using the input data, such as Σ in (2). Another advantage stems from the efficiency by which primal-dual interior-point methods are able to exploit the underlying structure of symmetric cones [38]. This advantage is reflected in the state-of-the-art optimization software, with high-performing implementations in all major commercial solvers; XPRESS [18], MOSEK [37], GUROBI [25] and CPLEX [27]. The open source projects listed in [43] are furthermore mostly based on variants of the method proposed in [38], including the significant contributions of SEDUMI [49] and SDPT3 [51]. SEDUMI, SDPT3 and MOSEK support all real-valued symmetric cones, while XPRESS, GUROBI and CPLEX omit support of the semidefinite cone. Integer constraints can be handled by all listed commercial solvers, but not by any of the open source projects. Open source support for conic mixed-integer optimization, however, is actively being added to the constraint integer programming framework SCIP through cone solver plugins [35] and outer approximations [6].

What is essentially missing from this development is a proper and publicly available benchmark library. Benchmark libraries are known to have a great effect on stimulating improvements in reliability and performance in optimization software. The *NETLIB LP* [20] library, for instance, was the first electronically distributed benchmark library for continuous linear optimization and often attributed for its major effect on the development of LP solvers. Correspondingly, *MIPLIB* [32] has played a major role in the field of mixed-integer linear optimization. In review of benchmark libraries for conic optimization, *SDPLIB* [8] and the library of structured semidefinite programming instances [15] are worth noticing although their focus is limited to the semidefinite cone. A mixture of different cone types were considered in the *7th DIMACS Implementation Challenge* [40], but the benchmark library established for this challenge has been inactive for years. The DIMACS instances are furthermore difficult to use without MATLAB [50], and were reformulated at the time to eliminate free variables even though the best was to handle free variables is still an open research question [3]. No benchmark libraries were found for conic mixed-integer optimization, although supported by all major commercial optimization software available today. The closest match is probably the BIQMAC library [55], containing pure-binary quadratic optimization problems which are second-order cone representable.

The Conic Benchmark Library (CBLIB) is an ongoing community-driven project, hosted at `http://cblib.zib.de`, with aims to stay updated with the conic mixed-integer and continuous capabilities of mainstream solvers. First, however, there are concrete areas to be nursed. As seen, mixed cone types and integer variables represent cases where current benchmark libraries do not challenge state-of-the-art solvers. Even worse, the shortage of these instances alongside infeasible, dual infeasible and facially reducible problems prevent proper testing of theoretical ideas as concluded in [44] and [22]. More fundamentally, however, is the need of a file format for these conic problems that is supported across all major solvers. With CBLIB 2014, we have taken the initial steps toward addressing these issues.

First of all, a focused effort was made on gathering applications of second-order cones, as we found it to be worst represented by current benchmark libraries. In this effort, instances formulated with convex quadratic constraints have been ignored, as there is usually a natural second-order cone representation that only the problem owner can retrieve. Portfolio optimization (2) is a good example of this, where a normalized, trimmed and often rank-reduced data matrix U is the origin of the commonly used sample covariance matrix $\Sigma = U^T U$. Today, with the help of contributors from various fields, CBLIB has become the largest collection of mixed-integer and continuous second-order cone instances available online under a free and open license policy.

Second of all, a detailed analysis of existing file formats were carried out eventually leading to the Conic Benchmark Format (CBF). Looking at the old MPS format [37, 27], several extensions has been proposed over time, two of which enables second-order cone support. MOSEK [37] uses an explicit cone extension, while CPLEX [27] reuses a quadratic extension by reformulating the cone as the intersection of a half-space and a non-convex quadratic constraint. This lack of consensus is less of an issue, however, compared to the overwhelming task of augmenting the MPS format with the matrix notation for coefficients, variables and inequalities needed to realize a semidefinite cone extension. As consequence, many are currently using either the SDPA format [56], simply describing a matrix inequality, or the SEDUMI format [49], which is a MATLAB-based binary format. The CBF format can be seen as an attempt to unify the SDPA and SEDUMI format under a common conic model (presented in Section 3) and in portable clear text. The format is furthermore designed to allow maximum performance reading into C, Python and MATLAB which makes a transition to the format less cumbersome.

The article is outlined as follows. Preliminaries are provided in Section 2. In Section 3, the CBLIB standard reference for a conic problem is formalized and related to the CBF file format. In Section 4, we discuss the notion of feasibility and exact result in conic optimization, as well as the four basic solution certificates for continuous problems. Section 5 describes the selection of problem instances for this paper, as well as the tools distributed with them. Final remarks are made in Section 6.

2 Notation and cone definitions

The notation in this section uses $x = [x]^+ - [x]^-$ as the decomposition of a vector into its nonnegative and nonpositive parts. That is, element-wise, $[x]_j^+ = \max(x_j, 0)$ and $[x]_j^- = \max(-x_j, 0)$. We use $\mathcal{S}^n \subset \mathbb{R}^{n\times n}$ as the subset of symmetric matrices, and $\langle X, Y\rangle = \sum_{ij} X_{ij}Y_{ij}$ as the standard trace inner product for such matrices. The Cartesian product, $\times$, is defined to satisfy

$$x \in \mathcal{K}_x,\ y \in \mathcal{K}_y \quad \Longleftrightarrow \quad \begin{bmatrix} x \\ y \end{bmatrix} \in \mathcal{K}_x \times \mathcal{K}_y, \tag{3}$$

for column vectors and

$$X \in \mathcal{S}^{n_1},\ Y \in \mathcal{S}^{n_2} \quad \Longleftrightarrow \quad \begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix} \in \mathcal{S}^{n_1} \times \mathcal{S}^{n_2}, \tag{4}$$

for matrices. A cone which is not the Cartesian product of smaller cones is said to be primitive. That is, $\mathbb{R}^2 = \mathbb{R} \times \mathbb{R}$ is not primitive. The Euclidean distance from a point $\tilde{x}$ to its projection y in $\mathcal{K}$, is given by $\text{dist}(\tilde{x}, \mathcal{K}) = \min_{y\in\mathcal{K}} \|\tilde{x} - y\|_2$. These distances are listed in the paragraphs below for projections y as shown in [9]. The minimum distance to a point in $\mathbb{Z}$, also known as the fractionality of a scalar $\tilde{x}$, is given by $\text{dist}(\tilde{x}, \mathbb{Z}) = |\tilde{x} - \text{round}(\tilde{x})|$.

Linear cones This family covers the set of reals $\mathbb{R}^n$, the set of zeros $\{0\}^n$, the nonnegative orthant $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x_j \geq 0 \text{ for } j = 1, \dots, n\}$, and the nonpositive orthant $\mathbb{R}^n_- = \{x \in \mathbb{R}^n \mid x_j \leq 0 \text{ for } j = 1, \dots, n\}$. Infeasible points $\tilde{x}$, have a strictly positive Euclidean distance given elementwise over the primitive cones by $|\tilde{x}_j|$ for $\{0\}$, $[\tilde{x}_j]^-$ for $\mathbb{R}_+$, and $[\tilde{x}_j]^+$ for $\mathbb{R}_-$. Points in $\mathbb{R}^n$ are always feasible.

Second-order cones This family, nicknamed the ice cream cones, covers the quadratic cone $\mathcal{Q}^{1+n} = \{(r, x) \in \mathbb{R}^1_+ \times \mathbb{R}^n \mid r^2 \geq x^T x\}$ and the rotated quadratic cone $\mathcal{Q}^{2+n}_r = \{(r, x) \in \mathbb{R}^2_+ \times \mathbb{R}^n \mid 2r_1r_2 \geq x^T x\}$. Infeasible points $\tilde{x}$, have a strictly positive Euclidean distance given by

$$\text{dist}(\tilde{x},\ \mathcal{Q}^n) = \begin{cases} \left[\frac{\tilde{x}_1 - \|\tilde{x}_{2:n}\|_2}{\sqrt{2}}\right]^- & \text{if } \tilde{x}_1 \geq -\|\tilde{x}_{2:n}\|_2, \\ \|\tilde{x}\|_2 & \text{otherwise,} \end{cases}$$

and

$$\operatorname{dist}(\tilde{x},\ \mathcal{Q}_r^n) = \operatorname{dist}(T\tilde{x},\ \mathcal{Q}^n),\ \text{ where } T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \\ 0 & 0 & \mathrm{I} \end{bmatrix}.$$

We point out that the rotated quadratic cone is often encountered without the factor 2 in front of $r_1 r_2$. This is also called a restricted hyperbolic constraint, originating with [33]. We did not consider the restricted hyperbolic constraint as a separate cone, however, as it is not symmetric, making duality more cumbersome, and because its transformation to a rotated quadratic cone has no computational disadvantage.

Semidefinite cones Refers to the real-valued symmetric positive semidefinite cone $\mathcal{S}_+^n = \{X \in \mathcal{S}^n \mid \lambda(X) \in \mathbb{R}_+^n\}$, where λ is the eigenvalue function. Infeasible matrix-points $\tilde{X}$, have a strictly positive Euclidean distance defined here by $\|\,[\,\lambda(\tilde{X})\,]^-\,\|_2$ (derived from the Schatten 2-norm). We point out an often encountered alternative, $\|\,[\,\lambda(\tilde{X})\,]^-\,\|_\infty$ (derived from the induced 2-norm), but leave the discussion on the best choice open.

3 Problem formulation

The simplicity of the conic form (1) is also its weakness in practice. It implies a constraint-oriented (as opposed to a column-oriented) representation, hides a lot of information, and is bloated with identity matrices, $A_i = \mathrm{I}$, to define variable domains as used, e.g., by conic form problems in standard form [9]. To approach the first issues we stack all affine maps, $g(x) = Ax - b = (g^1(x)^T, \ldots, g^{k_g}(x)^T)^T$ where $g^i(x) = A_i x - b_i$ from (1), and constrain them to the affine map cone $\mathcal{K}_g^{n_g} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_{k_g}$, with k_g being the number of cones and n_g the total number of affine map entries. The latter issue is addressed by introducing a variable domain cone $\mathcal{K}_x^{n_x} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_{k_x}$, with k_x being the number of cones and n_x the total number of variables. These changes lead to the conic form,

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & Ax \;-\; b \;\in\; \mathcal{K}_g^{n_g}, \\ & x \in \mathcal{K}_x^{n_x},\ \text{and } x_j \in \mathbb{Z} \text{ for } j \in \mathcal{I}, \end{array} \tag{5}$$

for which dimensions can be specified as $A \in \mathbb{R}^{n_g \times n_x}$, $b \in \mathbb{R}^{n_g}$, $c \in \mathbb{R}^{n_x}$ and $|\mathcal{I}| = n_i$. The conic form (5) is still cumbersome and ambiguous, however, when it comes to semidefinite cones, as it implies the use of linear indexes into symmetric matrices. This requires a consensus regarding whether matrices are seen as column-stacked or row-stacked and whether the symmetric upper or lower triangular elements are skipped or not. To address this issue, the conic form (5) has been augmented with an explicit matrix notation. The affected variables are combined in a matrix, X, and explicitly constrained to the semidefinite variable domain, $\mathcal{S}_+^{n_X}$, which is the Cartesian product of smaller semidefinite cones. Similarly, the affected affine maps are combine in a matrix-valued affine map, $G(x)$, and constrained to the semidefinite affine map domain,

$\mathcal{S}_+^{n_G}$, which is the Cartesian product of smaller semidefinite cones. With these changes we finally arrive at the standard reference for the primal problem used in CBLIB,

$$\begin{array}{lllllll} \underset{x,X}{\text{minimize}} & c^T x + \langle C, X \rangle & & & & & \\ \text{subject to} & Ax & + & \mathcal{F}(X) & - & b & \in \mathcal{K}_g^{n_g}, \\ & \mathcal{H}^*(x) & & & - & B & \in \mathcal{S}_+^{n_G}, \\ & \multicolumn{6}{l}{x \in \mathcal{K}_x^{n_x},\ X \in \mathcal{S}_+^{n_X},\ \text{and } x_j \in \mathbb{Z} \text{ for } j \in \mathcal{I},} \end{array} \tag{P}$$

where the linear operators from matrices to vectors, $\mathcal{F}(X)$, and from vectors to matrices, $\mathcal{H}^*(x)$, are defined by

$$\mathcal{F}(X) = \begin{bmatrix} \langle F_1, X \rangle \\ \vdots \\ \langle F_{n_g}, X \rangle \end{bmatrix}, \qquad \mathcal{H}^*(x) = \sum_{j=1}^{n_x} x_j H_j.$$

These definitions match the usual semidefinite program in standard and inequality form [9], and the dimensions are given by $C \in \mathcal{S}^{n_X}$, $B \in \mathcal{S}^{n_G}$, $F_i \in \mathcal{S}^{n_X}$ for $i = 1, \ldots, n_g$, and $H_j \in \mathcal{S}^{n_G}$ for $j = 1, \ldots, n_x$. For continuous problems, the standard reference for the dual problem used in CBLIB is given by the Lagrange-dual of (P) stated similarly as

$$\begin{array}{lllllll} \underset{y,Y}{\text{maximize}} & b^T y + \langle B, Y \rangle & & & & & \\ \text{subject to} & A^T y & + & \mathcal{H}(Y) & - & c & \in -(\mathcal{K}_x^{n_x})^*, \\ & \mathcal{F}^*(y) & & & - & C & \in -(\mathcal{S}_+^{n_X})^*, \\ & \multicolumn{6}{l}{y \in (\mathcal{K}_g^{n_g})^*,\ Y \in (\mathcal{S}_+^{n_G})^*,} \end{array} \tag{D}$$

where the adjoint linear operators from vectors to matrices, $\mathcal{F}^*(y)$, and from matrices to vectors, $\mathcal{H}(Y)$, are defined by

$$\mathcal{F}^*(y) = \sum_{i=1}^{n_g} y_i F_i, \qquad \mathcal{H}(Y) = \begin{bmatrix} \langle H_1, Y \rangle \\ \vdots \\ \langle H_{n_x}, Y \rangle \end{bmatrix}.$$

Note that the domains of (D) are specified in terms of dual cones indicated by a superscripted star. Nevertheless, this is easily dealt with as all cones mentioned in this paper are self-dual, e.g., $(\mathcal{S}_+^{n_X})^* = (\mathcal{S}_+^{n_X})$, with exception of the set of reals, $\mathbb{R}^n$, and the set of zeros, $\{0\}^n$, which are each others dual cone. Now note the negation of affine map domains in the maximization problem (D). Had the objective sense of (P) been to maximize, this would have been a negation of variable domains in the minimization problem (D). To memorize this relation, it is always the variable domains of the minimization problem that is subject to the sign change. On a pedagogical remark, this dualization procedure is just as applicable and produces the same result as the sensible-odd-bizarre rules [5] for linear optimization problems, but extends to support nonlinear cones.

3.1 The file format

The instances of CBLIB 2014 are stored in the Conic Benchmark Format which has a technical specification [19] matching the conic form (P). As a matter of fact, it only differs in its choice of objective sense which can be changed from minimize to maximize. In this section we will revisit the example from the introduction and comment on its formulation in the CBF file format.

With two investments and an upper triangular covariance factor U, the Markowitz portfolio optimization problem (2) can be written in the conic form (P) as follows.

$$
\begin{array}{ll}
\underset{x_0,x_1}{\text{maximize}} & \begin{bmatrix}\mu_0\\ \mu_1\end{bmatrix}^T \begin{bmatrix}x_0\\ x_1\end{bmatrix} \\
\text{subject to} & \begin{bmatrix}g_0\\ g_1\\ g_2\\ g_3\end{bmatrix} = \begin{bmatrix}0 & 0\\ U_{00} & U_{01}\\ 0 & U_{11}\\ 1 & 1\end{bmatrix}\begin{bmatrix}x_0\\ x_1\end{bmatrix} - \begin{bmatrix}-\gamma^{1/2}\\ 0\\ 0\\ \omega\end{bmatrix} \quad \in \quad \mathcal{Q}^3 \times \{0\}, \\
& \qquad\qquad\qquad\qquad\qquad\qquad \begin{bmatrix}x_0\\ x_1\end{bmatrix} \quad \in \quad \mathbb{R}^2_+.
\end{array}
\tag{6}
$$

This problem formulation translates into the CBF file format, shown in Table 1, as follows. First we agree to the technical specification [19], by specifying that the file is written in version 1 of the CBF format (line 4-5). This has to be the first non-commentary line of the file, and is followed by a description of the problem (6) separated into model structure and problem data.

```
01 ########################
02 ## FILE INFORMATION   ##
03 ########################

04 VER
05 1

06 ########################
07 ## MODEL STRUCTURE    ##
08 ########################

09 OBJSENSE
10 MAX

11 VAR
12 2 1
13 L+ 2

14 CON
15 4 2
16 Q 3
17 L= 1

18 ########################
19 ## PROBLEM DATA       ##
20 ########################

21 OBJACOORD
22 2
23 0 $\mu_0$
24 1 $\mu_1$

25 ACOORD
26 5
27 1 0 $U_{00}$
28 1 1 $U_{01}$
29 2 1 $U_{11}$
30 3 0 1
31 3 1 1

32 BCOORD
33 2
34 0 $\gamma^{1/2}$
35 3 $-\omega$
```

Table 1: A portfolio optimization problem in the CBF file format.

Stated as model structure, the objective sense is to maximize (line 9-10). The problem has two variables in one cone (line 11-12), namely $\mathbb{R}^2_+$ (line 13), and there are four affine maps in two cones (line 14-15), namely $\mathcal{Q}^3$ (line 16) and $\{0\}$ (line 17).

Stated as problem data, the objective function has two nonzero coefficients (line 21-22), namely μ_0 for the first variable x_0 (line 23) and μ_1 for the second variable x_1 (line 24). Note that all data is specified on a sparse coordinate form like this, with indexes counting from zero. The problem has five nonzero constraint coefficients (line 25-26), listed as U_{00} for the first variable x_0 in the second affine map g_1 (line 27), and so on. Finally, there are two nonzero constraint constants (line 32-33), namely $\gamma^{1/2}$ in the first affine map g_0 (line 34) and $-\omega$ in the fourth affine map g_3 (line 35).

Relevant to the benchmarking of warmstarting capability for continuous optimization problems [47], the CBF format also introduced the `CHANGE` keyword. At the end of a problem data specification, it can be used to start a new problem data specification appending to or modifying the previous. These relative changes allows the solver to reuse internal data structures in its reoptimization after every change. In the portfolio optimization problem (6), this could be used benchmark the solvers ability to generate (risk, return)-points on an investment curve for incrementing values of γ.

4 Solution validation

Since numerical computations are performed in finite precision, small errors may accumulate throughout the solution procedure. When a solver terminates with a claimed feasible solution, it may thus deviate from the mathematically exact feasible region by some tolerances defined in the solver. While a user may want tolerances to meet the needs of a specific application, knowing that lowering them can cause numerical issues rather than better solutions, a benchmarker may instead want to align solvers with each other. In any case, it is sensible to test the final result against vendor-independent error measures.

The best way to test the validity of a solution is to translate it to its natural application-specific representation, such as a schedule, and verify it there. More generally, and especially for comparative studies, a better basis of comparison may, however, be given by the fractionality of integer variables and Euclidean distances to each cone. These measures can for instance be used when the individual formulations are studied, as it implies that bad formulations cause solvers to struggle and yield large infeasibility measures. In contrast, when the individual solvers are studied, it is unfair to blame these for the occurrences of high infeasibility caused by badly formulated instances. In this latter case, the following precautions are therefore recommended:

- *Normalize affine expressions by the infinity norm of coefficients.* By definition of a cone, the constraint $Ax + b \in \mathcal{K}$ is invariant to positive scaling. Invariance to scaling-based reformulations can also be achieved in the infeasibility measure, by computing the Euclidean distance for the normed point $(Ax + b)/\max(1, \|\text{vec}(A)\|_\infty, \|b\|_\infty)$.

- *Treat each primitive cone separately.* By definition of the Cartesian product (3), two conic constraints can be merged into one. Invariance to such reformulations can also be achieved in the infeasibility measure, by computing the Euclidean distance separately for each factor of the Cartesian product. For a block-diagonal semidefinite matrix, this corresponds to computing it separately for each block.

Without going into details, a solution can be validated in terms of these measures by comparing the Euclidean distances to some chosen absolute error tolerance. In case of fractionality, it is also common to allow some relative error such that 1 000 000.1 is accepted as integer feasible while 1.1 is not. A brief survey of this and other solution validation criteria is found in [10, Chapter 1].

A relevant question at this point is whether we are able to obtain any kind of exact results freed from such tolerances. This question is addressed in [29] and [26] using interval arithmetic, and for the general case their conclusion is negative. Points that lie exactly on the boundary of a semidefinite cone are nontrivial to verify in practice, and to compute a finite interval, guaranteed to contain the optimal value, all primal variables have to be bounded. Another approach is through symbolic-numeric quantifier elimination [28], generalizing the concept of Fourier-Motzkin elimination from linear optimization. This algorithm has doubly exponential complexity, however, and is not practical for the instances of this benchmark library. This is in sharp contrast to continuous linear optimization in which exact solutions in rational arithmetic can be obtained fairly efficiently [31].

4.1 Validating status claims

Most solvers return from a successful termination with a claim such as *the solution is optimal* or *the problem is infeasible.* In conic continuous optimization, there exists simple certificates to support such claims. In terms of problem (P), the solver has

- *certified optimality* of a feasible point, when we are given a feasible point to problem (D) with the same objective value, $c^T x + \langle C, X\rangle = b^T y + \langle B, Y\rangle$ (within a tolerance). This is a direct consequence of weak duality.

- *certified infeasibility* when we are given a feasible point to problem (D), modified such that c and C are fixed to zero, with a strictly positive objective value, $b^T y + \langle B, Y\rangle > 0$ (above a tolerance). This is the conic generalization of the Farkas' lemma from linear optimization.

- *certified dual infeasibility* when we are given a feasible point to problem (P), modified such that b and B are fixed to zero, with a strictly negative objective value, $c^T x + \langle C, X\rangle < 0$ (below a tolerance). This is also a direction in which the objective value of any primal feasible point can be improved indefinitely.

- *certified facial reducibility* when we are given a feasible point to problem (D), modified such that c and C are fixed to zero, with a zero-valued objective value, $b^T y + \langle B, Y \rangle = 0$ (within a tolerance), and non-zero entries of any self-dual cone. This is a facial reduction certificate for (P) showing it to be ill-posed in the sense of Renegar [45].

- *certified dual facial reducibility* when we are given a feasible point to problem (P), modified such that b and B are fixed to zero, with a zero-valued objective value, $c^T c + \langle C, X \rangle = 0$ (within a tolerance), and non-zero entries of any self-dual cone. This is a facial reduction certificate for (D) showing it to be ill-posed in the sense of Renegar [45].

These certificates all follow from the basic theory of conic duality [9] and facial reduction [53], and the list is complete. That is, if problem (P) cannot be certified as facially reducible (nor dual facially reducible), it either has a feasible point that can be certified as optimal, an infeasibility certificate, or a dual infeasibility certificate. In a recent result of Permenter et al. [41], the primal-dual interior-point method [39] is shown capable of always finding one of these certificates in theory. Hence, there might be a cure for the numerical issues faced by all current implementations of the algorithm when used on facially reducible problems [23, 54].

5 The instance catalog

This section brings an overview of the problem instances in CBLIB 2014. A brief description of each instance is found in Table 2, along with references to the researchers who worked with and described the instances. Most instances have been found by data mining in the public domain, and the contributors of these instances to the CBLIB project have been recognized in the distributed benchmark library. Note that semidefinite cones are absent from this initial release due to our focus on second-order cones.

The instance statistics are found in Table 3. For each instance the table shows the total number of variables (*var*), affine expressions (*map*), and nonzero constraint coefficients (*nnz*) not counting constants and objective coefficients. It then shows the number of primitive linear (*lin*) and second-order (*so*) cones counted separately for each cone dimension. Primitive linear cones are always one-dimensional, and for second-order cones the dimension is followed by a colon and its count in a comma-separated list. Next follows the number of binary variables defined in a linear (b_{lin}) and second-order (b_{so}) variable domain cone. Similarly, the table shows the number of general integer variables defined in a linear (I_{lin}) and second-order (I_{so}) variable domain cone. The last columns indicate the instance status. The column (*obj*) reports the best primal objective

Packs	Origin and description	Instances
chainsing	Conn et al. [14], Kobayashi et al. [30]. The chained singular function (academic).	9
estein	Drewes [16]. Minimum Steiner tree problem.	9
filterdesign	Coleman et al. [13]. Optimal design of a delta-sigma ('ds' in name), a wideband ('wb' in name) or a nonlinear-phase FIR ('fir' in name) filter.	12
nb	Coleman and Vanderbei [12]. Calibration of antenna arrays, suppressing signals that do not come from a chosen direction.	4
portfoliocard	Vielma et al. [52]. Portfolio optimization with cardinality constraints.	24
pp	Ziegler [57]. Production planning.	8
sched	Skutella [48]. Job scheduling on parallel unrelated machines.	8
sssd	Bonami et al. [7], Elhedhli [17]. Stochastic service system design with M/M/1 queues using Strong formulation ('strong' in name), or weak formulation ('weak' in name).	16
strain	Andersen et al. [2], Christiansen and Andersen [11]. Collapse states for loaded plastic plates using the plain strain model ('nql' in name), or the supported plate model ('qssp' in name).	8
turbine	Drewes [16]. Balancing high-speed rotating machinery with either the least axial weight locations, the least distinct weight sets ('GF' in name), or minimum imbalance ('lowb' in name).	7
uflquad	Bonami et al. [7], Günlük et al. [24]. Separable quadratic uncapacitated facility location. With cuts ('psc' in name) or without cuts ('nopsc' in name).	16

Table 2: Description of packs in the CBLIB 2014 selection with references to the researchers who worked with and described the 121 instances.

	Size			Conic domains		Binary		Integer		Status		
Instances	var	map	nnz	lin	so	lin	so	lin	so	obj	M	C
chainsing												
chainsing-1000-1	12976	9982	17966	13976	[3:2994]					3.0180E+01	0	0
chainsing-1000-2	9985	7988	14975	10985	[3:1996, 1000:1]					3.0180E+01	0	0
chainsing-1000-3	6991	5992	11981	7991	[3:998, 1998:1]					3.0180E+01	0	0
chainsing-10000-1	129976	99982	179966	139976	[3:29994]					3.0261E+02	0	0
chainsing-10000-2	99985	79988	149975	109985	[3:19996, 10000:1]					3.0261E+02	0	0
chainsing-10000-3	69991	59992	119981	79991	[3:9998, 19998:1]					3.0261E+02	0	0
chainsing-50000-1	649976	499982	899966	699976	[3:149994]					1.5134E+03	0	0
chainsing-50000-2	499985	399988	749975	549985	[3:99996, 50000:1]					1.5134E+03	0	0
chainsing-50000-3	349991	299992	599981	399991	[3:49998, 99998:1]					1.5134E+03	0	0
estein												
estein4_A	67	108	128	148	[3:9]	9	0	0	0	8.0137E−01	0	0
estein4_B	67	108	128	148	[3:9]	9	0	0	0	1.1881E+00	0	0
estein4_C	67	108	128	148	[3:9]	9	0	0	0	1.0727E+00	0	0
estein4_nr22	67	108	128	148	[3:9]	9	0	0	0	5.0329E−01	0	0
estein5_A	132	211	258	289	[3:18]	18	0	0	0	1.0454E+00	0	0
estein5_B	132	211	258	289	[3:18]	18	0	0	0	1.1932E+00	0	0
estein5_C	132	211	258	289	[3:18]	18	0	0	0	1.4991E+00	0	0
estein5_nr1	132	211	258	289	[3:18]	18	0	0	0	1.6644E+00	0	0
estein5_nr21	132	211	258	289	[3:18]	18	0	0	0	1.8182E+00	0	0
filterdesign												
2013_dsNRL	61822	1616	66668564	1616	[3:20503, 313:1]					-9.6379E−06	0	0
2013_firL1	59706	20902	39787428	20902	[3:19902]					-3.6669E+00	0	0

2013_firL1Linfalph	119412	20903	79574856	20903	[3:39804]					-3.3116E+00	0	-
2013_firL1Linfeps	59173	30085	9873426	30086	[3:19724]					-1.5255E−02	0	-
2013_firL2L1alph	49612	30268	9985771	30269	[3:9922, 19845:1]					-2.4441E−01	0	0
2013_firL2L1eps	60708	20903	40288929	20903	[3:19902, 1002:1]					-3.0683E+00	0	0
2013_firL2Linfalph	91783	2002	121660011	2002	[3:29927, 2002:1]					-7.7910E−02	0	0
2013_firL2Linfeps	59636	24655	19108570	24655	[3:11927, 23855:1]					-1.0141E−02	0	0
2013_firL2a	10002	10001	50015001	10001	[10002:1]					-1.4368E−01	0	0
2013_firLinf	59856	2001	79771354	2001	[3:19952]					-1.0022E−02	0	-
2013_wbNRL	40450	1042	39138234	38123	[38:7, 1035:1, 2068:1]					-3.8759E−05	0	0
2013i_wbNRL	63312	1710	101934231	59827	[51:4, 52:3, 1431:1, 3404:1]					Unbounded	DI	P
nb												
nb	2383	123	191519	127	[3:793]					-5.0703E−02	0	0
nb_L1	3176	915	192312	1712	[3:793]					-1.3012E+01	0	0
nb_L2	4195	123	402285	127	[3:838, 1677:1]					-1.6290E+00	0	0
nb_L2_bessel	2641	123	208817	127	[3:838, 123:1]					-1.0257E−01	0	0
portfoliocard												
classical_50_1	152	255	2902	356	[51:1]	50	0	0	0	-9.4760E−02	0	0
classical_50_2	152	255	2902	356	[51:1]	50	0	0	0	-9.0528E−02	0	0
classical_50_3	152	255	2902	356	[51:1]	50	0	0	0	-8.8041E−02	0	0
classical_200_1	602	1005	41602	1406	[201:1]	200	0	0	0	-1.1668E−01[v]	P	P
classical_200_2	602	1005	41602	1406	[201:1]	200	0	0	0	-1.1009E−01[v]	P	P
classical_200_3	602	1005	41602	1406	[201:1]	200	0	0	0	-1.0607E−01[v]	P	P
robust_50_1	207	365	5564	468	[52:2]	51	0	0	0	-8.5695E−02	0	0
robust_50_2	207	365	5564	468	[52:2]	51	0	0	0	-1.4365E−01	0	0
robust_50_3	207	365	5564	468	[52:2]	51	0	0	0	-8.9803E−02	0	0
robust_100_1	407	715	21114	918	[102:2]	101	0	0	0	-7.2090E−02	0	P
robust_100_2	407	715	21114	918	[102:2]	101	0	0	0	-9.1574E−02	0	0
robust_100_3	407	715	21114	918	[102:2]	101	0	0	0	-1.1682E−01	0	0
robust_200_1	807	1415	82214	1818	[202:2]	201	0	0	0	-1.4275E−01	0	P
robust_200_2	807	1415	82214	1818	[202:2]	201	0	0	0	-1.2167E−01	0	P
robust_200_3	807	1415	82214	1818	[202:2]	201	0	0	0	-1.2911E−01[v]	P	P
shortfall_50_1	205	361	5612	464	[51:2]	51	0	0	0	-1.1018E+00	0	0
shortfall_50_2	205	361	5612	464	[51:2]	51	0	0	0	-1.0952E+00	0	0
shortfall_50_3	205	361	5612	464	[51:2]	51	0	0	0	-1.0923E+00	0	0
shortfall_100_1	405	711	21212	914	[101:2]	101	0	0	0	-1.1063E+00	0	P
shortfall_100_2	405	711	21212	914	[101:2]	101	0	0	0	-1.1007E+00[v]	P	P
shortfall_100_3	405	711	21212	914	[101:2]	101	0	0	0	-1.1031E+00	0	P
shortfall_200_1	805	1411	82412	1814	[201:2]	201	0	0	0	-1.1354E+00[v]	P	P
shortfall_200_2	805	1411	82412	1814	[201:2]	201	0	0	0	-1.1254E+00[v]	P	P
shortfall_200_3	805	1411	82412	1814	[201:2]	201	0	0	0	-1.1199E+00[v]	P	P
pp												
pp-n10-d10	50	31	59	51	[3:10]	0	10	0	0	7.2481E+01	0	0
pp-n10-d10000	50	31	59	51	[3:10]	0	10	0	0	1.4815E+03	0	-
pp-n100-d10	500	301	599	501	[3:100]	0	100	0	0	7.7728E+02[v]	P	P
pp-n100-d10000	500	301	597	501	[3:100]	0	100	0	0	1.9856E+04	0	-
pp-n1000-d10	5000	3001	5969	5001	[3:1000]	0	1000	0	0	7.3434E+03[v]	P	P
pp-n1000-d10000	5000	3001	5968	5001	[3:1000]	0	1000	0	0	2.1611E+05	P	-
pp-n100000-d10	500000	300001	597382	500001	[3:100000]	0	100000	0	0	0.0000E+00[a]	-	-
pp-n100000-d10000	500000	300001	597463	500001	[3:100000]	0	100000	0	0	1.8348E+07[a]	-	-
sched												
sched_50_50_orig	4979	2527	25488	5029	[3:1, 2474:1]					2.6673E+04[a]	-	-
sched_50_50_scaled	4977	2526	27985	5028	[2475:1]					7.8520E+00	0	-
sched_100_50_orig	9746	4844	55291	9846	[3:1, 4741:1]					1.8189E+05	-	0
sched_100_50_scaled	9744	4843	60288	9845	[4742:1]					6.7165E+01	0	-
sched_100_100_orig	18240	8338	104902	18340	[3:1, 8235:1]					7.1737E+05	-	0
sched_100_100_scaled	18238	8337	114899	18339	[8236:1]					2.7331E+01	0	-
sched_200_100_orig	37889	18087	260503	38089	[3:1, 17884:1]					1.4136E+05	-	0
sched_200_100_scaled	37887	18086	280500	38088	[17885:1]					5.1812E+01	0	-
sssd												
sssd-strong-15-4	125	180	372	269	[3:12]	72	0	0	0	3.2800E+05	0	0
sssd-strong-15-8	249	344	744	521	[3:24]	144	0	0	0	6.2251E+05	0	0
sssd-strong-20-4	145	205	432	314	[3:12]	92	0	0	0	2.8781E+05	0	0
sssd-strong-20-8	289	389	864	606	[3:24]	184	0	0	0	6.0035E+05	0	0
sssd-strong-25-4	165	230	492	359	[3:12]	112	0	0	0	3.1172E+05	0	0
sssd-strong-25-8	329	434	984	691	[3:24]	224	0	0	0	5.0075E+05	P	0
sssd-strong-30-4	185	255	552	404	[3:12]	132	0	0	0	2.6413E+05	0	0

sssd-strong-30-8	369	479	1104	776	[3:24]	264	0	0	0	5.2876E+05	P	O
sssd-weak-15-4	125	180	360	269	[3:12]	72	0	0	0	3.2800E+05	O	O
sssd-weak-15-8	249	344	720	521	[3:24]	144	0	0	0	6.2251E+05	O	O
sssd-weak-20-4	145	205	420	314	[3:12]	92	0	0	0	2.8781E+05	O	O
sssd-weak-20-8	289	389	840	606	[3:24]	184	0	0	0	6.0034E+05	P	O
sssd-weak-25-4	165	230	480	359	[3:12]	112	0	0	0	3.1172E+05	O	O
sssd-weak-25-8	329	434	960	691	[3:24]	224	0	0	0	5.0075E+05	P	O
sssd-weak-30-4	185	255	540	404	[3:12]	132	0	0	0	2.6413E+05	O	O
sssd-weak-30-8	369	479	1080	776	[3:24]	264	0	0	0	5.2876E+05	P	O
strain												
nql30	4501	6380	20569	8181	[3:900]					-9.4602E−01	O	O
nql60	18001	25360	82539	32561	[3:3600]					-9.3504E−01	O	O
nql90	40501	56940	185909	73141	[3:8100]					-9.3136E−01	O	O
nql180	162001	227280	744419	292081	[3:32400]					-9.2764E−01	O	O
qssp30	7565	11255	44414	11256	[4:1891]					-6.4967E+00	O	O
qssp60	29525	44105	178814	44106	[4:7381]					-6.5627E+00	O	O
qssp90	65885	98555	403214	98556	[4:16471]					-6.5942E+00	O	O
qssp180	261365	391505	1616414	391506	[4:65341]					-6.6391E+00	O	O
turbine												
turbine07	84	101	313	101	[3:25, 9:1]	0	0	11	0	2.0000E+00	O	O
turbine07GF	87	124	444	136	[3:25]	12	0	0	0	3.0000E+00	P	O
turbine07_aniso	83	108	313	116	[3:25]	0	0	11	0	3.0000E+00	O	O
turbine07_lowb	212	354	621	480	[2:1, 3:25, 9:1]	56	0	0	0	8.9930E−01	O	O
turbine07_lowb_aniso	210	361	621	496	[3:25]	56	0	0	0	1.3945E+00	-	O
turbine54	366	477	2099	477	[3:119, 9:1]	0	0	11	0	3.0000E+00	P	O
turbine54GF	369	500	2982	512	[3:119]	12	0	0	0	4.0000E+00	P	O
uflquad												
uflquad-nopsc-10-100	3011	5111	7010	5122	[3:1000]	10	0	0	0	5.4029E+02	O	O
uflquad-nopsc-10-150	4511	7661	10510	7672	[3:1500]	10	0	0	0	7.0965E+02	O	O
uflquad-nopsc-20-100	6021	10121	14020	10142	[3:2000]	20	0	0	0	3.9954E+02	O	O
uflquad-nopsc-20-150	9021	15171	21020	15192	[3:3000]	20	0	0	0	5.6872E+02	O	O
uflquad-nopsc-30-100	9031	15131	21030	15162	[3:3000]	30	0	0	0	3.5524E+02	O	P
uflquad-nopsc-30-150	13531	22681	31530	22712	[3:4500]	30	0	0	0	4.6816E+02[v]	P	P
uflquad-nopsc-30-200	18031	30231	42030	30262	[3:6000]	30	0	0	0	5.5491E+02[v]	P	P
uflquad-nopsc-30-300	27031	45331	63030	45362	[3:9000]	30	0	0	0	7.8479E+02[v]	P	P
uflquad-psc-10-100	3011	5111	8010	5122	[3:1000]	10	0	0	0	5.4029E+02	O	O
uflquad-psc-10-150	4511	7661	12010	7672	[3:1500]	10	0	0	0	7.0965E+02	O	O
uflquad-psc-20-100	6021	10121	16020	10142	[3:2000]	20	0	0	0	3.9954E+02	O	O
uflquad-psc-20-150	9021	15171	24020	15192	[3:3000]	20	0	0	0	5.6872E+02	O	O
uflquad-psc-30-100	9031	15131	24030	15162	[3:3000]	30	0	0	0	3.5524E+02	O	O
uflquad-psc-30-150	13531	22681	36030	22712	[3:4500]	30	0	0	0	4.6816E+02	O	O
uflquad-psc-30-200	18031	30231	48030	30262	[3:6000]	30	0	0	0	5.5491E+02	O	O
uflquad-psc-30-300	27031	45331	72030	45362	[3:9000]	30	0	0	0	7.6035E+02	O	O

[a(currancy)] Infeasibility meassures exceed 10^{-4} on some primitive cones or integer requirements (points not normalized).
[v(alue)] Objective neither claimed by a solver to be within an absolute and relative gap of 0.0 from optimality (mixed-integer case), nor certified to be within an absolute gap of 10^{-4} or relative gap of 10^{-7} from optimality (continuous case).

Table 3: CBLIB 2014 instance statistics.

value, whenever possible, among the primal feasible points found using MOSEK version 7.1.0.12 [37] and CPLEX version 12.6.0.0 [27] on a 64-bit linux platform. Default parameters settings were used in these runs, except for forcing single-threaded behavior, a time limit of one hour, as well as an absolute and relative optimality gap of zero for integer problems. Superscripts are appended to this column, *obj*, when solutions could not be validated using the tolerances on feasibility and optimality stated in the footnotes of the table. The same tolerances are used to label the output of MOSEK (column M) and CPLEX (column C). A dash, -, means that the output neither validated as a primal feasible point nor a certificate of any kind. Moreover, P means primal feasibility, `O` means optimality is claimed (mixed-integer case) or certified (continuous case), and `DI` means that a dual infeasibility certificate was recognized.

Overall, the CBLIB 2014 selection of instances can be described as follows. The library contains 121 instances out of which 80 are mixed-integer. Only eight of the 80 mixed-integer instances contain general integer variables, showing binary variables to be the most common as expected. This is in line with the mixed-integer linear instances of the *MIPLIB* library [32]. All instances contain second-order cones, but only three of the 80 mixed-integer instances require the entry of a second-order cone to be integer. Beware, that this latter observation is based solely on the domain of integer variables, and does not consider affine expression entries even though they might also be implied integer.

The average number of entries per second-order cone is close to three in many of the instances. Elaborating on this, 86 of the 121 instances contain at least one 3-dimensional second-order cone out of which 20 have exactly one other and 66 have no other second-order cones. In the other end of the scale we find nine of the 121 instances with more than a thousand entries per second-order cone on average. The total of second-order cones range as low as one (eleven instances) to more than 100 000 (three instances).

We will now elaborate on the differences between MOSEK and CPLEX as shown in Table 3, starting with instance `2013i_wbNRL`. This instance is an example of the fact that it is quite normal to make mistakes or forget something in the first attempt to formulate a problem. In this particular case, the problem features a direction which may improve the objective value of any feasible point indefinitely, and this direction is a dual infeasibility certificate. MOSEK found this certificate, while CPLEX terminated with a primal feasible point.

Another observation from Table 3 is that the "best" formulation is not always clear. MOSEK terminated with primal infeasibilities on all `sched_*_*_orig` instances, but solved all of the `sched_*_*_scaled` instances just fine. Thus, what can be solved and not is exactly opposite to CPLEX, with `sched_50_50_orig` as the only exception for which CPLEX also terminated with primal infeasibilities. Only together, were they able to solve nearly all of the `sched` instances.

Numerical issues is unfortunately not an isolated case, however, as CPLEX also terminated with primal infeasibilities on `2013_firL1Linfalph` as well as on `2013_firL1Linfeps`. Moreover, MOSEK refused to claim optimality on `turbine07GF`, `turbine54` and `turbine54GF`, even though terminating in time with the optimal solutions, presumably because numerical issues forced it to skip subproblems rather than to prune them from the search tree. There were also integer problems where optimality was claimed, but infeasible solutions were returned. This happened for CPLEX on all of the `pp-*-d10000` instances and for MOSEK on `turbine07_lowb_aniso`. In one case, `sssd-weak-30-8`, MOSEK moreover seems to have cut off the optimal solution as it claimed optimality although an objective improvement of 5.4 in absolute and 1.0E-05 in relative measures could be achieved.

Finally, to indicate the hardness of these instances in terms different from numerical issues, we may state that neither CPLEX nor MOSEK were able to find any feasible solution to `pp-n100000-d10` in time. Interestingly, this is a trivial task to perform by hand as seen by consulting the mathematical model [57]. It is also worth pointing out that the continuous instances of the filterdesign pack are absolutely huge, and CPLEX actually timed out on `2013_dsNRL` and `2013_firLinf`. This, despite actually outputting a valid solution and optimality certificate in the former case. On the integer problems, CPLEX and MOSEK timed out 20 and 21 times respectively. The 14 instances on which they both timed out is given by the special case of `pp-n100000-d10` (no solutions found) and otherwise match when the letter `P` appears simultaneously in column M and C of Table 3.

5.1 Filtering out instances of interest

Cone support is not uniform across all solvers, and it is often the case that benchmarks focus on a subset of instances with certain characteristics. For this reason the Python script, `filter.py`, has been developed to filter out instances of interest. It takes a string as input, substitute all occurrences of `||*|...||` with the value of the filter `*` given arguments `...`, and evaluate it as a boolean expression. Instances evaluating to true are listed.

```
python filter.py "||cones|so|| == ||cones|so|==3||"
```

Instances where all second-order cones have exactly three entries. In this command, `||cones||` counts the number of conic domains, and takes two arguments to limit its scope. The first argument specifies a cone type following the CBF format, with linear cones `F`, `L+`, `L-`, `L=`, or all four, `lin`, as well as second-order cones `Q`, `QR`, or both, `so`. The second argument is a relation with cone dimension as left-hand-side.

```
python filter.py "||int|| and ||cones|so|| and not ||psdcones||"
```

Mixed-integer second-order cone instances. This command uses Python boolean logic with `||int||` counting integer variables (the subset of binary variables is found by `||binary||`), and `||psdcones||` counting semidefinite cones.

```
python filter.py "||entries|so|| / ||cones|so|| <= 4"
```

Instances with no more than four entries per second-order cone on average. This command shows the use of Python mathematics, with `||entries||` summing the dimension of cones (here limited to second-order cones).

The script also accepts an execution argument, indicated by `-x`, whose result will be evaluated and printed. With an empty filter (always true), this can be used to generate tables of instance statistics.

```
python filter.py "" -x "[||path||, ||minimize||, ||var|F||, ||map|L=||]"
```

Instance statistics for all instances. The filter `||path||` is filepath (`||name||` is filename without extension) and `||minimize||` is whether the objective sense is to minimize. `||var||` and `||map||` are subsets of `||entries||` limited respectively to $\mathcal{K}_x^{n_x}$ and $\mathcal{K}_g^{n_g}$ from (P). Here, the former is further limited to free variables, and the latter to equality constraints.

This argument can be useful for exploring the instances and filters. Note that the filtering mechanism is implemented as a plugin system which can be extended by adding functions to the directory of filters in the distributed library.

5.2 Feeding instances into optimization software

A disadvantage of the CBF format is the lack of support in mainstream software. This concern has led to the development of tools which can aid integration with, or transformation to, the input format of most software packages. More specifically, the library is distributed with CBF parsers in various programming languages and a file converter tool.

Parsers of the CBF format has been written in the MATLAB, Python, and C++ programming languages. These parsers may be used to feed instances into optimization software through programming interfaces. An example of this concept has been made with the Python script, `run.py`, which uses the CBF parser in Python to feed instances into MOSEK [37] through its Python API. This script was, for example, used to generate the last column of Table 3 in the instance catalog. By default, the script is configured to save the optimization result of each instance with the extension, `.sol`. Subsequent analysis with the Python script, `summary.py`, is thus possible.

`python run.py runmosek -f [CBFFILE1] [CBFFILE2] ...`
Runs MOSEK on the listed instances, that is, `[CBFFILE1]`, `[CBFFILE2]`, and so on. A summary of these results can be shown by `python summary.py -f [CBFFILE1] [CBFFILE2] ...`.

`python run.py runmosek -s [SET]`
Runs MOSEK on the instances in `[SET]`. This can be a subdirectory of *cbf* in the distributed library, or a file formatted as the default output of the `filter.py` script (a stripped version of `ref.csv` in the distributed library). The summary is shown by `python summary.py -s [SET]`.

The file converter tool, named `cbftool`, uses the CBF parser written in C++ to convert instances into another file format. It is capable of transforming conic constraints, $Ax - b \in \mathcal{K}$, into $Ax - b = s$ and $s \in \mathcal{K}$, but is otherwise incapable of modifying problem formulations to match the limitations of a particular file format. Thus, although the sparse SDPA format [56] is supported by `cbftool`, nothing but matrix inequalities can be converted to this format. The tool supports the two extensions of the MPS format, mentioned in the introduction, to facilitate second-order cones. Examples of this are given below.

`cbftool -o mps-mosek [CBFFILE1] [CBFFILE2] ...`
Convert listed instances to the MPS format using the explicit second-order cone extension. Results are stored in the current directory.

`cbftool -o mps-cplex -opath [OUTPUTDIR] [CBFFILE1] [CBFFILE2] ...`
Convert listed instances to the MPS format using the quadratic extension with nonnegative variable bounds for second-order cones. Results are stored in `[OUTPUTDIR]`.

6 Final remarks

Conic optimization has become mainstream during the past ten years. Excellent commercial and open source solvers are available, frequent advancements are being made, and its potential usage stretch all the way to general convex optimization. Several issues have been identified in the availability of benchmark libraries, however, which may potentially slow down progress. Some of these issues have been addressed with the release of CBLIB 2014. There is now a large collection of mixed-integer and continuous second-order cone instances, and a new CBF file format which unifies the SDPA and SEDUMI format under a common mathematical formulation.

Since this publication, CBLIB has been used by XPRESS [18] (mentioned in [42]), by MOSEK [37] and GUROBI [25] (private communication), as well as in the public benchmarks of Hans Mittelmann [36]. Moreover, the library has continued to grow from a few hundred to more than a thousand instances distributed online. While this expansion includes new applications of conic optimization, it mostly provides a wider variety of data for some of the mathematical models, and some very hard and unsolved problems which are not suited for performance benchmarks. CBLIB 2014 thus remains representative as a benchmark selection of the entire collection.

Future work includes categorizing the instances into test sets similar to the sets of open, challenging and easy instances found in MIPLIB [32]. Adding native support of the CBF format to the open source solvers and algebraic modeling tools is also of high value to the project. This has already started to happen with PICOS [46] as first mover. Finally, we are interested in instances with properties rare to the existing library such as infeasibilities (as requested in [44]) or integer variables in cones (as requested in [22]), or simply representing new applications of conic optimization.

The Conic Benchmark Library, CBLIB, is a community project and grow through external submissions. Please consider contributing at `http://cblib.zib.de`.

Acknowledgments The author owe a special thanks to Erling D. Andersen and Mathias Stolpe who supervised the work on CBLIB, and to Ambros Gleixner and Thorsten Koch for hosting and maintaining the benchmarking library at Zuse Institute Berlin. Another great thank you goes to the anonymous peer reviewers for their valuable feedback, and to the many who has contributed instances or given feedback to drive the benchmark library forward. The author, Henrik A. Friberg, was funded by MOSEK ApS and the Danish Ministry of Higher Education and Science through the Industrial PhD project "Combinatorial Optimization over Second-Order Cones and Industrial Applications".

References

[1] F. Alizadeh and G. Goldfarb. Second-order cone programming. *Mathematical programming*, 51:3–51, 2003.

[2] K. D. Andersen, E. Christiansen, and M. L. Overton. Computing Limit Loads by Minimizing a Sum of Norms. *SIAM Journal on Scientific Computing*, 19(3):1046–1062, 1998.

[3] M. F. Anjos and S. Burer. On handling free variables in interior-point methods for conic linear optimization. *SIAM Journal on Optimization*, 18(4):1310–1325, 2007.

[4] A. Ben-tal and A. Nemirovski. *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, 2001. ISBN 978-0-89871-491-3.

[5] A. T. Benjamin. Sensible Rules for Remembering Duals - the SOB Method. *SIAM review*, 37 (1):85—-87, 1995.

[6] T. Berthold, S. Heinz, and S. Vigerske. Extending a CIP framework to solve MIQCPs. In *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 427–444. Springer New York, 2012.

[7] P. Bonami, M. Kilinc, and J. Linderoth. Algorithms and software for convex mixed integer nonlinear programs. *Mixed Integer Nonlinear Programming*, 154:1–39, 2012.

[8] B. Borchers. SDPLIB 1.2, a library of semidefinite programming test problems. *Optimization Methods and Software*, 11(1):683–690, 1999.

[9] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004. `https://www.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf`.

[10] J. W. Chinneck. *Feasibility and Infeasibility in Optimization*, volume 118 of *International Series in Operations Research and Management Science*. Springer US, Boston, MA, 2008.

[11] E. Christiansen and K. D. Andersen. Computation of collapse states with von Mises type yield condition. *International Journal for Numerical Methods in Engineering*, 46(8):1185–1202, 1999.

[12] J. O. Coleman and R. J. Vanderbei. Random-process formulation of computationally efficient performance measures for wideband arrays in the far field. In *Proceedings of the 42nd Midwest Symposium on Circuits and Systems*, volume 2, pages 761–764, 1999.

[13] J. O. Coleman, D. P. Scholnik, and J. J. Brandriss. A specification language for the optimal design of exotic FIR filters with second-order cone programs. *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, 1:341–345, 2002.

[14] A. R. Conn, N. I. M. Gould, and P. L. Toint. Testing a class of methods for solving minimization problems with simple bounds on the variables. *Mathematics of Computation*, 50(182):399–430, 1988.

[15] E. de Klerk and R. Sotirov. A new library of structured semidefinite programming instances. *Optimization Methods and Software*, 24(6):959–971, Dec. 2009.

[16] S. Drewes. *Mixed Integer Second Order Cone Programming*. PhD thesis, Department of Mathematics, Technical University Darmstadt, 2009. `http://www3.mathematik.tu-darmstadt.de/fileadmin/home/users/82/DissertationDrewes.pdf`.

[17] S. Elhedhli. Service System Design with Immobile Servers, Stochastic Demand, and Congestion. *Manufacturing & Service Operations Management*, 8(1):92–97, 2006.

[18] Fair Isaac Corporation. Xpress-Optimizer Reference manual, Release 20.00. Technical report, 2009. `http://www.fico.com/en/wp-content/secure_upload/Xpress-Optimizer-Refernce-Manual.pdf`.

[19] H. A. Friberg. The Conic Benchmark Format: Version 1 - Technical Reference Manual. Technical Report E-0047, Department of Wind Energy, Technical University of Denmark, 2014. `http://orbit.dtu.dk/services/downloadRegister/88492586/Conic_Benchmark_Format.pdf`.

[20] D. M. Gay. Electronic mail distribution of linear programming test problems. *Mathematical Programming Society COAL Newsletter*, 13:10–12, 1985.

[21] F. Glineur. Conic optimization: an elegant framework for convex optimization. *Belgian Journal of Operations Research, Statistics and Computer Science*, 41:5–28, 2001.

[22] J. C. Góez. *Mixed Integer Second Order Cone Optimization Disjunctive Conic Cuts: Theory and experiments*. PhD thesis, Lehigh University, 2013.

[23] G. Gruber and F. Rendl. Computational experience with ill-posed problems in semidefinite programming. *Computational Optimization and Applications*, 21(2):201–212, 2002.

[24] O. Günlük, J. Lee, and R. Weismantel. MINLP strengthening for separable convex quadratic transportation-cost UFL. Technical report, IBM Research Report RC24213, 2007. http://domino.watson.ibm.com/library/cyberdig.nsf/papers/F93A77B97033B1878525729F0051C343/$File/rc24213.pdf.

[25] Gurobi Optimization, Inc. Gurobi Optimizer Reference Manual - Version 5.6. Technical report, 2013. http://www.gurobi.com/documentation/5.6/reference-manual/refman.pdf.

[26] V. Härter, C. Jansson, and M. Lange. VSDP: A Matlab toolbox for verified semidefinite-quadratic-linear programming. Technical report, Institute for Reliable Computing, Hamburg University of Technology, 2012. http://www.ti3.tu-harburg.de/jansson/vsdp/VSDP2012Guide.pdf.

[27] IBM Corporation. IBM ILOG CPLEX Optimization Studio V12.6.0 documentation. Technical report, 2014. www-01.ibm.com/support/knowledgecenter/SSSA5P_12.6.0.

[28] H. Iwane, H. Yanami, H. Anai, and K. Yokoyama. An effective implementation of symbolic–numeric cylindrical algebraic decomposition for quantifier elimination. *Theoretical Computer Science*, 479:43–69, 2013.

[29] C. Jansson. Guaranteed Accuracy for Conic Programming Problems in Vector Lattices. Technical report, Institute for Reliable Computing, Technical University Hamburg, 2007. http://arxiv.org/pdf/0707.4366v1.

[30] K. Kobayashi, S. Kim, and M. Kojima. Sparse second order cone programming formulations for convex optimization problems. *Journal of the Operations Research Society of Japan*, 51 (3):241–264, 2008.

[31] T. Koch. The final NETLIB-LP results. *Operations Research Letters*, 32(2):138–142, 2004.

[32] T. Koch, T. Achterberg, E. D. Andersen, O. Bastert, T. Berthold, R. E. Bixby, E. Danna, G. Gamrath, A. M. Gleixner, S. Heinz, A. Lodi, H. D. Mittelmann, T. Ralphs, D. Salvagnin, D. E. Steffy, and K. Wolter. MIPLIB 2010. *Mathematical Programming Computation*, 3(2): 103–163, 2011.

[33] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.

[34] H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952.

[35] S. Mars and L. Schewe. An SDP-Package for SCIP. Technical Report August, TU Darmstadt, 2012. URL http://www.opt.tu-darmstadt.de/~smars/scip_sdp.html.

[36] H. D. Mittelmann. MISOCP and large SOCP Benchmark, 2014. URL http://plato.asu.edu/ftp/socp.html.

[37] MOSEK ApS. The MOSEK C Optimizer API manual, Version 7.0. Technical report, 2013. http://docs.mosek.com/7.0/capi.pdf.

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

[39] Y. Nesterov, M. J. Todd, and Y. Ye. Infeasible-start primal-dual methods and infeasibility detectors for nonlinear programming problems. *Mathematical Programming*, 84(2):227–267, 1999.

[40] G. Pataki and S. H. Schmieta. The DIMACS library of semidefinite-quadratic-linear programs. Technical report, Computational Optimization Research Center, Columbia University, 2002. http://dimacs.rutgers.edu/Challenges/Seventh/Instances/lib.ps.

[41] F. Permenter, H. A. Friberg, and E. D. Andersen. Solving conic optimization problems via self-dual embedding and facial reduction : a unified approach. Technical report, 2015. http://www.optimization-online.org/DB_HTML/2015/09/5104.html.

[42] M. Perregaard. Advances in Convex Quadratic Integer Optimization with Xpress, 2014. Presented at INFORMS Annual Meeting 2014.

[43] I. Pólik. Conic optimization software. In J. J. Cochran, L. A. Cox Jr., P. Keskinocak, J. P. Kharoufeh, and J. C. Smith, editors, *Wiley Encyclopedia of Operations Research and Management Science*. 2011.

[44] I. Pólik and T. Terlaky. New stopping criteria for detecting infeasibility in conic optimization. *Optimization Letters*, 3(2):187–198, 2009.

[45] J. Renegar. Incorporating Condition Measures into the Complexity Theory of Linear Programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995.

[46] G. Sagnol. PICOS: A Python Interface for Conic Optimization Solvers, 2015. URL `http://picos.zib.de`.

[47] A. Skajaa, E. D. Andersen, and Y. Ye. Warmstarting the homogeneous and self-dual interior point method for linear and conic quadratic problems. *Mathematical Programming Computation*, 5(1):1–25, 2012.

[48] M. Skutella. Convex quadratic and semidefinite programming relaxations in scheduling. *Journal of the ACM*, 48(2):206–242, 2001.

[49] J. F. Sturm. Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

[50] The MathWorks, Inc. MATLAB Primer, R2013b. Technical report, 2013. `http://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf`.

[51] R. H. Tütüncü, K. C. Toh, and M. J. Todd. Solving semidefinite-quadratic-linear programs using SDPT3. *Mathematical Programming*, 95(2):189–217, 2003.

[52] J. P. Vielma, S. Ahmed, and G. L. Nemhauser. A Lifted Linear Programming Branch-and-Bound Algorithm for Mixed Integer Conic Quadratic Programs. *INFORMS Journal on Computing*, 20:438–450, 2008.

[53] H. Waki and M. Muramatsu. Facial reduction algorithms for conic optimization problems. *Journal of Optimization Theory and Applications*, 158(1):188–215, 2013.

[54] H. Waki, M. Nakata, and M. Muramatsu. Strange behaviors of interior-point methods for solving semidefinite programming problems in polynomial optimization. *Computational Optimization and Applications*, 53(3):823–844, Sept. 2011.

[55] A. Wiegele. Biq Mac Library - A collection of Max-Cut and quadratic 0-1 programming instances of medium size Quadratic 0-1 Programming problems. Technical Report September, 2007. URL `http://biqmac.uni-klu.ac.at/biqmaclib.html`.

[56] M. Yamashita, K. Fujisawa, K. Nakata, M. Nakata, M. Fukuda, K. Kobayashi, and K. Goto. A high-performance software package for semidefinite programs: SDPA 7. Technical report, 2010. `http://www.optimization-online.org/DB_HTML/2010/01/2531.html`.

[57] H. Ziegler. Solving certain singly constrained convex optimization problems in production planning. *Operations Research Letters*, 1(6):246–252, 1982.

Appendix C

[Friberg 81]
Solving conic optimization problems via self-dual embedding and facial reduction: a unified approach

Solving conic optimization problems via self-dual embedding and facial reduction: a unified approach

Frank Permenter [*] Henrik A. Friberg [§] Erling D. Andersen [¶]

June 20, 2016

Abstract

We establish connections between the facial reduction algorithm of Borwein and Wolkowicz and the self-dual homogeneous model of Goldman and Tucker when applied to conic optimization problems. Specifically, we show the self-dual homogeneous model returns facial reduction certificates when it fails to return a primal-dual optimal solution or a certificate of infeasibility. Using this observation, we give algorithms, based on facial reduction, for solving the primal or dual problem that, in principle, always succeed. These algorithms have the appealing property that they only perform facial reduction when it is required, not when it is possible; e.g. if a primal-dual optimal solution exists, it will be found in lieu of a facial reduction certificate even if Slater's condition fails. We interpret this phenomenon geometrically by studying the cone of solutions to the homogeneous model—an interesting object in its own right. For the case of semidefinite programming, we show our method can be implemented using existing central-path-following techniques.

1 Introduction

For $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$, consider the following primal-dual pair of conic optimization problems over the non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^n$ and its dual cone $\mathcal{K}^* \subseteq \mathbb{R}^n$:

$$\begin{array}{llcll} \text{minimize} & c^T x & \qquad & \text{maximize} & b^T y \\ \text{subject to} & Ax = b & & \text{subject to} & c - A^T y = s \\ & x \in \mathcal{K} & & & s \in \mathcal{K}^*,\ y \in \mathbb{R}^m, \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$ is the decision variable of the primal problem and $(s, y) \in \mathbb{R}^n \times \mathbb{R}^m$ is the decision variable of the dual. The *self-dual embedding* technique, originally due to Goldman and Tucker [8] and generalized in [12, 7, 13, 18], solves (1) by finding solutions to the following *self-dual homogeneous model*:

$$\begin{array}{rcl} Ax - b\tau &=& 0, \\ -A^T y - s + c\tau &=& 0, \\ b^T y - c^T x - \kappa &=& 0, \\ (x, s, y, \tau, \kappa) \in \mathcal{K} \times \mathcal{K}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+, && \end{array} \tag{2}$$

[*]F. Permenter is with the Laboratory for Information and Decision Systems, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge MA 02139, USA. Email: fperment@mit.edu.

[§]H. A. Friberg is with the Department of Wind Energy, Technical University of Denmark, DK-4000 Roskilde, Denmark, and employed at MOSEK ApS. E-mail: metronware@gmail.com.

[¶]E. D. Andersen is the founder and CEO of MOSEK ApS, DK-2100 Copenhagen O, Denmark. E-mail: e.d.andersen@mosek.com.

where any solution (x, s, y, τ, κ) of (2) satisfies the complementarity condition $\tau\kappa = 0$—an easy consequence of weak duality. If $\tau > 0$, then $\frac{1}{\tau}(x, s, y)$ is a primal-dual optimal solution for (1) with duality gap $\frac{1}{\tau^2}x^T s$ equal to zero; in other words, $\frac{1}{\tau}(x, s, y)$ is a *complementary solution* for (1). If $\kappa > 0$, then x and/or y are *improving rays* that certify dual and/or primal infeasibility. If $\tau = \kappa = 0$ holds for all solutions, then at least one problem—primal or dual—is ill-posed in the sense of Renegar [21], and no point (x, s, y, τ, κ) yields an improving ray or a complementary solution[1]. In addition, at least one problem—primal or dual—fails the Slater constraint qualification.

In this paper, we reexamine this latter case. Specifically, we show relative interior solutions of (2) yield *facial reduction certificates* for (1) when $\tau = \kappa = 0$ holds for all solutions. (Restricting to the relative interior is inspired by the analysis of de Klerk et al. [7].) As we review, these certificates allow one to regularize the primal or dual in the manner proposed by Borwein and Wolkowicz [5] and resolve (2), repeating until a complementary solution or an improving ray is obtained. As we show, this idea leads to simple algorithms that solve *arbitrary* instances of the primal or dual, where solve means to find a finite certificate for the optimal value, and a solution if one exists. These algorithms, of course, rely on a subroutine that produces relative interior solutions to (2). As we show, such solutions are obtained from relative interior solutions to the *extended-embedding* of Ye et al. [26], a strictly-feasible conic optimization problem with strictly-feasible dual. If the extended-embedding is a semidefinite program (SDP), tracking its central path with an interior-point method [14] produces one of its relative interior solutions (by strict feasibility and results of [9]). Hence, implementations of our algorithms that solve arbitrary SDPs are conceptually simple, involving only basic linear algebra (for regularization) and repeated calls to an interior-point method. Tracking the central paths of extended-embeddings with sufficient accuracy, however, is likely a difficult numerical task and is a topic we do not address here.

This paper also contributes to the facial reduction literature in a few ways. To explain these contributions, we first review prior work. Executing the facial reduction algorithm of Borwein and Wolkowicz [5], or the simplified versions of Pataki [15] and Waki and Muramatsu [25], requires one obtain facial reduction certificates, which themselves are solutions to conic optimization problems (so-called auxiliary problems). The recent papers of Cheung et al. [6] and Lourenço et al. [11] propose methods for finding certificates, addressing issues of numerical robustness [6] and strict feasibility of auxiliary problems [6, 11], whereas papers of the first author find certificates using conservative approximations [16, 17]. In this paper, we show it is possible to find certificates only when they are *needed* (complementary solutions and improving rays do not exist) as opposed to when they *exist* (e.g., Slater's condition fails for feasible problems). This is done by finding relative interior solutions to (2) by solving, for instance, strictly-feasible extended-embeddings. In contrast, the methods of [5, 15, 25, 6, 11] find a complete set of certificates for feasible problems and regularize until Slater's condition holds (which can be costly and unnecessary), and the approximation-based methods of [16, 17] may fail to find needed certificates. We also show solutions to (2) automatically identify which problem—primal or dual—needs regularization. In contrast, facial reduction procedures often only regularize the problem one is interested in solving. This is insufficient, for instance, to solve the primal problem if it has a finite unattained optimal value; in this case, dual regularization, or equivalently, regularization in the sense of Abrams [1], is required. As we will show, relative interior solutions to (2) always provide the necessary certificate for the required regularization. Indeed, the certificates provided by (2) allow one to handle all pathologies (duality gaps, unattainment, etc.) in a unified facial-reduction-based framework, where, in contrast, the method of [11] for SDP uses a combination of techniques.

[1]If all solutions to (2) satisfy $\tau = \kappa = 0$, the asymptotic behavior of central-path-following techniques may reveal additional information about (1). See Luo et al. [12] and de Klerk et al. [7].

Case	Interpretation
$\tau > 0, \kappa = 0$	Complementary solution
$\tau = 0, \kappa > 0$	Improving ray(s)
$\tau = 0, \kappa = 0$	Facial reduction certificate(s)

Table 1: Interpretation of a relative interior solution to the self-dual homogeneous model (2) in terms of the conic optimization problem (1). The main observation of this paper, given by Corollary 1, is summarized by the last row.

This paper is organized as follows. Section 2 introduces notation and reviews facial reduction. Section 3 introduces the set $\mathbf{H}(\mathcal{C})$—defined as the cone of solutions to the homogeneous model (2) if $\mathcal{K}$ is replaced with some specified non-empty, closed, convex cone $\mathcal{C}$. This section then characterizes the relative interior of $\mathbf{H}(\mathcal{C})$, yielding the results reported in Table 1. The relative boundary of $\mathbf{H}(\mathcal{C})$ is also studied, yielding interesting geometric interpretations of facial reduction certificates. We then show how to find points in the relative interior of $\mathbf{H}(\mathcal{C})$ via extended-embeddings. Section 4 gives algorithms for solving the primal or dual problem using relative interior points in a sequence of cones $\mathbf{H}(\mathcal{C}_i)$, where $\mathcal{C}_i$ is computed from $\mathcal{C}_{i-1}$ using facial reduction. Section 5 contains simple illustrative examples.

2 Background on facial reduction

This section reviews the basic concepts underlying facial reduction algorithms [5, 15, 25]. These algorithms take as input either the primal problem or the dual problem of (1) and replace $\mathcal{K}$ or $\mathcal{K}^*$ with a face containing the primal feasible set or the set of dual feasible slacks. After this replacement, the new problem has optimal value equal to that of the input problem. Moreover, if the input problem is feasible, the new problem satisfies Slater's condition, and if the input problem is infeasible, the new problem is *strongly* infeasible, meaning a strictly-separating hyperplane exists proving infeasibility. For this reason, this replacement is called *regularization.*

These algorithms work by finding a finite sequence of hyperplanes that provably contain the feasible set, where the normal vectors of these hyperplanes are called *facial reduction certificates.* They terminate when facial reduction certificates no longer exist. The number of steps taken by these algorithms depends on the facial reduction certificates used. The minimum number of steps taken is called the *singularity degree* [24] and is intrinsic to the input problem. To explain these ideas in detail, we first review properties of non-empty, closed, convex cones and their faces.

2.1 Cones and faces

A subset $\mathcal{C}$ of $\mathbb{R}^n$ is called a *cone* if it is closed under positive scaling, i.e. $\lambda x \in \mathcal{C}$ for any $\lambda > 0$ when $x \in \mathcal{C}$. A *convex cone* is a cone that is convex. The *dual cone* $\mathcal{C}^*$ of any subset $\mathcal{C}$ of $\mathbb{R}^n$ is the convex cone $\{z \in \mathbb{R}^n : z^T x \geq 0, \ \forall x \in \mathcal{C}\}$. In this paper, we are only concerned with convex cones that are also *closed* and *non-empty.* Note if $\mathcal{C}$ is a non-empty, closed, convex cone, then $\mathcal{C}$ contains the origin and $\mathcal{C}^{**} = \mathcal{C}$.

Let $\mathcal{C}$ be a non-empty, closed, convex cone. A *face* $\mathcal{F}$ of $\mathcal{C}$ is a closed convex subset for which $a, b \in \mathcal{C}$ and $\frac{a+b}{2} \in \mathcal{F}$ implies $a, b \in \mathcal{F}$. A face $\mathcal{F}$ of $\mathcal{C}$ is called a *proper face* if it is non-empty and not equal to $\mathcal{C}$. (Note this definition includes $\mathcal{C} \cap (-\mathcal{C})$ as a proper face, which some authors exclude.) Let $z^\perp$ denote the hyperplane $\{x \in \mathbb{R}^n : x^T z = 0\}$. For $z \in \mathcal{C}^*$, the set $\mathcal{C} \cap z^\perp$ is easily seen to be non-empty face, said to be *exposed* by z. It thus holds $\mathcal{C} \cap z^\perp$ is a proper face if and only if $\mathcal{C} \cap z^\perp \subseteq \mathcal{C}$ holds strictly. Hence, the set $\mathcal{C} \cap z^\perp$ is a proper face if an only if $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$, where $\mathcal{C}^* \setminus \mathcal{C}^\perp$ is the subset of $\mathcal{C}^*$ not contained in the orthogonal complement of the span of $\mathcal{C}$. Since the dual cone $\mathcal{C}^*$ and the proper faces of $\mathcal{C}$ are also nonempty, closed, convex cones, all of these concepts translate if $\mathcal{C}$ is replaced with a proper face $\mathcal{F}$, with the dual cone $\mathcal{C}^*$, or with one of the proper faces of the dual cone. For instance, $z \in \mathcal{F}^* \setminus \mathcal{F}^\perp$ exposes a proper face $\mathcal{F} \cap z^\perp$ of $\mathcal{F}$, just as $z \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp$ exposes a proper face $\mathcal{C}^* \cap z^\perp$ of $\mathcal{C}^*$.

2.2 Primal and dual problems

To explain facial reduction, it is convenient to define primal and dual problems parametrized by an arbitrary non-empty, closed, convex cone $\mathcal{C}$:

Definition 1. *For a non-empty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$, and the problem data $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ of the conic optimization problem* (1)*, let $\mathbf{P}(\mathcal{C})$ denote the primal optimization problem*

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \in \mathcal{C}, \end{array}$$

and let $\mathbf{D}(\mathcal{C})$ denote its dual optimization problem

$$\begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & c - A^T y = s \\ & s \in \mathcal{C}^*,\ y \in \mathbb{R}^m. \end{array}$$

Note $\mathbf{P}(\mathcal{K})$ and $\mathbf{D}(\mathcal{K})$ denote the primal and dual of (1), respectively. We will call $x \in \mathcal{C}$ an improving ray for $\mathbf{P}(\mathcal{C})$ if $Ax = 0$ and $c^T x < 0$. Similarly, we call $(-A^T y, y) \in \mathcal{C}^* \times \mathbb{R}^m$ an improving ray for $\mathbf{D}(\mathcal{C})$ if $y^T b > 0$. A *complementary solution* $(x, s, y) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ for $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$ is a primal-dual optimal solution with zero duality gap, i.e. it consists of a primal feasible point x and dual feasible point (s, y) for which $c^T x = b^T y$, or equivalently, $x^T s = 0$.

We also say $\mathbf{P}(\mathcal{C})$ satisfies Slater's condition if there exists a feasible $x \in \operatorname{relint} \mathcal{C}$. Similar, $\mathbf{D}(\mathcal{C})$ satisfies Slater's condition if there is a feasible $s \in \operatorname{relint} \mathcal{C}^*$. Such x and s are called *Slater points*. Finally, we call $\inf \{c^T x : Ax = b, x \in \mathcal{C}\}$ the *optimal value of* $\mathbf{P}(\mathcal{C})$ and $\sup \{b^T y : c - A^T y \in \mathcal{C}^*, y \in \mathbb{R}^m\}$ the *optimal value of* $\mathbf{D}(\mathcal{C})$.

2.3 Facial reduction certificates

A facial reduction certificate is the normal vector to a particular type of hyperplane. It is defined in terms of a cone $\mathcal{C} \subseteq \mathbb{R}^n$ and either the primal problem $\mathbf{P}(\mathcal{C})$ or dual problem $\mathbf{D}(\mathcal{C})$.

Definition 2. *For a non-empty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$, and the problem data $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ of* (1)*, define facial reduction certificates as follows:*

- *Call $s \in \mathcal{C}^*$ a* facial reduction certificate for $\mathbf{P}(\mathcal{C})$ *if the hyperplane $s^\perp$ contains the affine set $\{x \in \mathbb{R}^n : Ax = b\}$ and $\mathcal{C} \cap s^\perp \subseteq \mathcal{C}$ holds strictly.*
- *Call $x \in \mathcal{C}$ a* facial reduction certificate for $\mathbf{D}(\mathcal{C})$ *if the hyperplane $x^\perp$ contains the affine set $\{c - A^T y : y \in \mathbb{R}^m\}$ and $\mathcal{C}^* \cap x^\perp \subseteq \mathcal{C}^*$ holds strictly.*

A facial reduction certificate for $\mathbf{P}(\mathcal{C})$ exposes a proper face of $\mathcal{C}$ containing the feasible set of $\mathbf{P}(\mathcal{C})$. Similarly, a facial reduction certificate for $\mathbf{D}(\mathcal{C})$ exposes a proper face of $\mathcal{C}^*$ containing the set of dual feasible slacks for $\mathbf{D}(\mathcal{C})$. Hence, existence of these certificates imply failure of Slater's condition for $\mathbf{P}(\mathcal{C})$ or $\mathbf{D}(\mathcal{C})$. For feasible problems, the converse is also true: a facial reduction certificate exists if Slater's condition fails. Specifically, the following is well-known.

Proposition 1. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. The following statements hold.*

- *Suppose the primal problem $\mathbf{P}(\mathcal{C})$ is feasible. A facial reduction certificate for $\mathbf{P}(\mathcal{C})$ exists if and only if the set of Slater points* $\operatorname{relint} \mathcal{C} \cap \{x \in \mathbb{R}^n : Ax = b\}$ *is empty.*
- *Suppose the dual problem $\mathbf{D}(\mathcal{C})$ is feasible. A facial reduction certificate for $\mathbf{D}(\mathcal{C})$ exists if and only if the set of Slater points* $\operatorname{relint} \mathcal{C}^* \cap \{c - A^T y : y \in \mathbb{R}^m\}$ *is empty.*

(See, e.g., Lemma 2 of [17] for a proof, and Theorem 7.1 of [5], Lemma 12.6 of [6], or Lemma 1 of [15] for closely-related statements.)

Facial reduction certificates are also solutions to conic feasibility problems (so-called auxiliary problems). Indeed, a hyperplane contains a non-empty affine set if and only if it has a normal vector satisfying certain linear equations. Hence, the set of facial reduction certificates for $\mathbf{P}(\mathcal{C})$ or $\mathbf{D}(\mathcal{C})$ is defined by particular linear and conic constraints:

Proposition 2. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. The following statements hold.*

- *$s \in \mathbb{R}^n$ is a facial reduction certificate for the primal problem $\mathbf{P}(\mathcal{C})$ if there exists $y \in \mathbb{R}^m$ for which*
$$b^T y = 0, \qquad s = -A^T y, \qquad s \in \mathcal{C}^* \setminus \mathcal{C}^\perp,$$
and all facial reduction certificates are of this form if $\{x \in \mathbb{R}^n : Ax = b\}$ is non-empty.
- *$x \in \mathbb{R}^n$ is a facial reduction certificate for the dual problem $\mathbf{D}(\mathcal{C})$ if*
$$c^T x = 0, \qquad Ax = 0, \qquad x \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp,$$
and all facial reduction certificates are of this form (since $\{c - A^T y : y \in \mathbb{R}^m\}$ is non-empty).

Note the constraint $x \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp$ is satisfied if and only if $x \in \mathcal{C}$ and x has non-zero inner-product with any point in $\operatorname{relint} \mathcal{C}^*$, and similarly for $s \in \mathcal{C}^* \setminus \mathcal{C}^\perp$.

Optimal facial reduction certificates. Faces exposed by facial reduction certificates are partially-ordered by set inclusion. Thus, there is a natural notion of optimality for certificates. Formally:

Definition 3. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. Let $Z_p \subseteq \mathbb{R}^n$ denote the set of facial reduction certificates for $\mathbf{P}(\mathcal{C})$ and $Z_d \subseteq \mathbb{R}^n$ denote the set of facial reduction certificates for $\mathbf{D}(\mathcal{C})$.*

- *$s \in Z_p$ is an* optimal facial reduction certificate *for $\mathbf{P}(\mathcal{C})$ if $\mathcal{C} \cap s^\perp$ satisfies*
$$\mathcal{C} \cap s^\perp \subseteq \mathcal{C} \cap \hat{s}^\perp \quad \textit{for all } \hat{s} \in Z_p.$$
- *$x \in Z_d$ is an* optimal facial reduction certificate *for $\mathbf{D}(\mathcal{C})$ if $\mathcal{C}^* \cap x^\perp$ satisfies*
$$\mathcal{C}^* \cap x^\perp \subseteq \mathcal{C}^* \cap \hat{x}^\perp \quad \textit{for all } \hat{x} \in Z_d.$$

If facial reduction certificates exist, then so do optimal ones. Since facial reduction certificates are closed under addition, this follows easily from the following identities

$$\mathcal{C} \cap (s_1 + s_2)^\perp = \mathcal{C} \cap s_1^\perp \cap s_2^\perp, \qquad \mathcal{C}^* \cap (x_1 + x_2)^\perp = \mathcal{C}^* \cap x_1^\perp \cap x_2^\perp,$$

which hold for any $s_1, s_2 \in \mathcal{C}^*$ and $x_1, x_2 \in \mathcal{C}$. Hence, the sum of a maximal set of linearly independent certificates for $\mathbf{P}(\mathcal{C})$ is an optimal certificate for $\mathbf{P}(\mathcal{C})$, and similarly for $\mathbf{D}(\mathcal{C})$.

2.4 Facial reduction algorithms

A facial reduction algorithm regularizes the primal problem $\mathbf{P}(\mathcal{K})$ of (1) by finding a sequence of facial reduction certificates for $\mathbf{P}(\mathcal{F}_i)$, where each $\mathcal{F}_i$ is a face of $\mathcal{K}$. Similarly, it regularizes the dual problem $\mathbf{D}(\mathcal{K})$ using a sequence of facial reduction certificates for dual problems defined by faces of $\mathcal{K}^*$. We explain the basic idea using the primal problem, and then summarize how it extends to the dual. Additional details can be found in [5, 15, 25].

Facial reduction of the primal problem. Suppose we had a facial reduction certificate $z \in \mathcal{K}^*$ for $\mathbf{P}(\mathcal{K})$. Replacing $\mathcal{K}$ with the face $\mathcal{K} \cap z^\perp$ yields a new primal-dual pair

$$\begin{array}{llll} \text{minimize} & c^T x & \text{maximize} & b^T y \\ \text{subject to} & Ax = b & \text{subject to} & c - A^T y = s \\ & x \in \mathcal{K} \cap z^\perp & & s \in (\mathcal{K} \cap z^\perp)^*, \ y \in \mathbb{R}^m, \end{array} \tag{3}$$

where the primal problem $\mathbf{P}(\mathcal{K} \cap z^\perp)$ and the original $\mathbf{P}(\mathcal{K})$ have the same feasible set and equal optimal values (since, by Definition 2, the hyperplane $z^\perp$ contains all solutions to $Ax = b$). We can, of course, repeat this process. For an integer $d_P > 0$, consider the recursion

$$\mathcal{F}_0 = \mathcal{K}, \quad \mathcal{F}_i = \mathcal{F}_{i-1} \cap z_i^\perp \quad i \in \{1, \ldots, d_P\},$$

where $z_i \in \mathcal{F}_{i-1}^*$ is a facial reduction certificate for $\mathbf{P}(\mathcal{F}_{i-1})$. If $\mathcal{C}$ is replaced with any face $\mathcal{F}_i$ in this recursion, the primal problem $\mathbf{P}(\mathcal{F}_i)$ and the original $\mathbf{P}(\mathcal{K})$ also have the same feasible set and equal optimal values, given that each hyperplane $z_i^\perp$ contains all solutions to $Ax = b$. We call replacement of $\mathcal{K}$ with one of these faces *primal regularization.*

Facial reduction algorithms compute the recursion $\mathcal{F}_i = \mathcal{F}_{i-1} \cap z_i^\perp$ and terminate when a facial reduction certificate for $\mathbf{P}(\mathcal{F}_i)$ does not exist. If optimal certificates z_i are used, the length d_P of the sequence $\mathcal{F}_0, \ldots, \mathcal{F}_{d_P}$ is unique and does not depend on the specific certificates z_i. This length is called the *singularity degree* of $\mathbf{P}(\mathcal{K})$. If $\mathbf{P}(\mathcal{K})$ is feasible, the last face $\mathcal{F}_{d_P}$ in the sequence is called the *minimal face* of $\mathbf{P}(\mathcal{K})$. One can show if $\mathbf{P}(\mathcal{K})$ is feasible, the regularized problem $\mathbf{P}(\mathcal{F}_{d_P})$ satisfies Slater's condition, and if $\mathbf{P}(\mathcal{K})$ is infeasible, the dual $\mathbf{D}(\mathcal{F}_{d_P})$ of the regularized primal problem has an improving ray.

Facial reduction of the dual problem. The dual problem $\mathbf{D}(\mathcal{K})$ of (1) is regularized in a similar way. Given a facial reduction certificate $z \in \mathcal{K}$ for $\mathbf{D}(\mathcal{K})$, one can reformulate (1) as:

$$\begin{array}{llll} \text{minimize} & c^T x & \text{maximize} & b^T y \\ \text{subject to} & Ax = b & \text{subject to} & c - A^T y = s \\ & x \in (\mathcal{K}^* \cap z^\perp)^* & & s \in \mathcal{K}^* \cap z^\perp, \ y \in \mathbb{R}^m, \end{array} \tag{4}$$

where $\mathcal{K}^*$ has been replaced with the face $\mathcal{K}^* \cap z^\perp$. Since the hyperplane $z^\perp$ contains all vectors of the form $c - A^T y$, the dual problem $\mathbf{D}\big((\mathcal{K}^* \cap z^\perp)^*\big)$ and the original $\mathbf{D}(\mathcal{K})$ have the same feasible set and equal optimal values.

As with the primal problem, we can repeat this process. Specifically, we can identify a sequence of faces of $\mathcal{K}^*$ via the recursion $\mathcal{F}_i = \mathcal{F}_{i-1} \cap z_i^{\perp}$, where $\mathcal{F}_0 := \mathcal{K}^*$ and $z_i \in \mathcal{F}_{i-1}^*$ is a facial reduction certificate for $\mathbf{D}(\mathcal{F}_{i-1}^*)$, terminating when facial reduction certificates no longer exist. We call replacement of $\mathcal{K}^*$ with one of the faces $\mathcal{F}_i$ *dual regularization*. As with the primal problem, if optimal certificates z_i are used, the length d_D of the sequence $\mathcal{F}_0, \ldots, \mathcal{F}_{d_D}$ is unique and does not depend on the specific certificates z_i. This length is called the *singularity degree* of $\mathbf{D}(\mathcal{K})$. Similarly, the last face $\mathcal{F}_{d_D}$ in the sequence is called the minimal face of $\mathbf{D}(\mathcal{K})$ when $\mathbf{D}(\mathcal{K})$ is feasible. One can show if $\mathbf{D}(\mathcal{K})$ is feasible, the regularized problem $\mathbf{D}(\mathcal{F}_{d_D}^*)$ satisfies Slater's condition, and if $\mathbf{D}(\mathcal{K})$ is infeasible, the primal problem $\mathbf{P}(\mathcal{F}_{d_D}^*)$ has an improving ray.

2.5 Primal-dual facial reduction asymmetry

While a facial reduction algorithm leaves the feasible set of the input problem unchanged, the same is not true for the corresponding Lagrangian dual problem—in other words, facial reduction is asymmetric with respect to duality. Compare the primal-dual pair (1) with the primal-regularized pair (3). While the *primal* feasible sets are the same, the *dual* feasible set of (3) is potentially larger. An analogous statement holds when comparing (1) with the dual-regularized pair (4); while the *dual* feasible sets are the same, the *primal* feasible set of (4) is potentially larger. Hence, by solving (3) or (4), one won't (generally) find solutions to both the primal $\mathbf{P}(\mathcal{K})$ and dual $\mathbf{D}(\mathcal{K})$ of (1).

Of course, this should not always be viewed as a negative "side-effect" of facial reduction; enlarging the primal or dual feasible set may be necessary to remove duality gaps and find improving rays, which, in a sense, is the entire point of the technique. Nevertheless, a primal-dual solver that also performs facial reduction needs more than just the problem data (A, b, c); it must also know which problem—primal or dual—is of actual interest.

2.6 Connections with our approach

In this paper, we show optimal facial reduction certificates are obtained from the homogeneous model when complementary solutions or improving rays do not exist. This allows us to perform primal or dual regularization and resolve the homogeneous model, repeating until a complementary solution or improving ray is obtained. In addition, complementary solutions and improving rays will always be obtained when they exist, even if facial reduction certificates exist as well. As a consequence, we can find complementary solutions without having to first identify the minimal face—i.e., we do not have to regularize until Slater's condition holds. Towards making these statements precise, we now study homogeneous models in more detail.

3 Solutions to homogeneous models

In this section, we examine the solution sets of homogeneous models and present our main theoretical results. The main object of interest is the convex cone $\mathbf{H}(\mathcal{C})$, defined as follows:

Definition 4. *For a non-empty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$, and the problem data $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ of* (1)*, define $\mathbf{H}(\mathcal{C})$ as the convex cone of solutions (x, s, y, τ, κ) to the system:*

$$\begin{array}{rcl} Ax - b\tau & = & 0, \\ -A^T y - s + c\tau & = & 0, \\ b^T y - c^T x - \kappa & = & 0, \\ \multicolumn{3}{c}{(x, s, y, \tau, \kappa) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+.} \end{array}$$

Note if $\mathcal{C} = \mathcal{K}$, then $\mathbf{H}(\mathcal{C})$ equals the solution set of the homogeneous model (2). (Allowing $\mathcal{C}$ to differ from $\mathcal{K}$ is convenient for stating algorithms in the next section.)

We will study both the relative interior and relative boundary of $\mathbf{H}(\mathcal{C})$. Our study of the relative interior is inspired by [7], which considers 'maximally-complementary' solutions of self-dual embeddings for semidefinite programs. Our main result (Theorem 1) classifies the relative interior of $\mathbf{H}(\mathcal{C})$ and implies results of Table 1 as a corollary—specifically, it implies all relative interior solutions of $\mathbf{H}(\mathcal{C})$ yield complementary solutions, improving rays or optimal facial reduction certificates. Our study of the relative boundary yields a geometric interpretation of the non-necessity of Slater's condition: in most cases, facial reduction certificates correspond only to relative boundary points when complementary solutions or improving rays exist. We also show all sub-optimal facial reduction certificates are contained in the relative boundary. Finally, we show relative interior solutions to extended-embeddings, which are found by central-path-following techniques in the case of semidefinite programming, yield points in $\operatorname{relint} \mathbf{H}(\mathcal{C})$.

3.1 The relative interior

The following theorem classifies $(x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{C})$ by the values of τ and κ. A corollary follows restating key statements in terms of complementary solutions, improving rays and facial reduction certificates for the primal-dual pair given by $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$.

Theorem 1. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. For $(x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{C})$, the following statements hold:*

1. *If $\tau > 0$, then $\frac{1}{\tau}(Ax) = b$ and $\frac{1}{\tau}(A^T y + s) = c$, and $b^T y = c^T x$.*
2. *If $\kappa > 0$, then $Ax = 0$, $A^T y + s = 0$ and $b^T y > c^T x$.*
3. *If $\tau = \kappa = 0$, then τ and κ vanish for all points in $\mathbf{H}(\mathcal{C})$. In addition, letting $\mathcal{F}_p := \mathcal{C} \cap s^\perp$, $\mathcal{F}_d := \mathcal{C}^* \cap x^\perp$, $\mathcal{A}_p := \{x \in \mathbb{R}^n : Ax = b\}$ and $\mathcal{A}_d := \{c - A^T y : y \in \mathbb{R}^m\}$,*
   - *(a) The hyperplane $s^\perp$ contains $\mathcal{A}_p$;*
   - *(b) The hyperplane $x^\perp$ contains $\mathcal{A}_d$;*
   - *(c) The face $\mathcal{F}_p$ is proper if and only if $\operatorname{relint} \mathcal{C} \cap \mathcal{A}_p$ is empty;*
   - *(d) The face $\mathcal{F}_d$ is proper if and only if $\operatorname{relint} \mathcal{C}^* \cap \mathcal{A}_d$ is empty;*
   - *(e) At least one of the faces $\mathcal{F}_p$ or $\mathcal{F}_d$ is proper;*
   - *(f) The inclusion $\mathcal{F}_p \subseteq \mathcal{C} \cap \hat{s}^\perp$ holds for all $\hat{s} \in \mathcal{C}^*$ satisfying $\mathcal{A}_p \subseteq \hat{s}^\perp$;*
   - *(g) The inclusion $\mathcal{F}_d \subseteq \mathcal{C}^* \cap \hat{x}^\perp$ holds for all $\hat{x} \in \mathcal{C}$ satisfying $\mathcal{A}_d \subseteq \hat{x}^\perp$.*

Proof. Statements one and two are immediate by showing $\tau\kappa = 0$ for any solution (x, s, y, τ, κ) in $\mathbf{H}(\mathcal{C})$. In particular, it holds that

$$0 \leq x^T s = x^T(c\tau - A^T y) = \tau(c^T x - b^T y) = -\tau\kappa \leq 0.$$

Hence, if $\tau > 0$, then $\kappa = 0$, showing the first statement. If $\kappa > 0$, then $\tau = 0$, showing the second statement.

We now prove the third statement where $\tau = \kappa = 0$. We let $w := (x, s, y, \tau, \kappa)$. To begin, since $w \in \operatorname{relint} \mathbf{H}(\mathcal{C})$, there can be no point $\hat{w} := (\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa}) \in \mathbf{H}(\mathcal{C})$ with $\hat{\kappa} \neq 0$ or $\hat{\tau} \neq 0$. Otherwise, we'd have $w - \alpha\hat{w} \notin \mathbf{H}(\mathcal{C})$ for every $\alpha > 0$, contradicting the fact $w \in \operatorname{relint} \mathbf{H}(\mathcal{C})$. This shows the first part of the third statement. This also implies $c^T x = b^T y = 0$ when $\tau = \kappa = 0$. To see this, note if $w := (x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{C})$ for $\tau = \kappa = 0$, then $c^T x = b^T y$. Since at least one of the points—$(0, s, y, 0, b^T y)$ or $(x, 0, 0, 0, -c^T x)$—is in $\mathbf{H}(\mathcal{C})$, it must hold that $c^T x = b^T y = 0$. We now use this fact to show statements (3a)-(3b).

To see that statements (3a)-(3b) hold, note that $\hat{x}^T s = -\hat{x}^T A^T y = -b^T y = 0$ for all solutions $\hat{x}$ of $Ax = b$. Hence, the solution set $\{x \in \mathbb{R}^n : Ax = b\}$ is contained in the hyperplane $s^\perp$. Likewise, $x^T \hat{s} = c^T x = 0$ for all $\hat{s} \in \{c - A^T y : y \in \mathbb{R}^m\}$, hence $x^\perp$ contains $\{c - A^T y : y \in \mathbb{R}^m\}$.

We now show (3d). One direction is trivial; if $\mathcal{A}_d$ is contained in a proper face of $\mathcal{C}^*$, then $\operatorname{relint} \mathcal{C}^* \cap \mathcal{A}_d$ must be empty. For the converse direction, suppose $\operatorname{relint} \mathcal{C}^* \cap \{c - A^T y : y \in \mathbb{R}^m\}$ is empty. The main separation theorem (Theorem 11.3) of [22] states that a hyperplane exists *properly* separating these sets. Using Theorem 11.7 of [22], we can additionally assume this hyperplane passes through the origin since $\mathcal{C}^*$ is a cone. In other words, there exists $\hat{x} \in \mathcal{C}$, satisfying $\hat{x}^T z \geq 0$ for all $z \in \mathcal{C}^*$ (by definition), for which

$$\hat{x}^T (c - A^T y) \leq 0, \quad \forall y \in \mathbb{R}^m,$$
$$\hat{x}^T z \neq 0 \text{ for some } z \in \{c - A^T y : y \in \mathbb{R}^m\} \cup \mathcal{C}^*.$$

It follows that $c^T \hat{x} \leq y^T A \hat{x}$ for arbitrary $y \in \mathbb{R}^m$, which implies $A\hat{x} = 0$ and $c^T \hat{x} \leq 0$. But $c^T \hat{x} = 0$, otherwise $\hat{x}$ is an improving ray for $\mathbf{P}(C)$, and $(\hat{x}, 0, 0, 0, -c^T \hat{x}) \in \mathbf{H}(\mathcal{C})$ with $\kappa > 0$. Hence, the hyperplane $\hat{x}^\perp$ contains $\{c - A^T y : y \in \mathbb{R}^m\}$ implying $\hat{x}^T z \neq 0$ for some $z \in \mathcal{C}^*$ given proper separation of the sets. That is, $\hat{x}$ exposes a proper face of $\mathcal{C}^*$, but we have yet to show that x exposes a proper face of $\mathcal{C}^*$ as claimed. Clearly, $\hat{w} := (\hat{x}, 0, 0, 0, 0) \in \mathbf{H}(\mathcal{C})$. Since w is in the relative interior of $\mathbf{H}(\mathcal{C})$, it holds that $w \pm \alpha \hat{w} \in \mathbf{H}(\mathcal{C})$, and thus $x \pm \alpha \hat{x} \in \mathcal{C}$, for some $\alpha > 0$. Hence, for any $u \in \mathcal{C}^*$, the inequality $u^T(x \pm \alpha \hat{x}) \geq 0$ holds, which in turn implies $u^T \hat{x} = 0$ when $u^T x = 0$. In other words, $\mathcal{C}^* \cap x^\perp$ is contained in a proper face, i.e.,

$$\mathcal{C}^* \cap x^\perp \subseteq \mathcal{C}^* \cap \hat{x}^\perp, \tag{5}$$

and is hence proper.

Applying the argument of the previous paragraph to the set $\operatorname{relint} \mathcal{C} \cap \mathcal{A}_p$ shows (3c).

Statement (3e) follows if at least one of the sets—$\operatorname{relint} \mathcal{C} \cap \{x \in \mathbb{R}^n : Ax = b\}$ or $\operatorname{relint} \mathcal{C}^* \cap \{c - A^T y : y \in \mathbb{R}^m\}$—is empty. Suppose this weren't the case. Then, Slater's condition is satisfied for both $\mathbf{P}(C)$ and $\mathbf{D}(C)$ showing existence of an optimal primal-dual solution with zero duality gap (see [4]; Section 7.2.2). Hence, there exists a point in $\mathbf{H}(\mathcal{C})$ with $\tau > 0$, contradicting the assumption that (x, s, y, τ, κ) is in the relative interior of $\mathbf{H}(\mathcal{C})$.

The same argument that shows the containment (5) shows (3f) and (3g). □

The following corollary arises simply from definitions. Variants of the first two statements (and their converses) are well-known and given in [7] for semidefinite programming. The third statement is the main observation of this paper. Note taking $\mathcal{C} = \mathcal{K}$ yields Table 1.

Corollary 1. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone, and suppose $(x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{C})$. The following statements hold for the primal-dual pair $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$.*

1. *If $\tau > 0$, then $\frac{1}{\tau}(x, s, y)$ is a complementary solution for $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$.*
2. *If $\kappa > 0$, then x is an improving ray for $\mathbf{P}(\mathcal{C})$ and/or (s, y) is an improving ray for $\mathbf{D}(\mathcal{C})$.*
3. *If $\tau = \kappa = 0$, then x and/or s are optimal facial reduction certificates in the sense of Definition 3.*

Moreover, converses of the first two statements hold: if $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$ have a complementary solution, then $\tau > 0$; if $\mathbf{P}(\mathcal{C})$ and/or $\mathbf{D}(\mathcal{C})$ have an improving ray, then $\kappa > 0$.

Facial reduction certificates and improving rays for both problems. Statements two and three of Corollary 1 are subtle. In particular, statement three does not guarantee s and x are *both* facial reduction certificates when such certificates exist for *both* primal and dual problems. Similarly, statement two does not guarantee s and x are *both* improving rays when such rays exist for *both* problems. Using (3c) and (3d) of Theorem 1 we can strengthen statement three to make this guarantee for facial reduction certificates:

Corollary 2. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone, and suppose $(x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{C})$ with $\tau = \kappa = 0$. The following statements hold:*

- *If $\operatorname{relint} \mathcal{C} \cap \{x \in \mathbb{R}^n : Ax = b\}$ is empty, then s is an optimal facial reduction certificate for $\mathbf{P}(\mathcal{C})$.*
- *If $\operatorname{relint} \mathcal{C}^* \cap \{c - A^T y : y \in \mathbb{R}^m\}$ is empty, then x is an optimal facial reduction certificate for $\mathbf{D}(\mathcal{C})$.*

Statement two of Corollary 1, on the other hand, cannot be strengthened—in the $\kappa > 0$ case, there are instances for which relative interior points do not yield improving rays for both problems, even if these rays exist. This is a known shortcoming of the homogeneous self-dual model that occurs even in the linear programming case (see, e.g., [26]). The following illustrates this shortcoming:

Example 1. *Consider the following primal-dual pair of linear programs*

$$\begin{array}{ll} \text{minimize} & -x_1 \\ \text{subject to} & x_1 - x_2 = 0 \\ & -(x_1 - x_2) = 1 \\ & x \in \mathbb{R}^2_+ \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \binom{-1}{0} - \binom{y_1 - y_2}{-(y_1 - y_2)} = s \\ & s \in \mathbb{R}^2_+, \end{array}$$

where both the primal and dual problem are infeasible. Indeed, the point $(\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa}) \in \operatorname{relint} \mathbf{H}(\mathbb{R}^2_+)$ yields an improving ray $\hat{x}$ for the primal and an improving ray $\hat{y}$ for the dual, where

$$\hat{x} = (1, 1), \quad \hat{s} = (0, 0), \quad \hat{y} = (1, 1), \quad \hat{\tau} = 0, \quad \hat{\kappa} = 2.$$

Nevertheless, the entire family of points:

$$\tilde{x} = (r, r), \quad \tilde{s} = (0, 0), \quad \tilde{y} = (t, t), \quad \tilde{\tau} = 0, \quad \tilde{\kappa} = r + t, \quad \textit{for } r > -t \geq 0,$$

are also in the relative interior of solutions to the homogeneous model, and only give improving rays for the primal problem.

In Example 1, any point from the family $(\tilde{x}, \tilde{s}, \tilde{y}, \tilde{\tau}, \tilde{\kappa})$ leaves the status of the primal problem unknown; from such point, we can only conclude the primal problem is infeasible or unbounded. To resolve this ambiguity, one can set the objective function $c^T x$ of the primal problem to zero and solve the resulting feasibility problem:

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & x_1 - x_2 = 0 \\ & -(x_1 - x_2) = 1 \\ & x \in \mathbb{R}^2_+ \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \binom{0}{0} - \binom{y_1 - y_2}{-(y_1 - y_2)} = s \\ & s \in \mathbb{R}^2_+. \end{array}$$

Since dual improving rays exists, but there can be no primal improving rays by construction, *all* points in $\operatorname{relint} \mathbf{H}(\mathbb{R}^2_+)$, such as the point

$$\hat{x} = (1, 1), \quad \hat{s} = (0, 0), \quad \hat{y} = (1, 1), \quad \hat{\tau} = 0, \quad \hat{\kappa} = 2,$$

satisfy $\kappa > 0$ and certify primal infeasibility. A general algorithm resolving ambiguity in this way is given in Section 4.1.2.

Finding Slater points. Suppose the primal problem $\mathbf{P}(\mathcal{C})$ is strictly feasible. Now consider the problem of finding a strictly feasible point (i.e., a Slater point) for $\mathbf{P}(\mathcal{C})$. Not surprisingly, such points are found from points in $\operatorname{relint}\mathbf{H}(\mathcal{C})$ if one sets the cost vector $c \in \mathbb{R}^n$ to zero. Similar statements hold for $\mathbf{D}(\mathcal{C})$. Formally:

Theorem 2. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone, let $\mathcal{A}_p := \{x \in \mathbb{R}^n : Ax = b\}$ and let $\mathcal{A}_d := \{c - A^T y : y \in \mathbb{R}^m\}$. For $(x, s, y, \tau, \kappa) \in \operatorname{relint}\mathbf{H}(\mathcal{C})$, the following statements hold.*

1. *If $c = 0$ and $\mathcal{A}_p \cap \operatorname{relint}\mathcal{C}$ is non-empty, then $\tau > 0$ and $\frac{1}{\tau}x \in \mathcal{A}_p \cap \operatorname{relint}\mathcal{C}$.*
2. *If $b = 0$ and $\mathcal{A}_d \cap \operatorname{relint}\mathcal{C}^*$ is non-empty, then $\tau > 0$ and $\frac{1}{\tau}s \in \mathcal{A}_d \cap \operatorname{relint}\mathcal{C}^*$.*

Proof. We only prove statement one, given proof of statement two follows by similar reasoning. To begin, note by assumption there exists $x_0 \in \mathcal{A}_p \cap \operatorname{relint}\mathcal{C}$, where $(x_0, 0, 0) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ is a complementary solution for $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$ given that $c = 0$. By Corollary 1, we conclude $\tau > 0$. Letting $w = (x, s, y, \tau, \kappa)$, we note $\frac{1}{\tau}w \in \operatorname{relint}\mathbf{H}(\mathcal{C})$, as the relative interior of the cone $\mathbf{H}(\mathcal{C})$ is itself a cone (Corollary 6.6.1 of [22]). For some $\alpha > 0$, it thus holds that

$$\frac{1}{\tau}w + \alpha\left(\frac{1}{\tau}w - (x_0, 0, 0, 1, 0)^T\right) \in \mathbf{H}(\mathcal{C}), \quad \text{and hence} \quad \frac{1}{\tau}x + \alpha\left(\frac{1}{\tau}x - x_0\right) \in \mathcal{A}_p \cap \mathcal{C}.$$

Hence, by convexity, $\mathcal{A}_p \cap \mathcal{C}$ contains the line segment from x_0 to $\frac{1}{\tau}x + \alpha\left(\frac{1}{\tau}x - x_0\right)$, where $\frac{1}{\tau}x$ is in the relative interior of this line segment. By Corollary 6.5.1 of [22], we also have that

$$\mathcal{A}_p \cap \operatorname{relint}\mathcal{C} = \operatorname{relint}\left(\mathcal{A}_p \cap \mathcal{C}\right). \tag{6}$$

Hence, $x_0 \in \operatorname{relint}(\mathcal{A}_p \cap \mathcal{C})$, which, by the Line Segment Principle (Proposition 1.3.1 of [3]) implies $\frac{1}{\tau}x \in \operatorname{relint}(\mathcal{A}_p \cap \mathcal{C})$ as well. By (6), we conclude $\frac{1}{\tau}x \in \mathcal{A}_p \cap \operatorname{relint}\mathcal{C}$, as desired.

□

Theorem 2 will be used in Section 4.1.3, which describes generalized certificates for the optimal value of $\mathbf{P}(\mathcal{C})$ that, in some cases, consists of a Slater point.

3.2 The relative boundary

We now study the relative boundary of $\mathbf{H}(\mathcal{C})$. The relative boundary $\partial\mathbf{H}(\mathcal{C})$ is the subset of $\mathbf{H}(\mathcal{C})$ not contained in the relative interior of $\mathbf{H}(\mathcal{C})$; in other words, $\partial\mathbf{H}(\mathcal{C}) := \mathbf{H}(\mathcal{C}) \setminus \operatorname{relint}\mathbf{H}(\mathcal{C})$. This set has interesting interpretations in terms of Slater's condition and facial reduction certificates failing the optimality criterion of Definition 3.

Non-necessity of Slater's condition. Suppose the primal and dual of (1) satisfy Slater's condition, i.e., suppose there exists a primal feasible point in the relative interior of $\mathcal{C}$ and a dual feasible slack in the relative interior of $\mathcal{C}^*$. Standard results from conic duality theory (e.g. [4], Section 7.2.2) imply a complementary solution $(\hat{x}, \hat{y}, \hat{s}) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ exists solving $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$. Slater's condition, of course, is only a *sufficient condition* for existence of $(\hat{x}, \hat{y}, \hat{s})$; in particular, complementary solutions and facial reduction certificates can co-exist. The next proposition describes these certificates geometrically, showing they correspond to relative boundary points of $\mathbf{H}(\mathcal{C})$ in most situations:

Proposition 3. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone and let $(\hat{x}, \hat{s}, \hat{y}) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ be a complementary solution for the primal-dual pair $\mathbf{P}(\mathcal{C})$ and $\mathbf{D}(\mathcal{C})$. The following statements hold, where facial reduction certificates are in the sense of Definition 2:*

1. *If $\mathbf{D}(\mathcal{C})$ fails Slater's condition, there exists $(x, s, y, \tau, \kappa) \in \partial\mathbf{H}(\mathcal{C})$ satisfying $(x, s, y, \tau, \kappa) = (x, 0, 0, 0, 0)$, where $x \in \mathcal{C}$ is a facial reduction certificate for $\mathbf{D}(\mathcal{C})$.*

2. *If $\mathbf{P}(\mathcal{C})$ fails Slater's condition, there exists $(x, s, y, \tau, \kappa) \in \partial\mathbf{H}(\mathcal{C})$ satisfying $(x, s, y, \tau, \kappa) = (0, s, y, 0, 0)$, where $s \in \mathcal{C}^*$ is a facial reduction certificate for $\mathbf{P}(\mathcal{C})$.*

3. *Let $(x, s, y, \tau, \kappa) \in \operatorname{relint}\mathbf{H}(\mathcal{C})$. If x is a facial reduction certificate for $\mathbf{D}(\mathcal{C})$, then x is an optimal solution of $\mathbf{P}(\mathcal{C})$, $b = 0$ and $c^T x = c^T \hat{x} = 0$. If s is a facial reduction certificate for $\mathbf{P}(\mathcal{C})$ and $s = -A^T y$, then (s, y) is an optimal solution of $\mathbf{D}(\mathcal{C})$, $c = 0$ and $b^T y = b^T \hat{y} = 0$.*

Proof. Facial reduction certificates exist in statements one and two because, for feasible problems, existence is a strong alternative to Slater's condition as stated in Proposition 1. That these certificates cannot correspond to points in $\operatorname{relint}\mathbf{H}(\mathcal{C})$ follows because $\tau > 0$ for all points in $\operatorname{relint}\mathbf{H}(\mathcal{C})$ given a complementary solution exists by assumption. If x is a facial reduction certificate, then x is orthogonal to the affine set $\{c - A^T y : y \in \mathbb{R}^m\}$. Hence, $Ax = 0$ and $c^T x = 0$ by Proposition 2. Since $\tau > 0$ and $Ax = \tau b$ holds for solutions of $\mathbf{H}(\mathcal{C})$ by Definition 4, it follows that $b = 0$ which implies x is primal feasible. In addition, $c^T \hat{x} = b^T \hat{y} = 0 = c^T x$, which shows optimality x. Likewise, if $s = -A^T y$ is a facial reduction certificate, then by definition s is orthogonal to all solutions to $Ax = b$. We conclude $\hat{x}^T s = -\hat{x}^T A^T y = -b^T y = 0$. Since $\tau > 0$ and $A^T y + s = \tau c$, we also conclude that $c = 0$, which shows dual feasibility of (s, y). Since $b^T \hat{y} = c^T \hat{x} = 0 = b^T y$, optimality of (s, y) follows.

□

The following example illustrates aspects of Proposition 3.

Example 2. *For the quadratic cone $\mathcal{Q}^3 := \{(x_1, x_2, x_3) : \sum_{i=1}^3 x_i^2 \leq x_1^2\}$, consider the primal-dual pair*

$$\begin{array}{ll} \text{minimize} & 2x_1 + x_2 + x_3 \\ \text{subject to} & x_1 - x_2 = 0 \\ & x \in \mathcal{Q}^3 \end{array} \qquad \begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} y \\ -y \\ 0 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3. \end{array} \tag{7}$$

Here, the primal $\mathbf{P}(\mathcal{Q}^3)$ fails Slater's condition but a complementary solution exists. Indeed, the point $(\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa}) \in \operatorname{relint}\mathbf{H}(\mathcal{Q}^3)$ given by

$$\hat{x} = (0, 0, 0), \quad \hat{s} = (2, 1, 1), \quad \hat{y} = 0, \quad \hat{\tau} = 1, \quad \hat{\kappa} = 0$$

yields a complementary solution $(\hat{x}, \hat{s}, \hat{y})$. On the other hand, the point $(\tilde{x}, \tilde{s}, \tilde{y}, \tilde{\tau}, \tilde{\kappa}) \in \partial\mathbf{H}(\mathcal{Q}^3)$ given by

$$\tilde{x} = (0, 0, 0), \quad \tilde{s} = (1, -1, 0), \quad \tilde{y} = -1, \quad \tilde{\tau} = \tilde{\kappa} = 0$$

yields a facial reduction certificate $\tilde{s}$.

Sub-optimal facial reduction certificates. Definition 3 defined an optimal facial reduction certificate as one exposing a face as small as possible. Here, we consider *sub-optimal* certificates, and find they appear in the relative boundary:

Proposition 4. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. Suppose x is a facial reduction certificate for $\mathbf{D}(\mathcal{C})$ and, for $y \in \mathbb{R}^m$ satisfying $b^T y = 0$, suppose $s := -A^T y$ is a facial reduction certificate for $\mathbf{P}(\mathcal{C})$. If s and x are* not *optimal in the sense of Definition 3, then $(x, s, y, 0, 0) \in \partial\mathbf{H}(\mathcal{C})$.*

Proof. This follows directly from statements (3f)-(3g) of Theorem 1, which imply certificates in the relative interior of $\mathbf{H}(\mathcal{C})$ are optimal. □

3.3 Finding relative interior points via extended-embeddings

We now address a practical question: how can one find points in relint $\mathbf{H}(\mathcal{C})$? To provide an answer, we show points in relint $\mathbf{H}(\mathcal{C})$ are obtained from points in the relative interior of the set of optimal solutions to the *extended-embedding* of Ye et al. [26]:

$$\begin{array}{ll} \text{minimize} & \mu\theta \\ \text{subject to} & \begin{aligned} Ax - b\tau &= r_p\theta \\ -A^T y - s + c\tau &= r_d\theta \\ b^T y - c^T x - \kappa &= r_g\theta \\ r_p^T y + r_d^T x + r_g\tau &= -\mu, \end{aligned} \end{array} \tag{8}$$

where $(x, s, y, \tau, \kappa, \theta) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+ \times \mathbb{R}$ is the decision variable. The parameters are defined by a chosen point $(\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa})$ in the relative interior of $\mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}_+$ via:

$$r_p = A\hat{x} - b\hat{\tau}, \qquad r_d = -A^T\hat{y} - \hat{s} + c\hat{\tau}, \qquad r_g = b^T\hat{y} - c^T\hat{x} - \hat{\kappa}, \qquad \mu = \hat{x}^T\hat{s} + \hat{\tau}\hat{\kappa},$$

which implies $(\hat{x}, \hat{s}, \hat{y}, \hat{\tau}, \hat{\kappa}, 1)$ is a strictly feasible point. The embedding (8) is also self-dual, meaning the dual problem is to maximize $-\mu\bar{\theta}$ over dual variables $(\bar{x}, \bar{s}, \bar{y}, \bar{\tau}, \bar{\kappa}, \bar{\theta})$ satisfying identical constraints. Since a dual optimal point can be identified with a primal optimal point, and since strong duality holds by strict feasibility of both primal and dual problems, we conclude $-\mu\theta = \mu\theta$ at optimality (which implies $\theta = 0$ at optimality.) Hence, by inspection, projecting away the θ coordinate from any optimal solution produces a point in $\mathbf{H}(\mathcal{C})$. This projection also maps relative interior optimal solutions to relative interior points in $\mathbf{H}(\mathcal{C})$. To show this, we first need the following technical result:

Lemma 1. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone and suppose $w := (x, s, y, \tau, \kappa) \in$ relint $\mathbf{H}(\mathcal{C})$. Then, there exists $\beta > 0$ such that $(\beta w, 0)$ is an optimal solution to* (8).

Proof. We only need to verify existence of $\beta > 0$ such that $(\beta w, 0)$ satisfies the last equality constraint. Since $\mu > 0$, it suffices to show $r_d^T x + r_p^T y + r_g\tau < 0$:

$$\begin{aligned} r_p^T y + r_d^T x + r_g\tau &= (A\hat{x} - b\hat{\tau})^T y + (-A^T\hat{y} - \hat{s} + c\hat{\tau})^T x + (b^T\hat{y} - c^T\hat{x} - \hat{\kappa})\tau \\ &= -\hat{x}^T(-A^T y + c\tau) - \hat{s}^T x - \hat{y}^T(Ax - b\tau) - \hat{\tau}(b^T y - c^T x) - \hat{\kappa}\tau \\ &= -\hat{x}^T s - x^T\hat{s} - \hat{\tau}\kappa - \tau\hat{\kappa} \end{aligned}$$

Since $\hat{x}^T s + x^T\hat{s} + \hat{\tau}\kappa + \tau\hat{\kappa}$ is a sum of non-negative numbers, the claim follows if at least one summand is non-zero. Suppose then that $\tau = \kappa = \hat{x}^T s = x^T\hat{s} = 0$. Since $\hat{x} \in$ relint $\mathcal{C}$, we have for all $x_0 \in \mathcal{C}$ a scalar $\zeta > 0$ for which

$$0 \leq (\hat{x} \pm \zeta x_0)^T s = \pm\zeta x_0^T s.$$

Hence, $s \in \mathcal{C}^\perp$. A similar argument shows $x \in (\mathcal{C}^*)^\perp$. By Theorem 1, this cannot hold if $w \in$ relint $\mathbf{H}(\mathcal{C})$, since x or s must expose a proper face when $\tau = \kappa = 0$. $\square$

We now state our result:

Theorem 3. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone and let $\mathbf{\Omega}(\mathcal{C})$ denote the set of optimal solutions to the extended-embedding* (8)*. For any $(x, s, y, \tau, \kappa, \theta) \in$ relint $\mathbf{\Omega}(\mathcal{C})$, the point (x, s, y, τ, κ) is in* relint $\mathbf{H}(\mathcal{C})$.

Proof. If $(w, \theta) \in \mathbf{\Omega}(\mathcal{C})$, then $\theta = 0$ as argued. Hence, by inspection, $\mathbf{\Omega}(\mathcal{C}) = M \cap (\mathbf{H}(\mathcal{C}) \times \{0\})$ for the hyperplane $M = \{(x, s, y, \tau, \kappa, \theta) : r_p^T y + r_d^T x + r_g \tau = -\mu\}$, whereby the claim follows from

$$\operatorname{relint} \mathbf{\Omega}(\mathcal{C}) = M \cap \operatorname{relint}(\mathbf{H}(\mathcal{C}) \times \{0\}) = M \cap (\operatorname{relint} \mathbf{H}(\mathcal{C}) \times \{0\}) \subseteq \operatorname{relint} \mathbf{H}(\mathcal{C}) \times \{0\}.$$

The second equality follows from the definition of relative interior noting that affine hulls are invariant to Cartesian products with the set $\{0\}$. The first equality follows from Corollary 6.5.1 of [22], as $M \cap (\operatorname{relint} \mathbf{H}(\mathcal{C}) \times \{0\})$, and hence $M \cap \operatorname{relint}(\mathbf{H}(\mathcal{C}) \times \{0\})$, is non-empty by Lemma 1. □

Note if the extended-embedding is a semidefinite program, then by strict feasibility and self-duality the central path exists and converges to a point in the relative interior of the solution set [9]. Hence, for SDP, central-path-following techniques can produce a relative interior solution to the extended-embedding and, by Theorem 3, a point in $\operatorname{relint} \mathbf{H}(\mathcal{C})$.

4 Algorithms based on homogeneous models

In this section, we give an algorithm for solving the primal conic optimization problem $\mathbf{P}(\mathcal{K})$ and then sketch an analogous algorithm for solving its dual $\mathbf{D}(\mathcal{K})$. (Different algorithms are needed given that facial reduction of the primal/dual, as mentioned in Section 2.5, results in an inequivalent dual/primal.) An instance of $\mathbf{P}(\mathcal{K})$ is solved (in a sense we soon make precise) by finding relative interior solutions to a sequence of homogeneous models produced using facial reduction certificates. This sequence terminates once an improving ray or complementary solution is obtained. If necessary, another sequence of homogeneous models is solved (arising from a closely-related feasibility problem) to distinguish between unboundedness and infeasibility.

4.1 Solving the primal problem

To formally state our notion of 'solve,' we need the following set of definitions:

Definition 5. *Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a non-empty, closed, convex cone. For the conic optimization problem $\mathbf{P}(\mathcal{C})$, define the following:*

- *A* complementary solution *is a triple $(x, s, y) \in \mathcal{C} \times \mathcal{C}^* \times \mathbb{R}^m$ satisfying*

$$Ax = b, \qquad s = c - A^T y, \qquad x^T s = 0.$$

- *An* unboundedness certificate *is a tuple $(x, x_{ray}) \in \mathcal{C} \times \mathcal{C}$ satisfying*

$$Ax = b, \qquad Ax_{ray} = 0, \qquad c^T x_{ray} < 0.$$

- *An* infeasibility certificate *is a tuple $(s_{ray}, y_{ray}) \in \mathcal{C}^* \times \mathbb{R}^m$ satisfying*

$$s_{ray} = -A^T y_{ray}, \qquad b^T y_{ray} > 0.$$

Here, complementary solutions are defined as in Section 2, unboundedness certificates consist of a feasible point and an improving ray, and infeasibility certificates are simply improving rays for the dual problem $\mathbf{D}(\mathcal{C})$.

Since some instances of $\mathbf{P}(\mathcal{K})$ do not have complementary solutions, unboundedness certificates, or infeasibility certificates, it is insufficient to say $\mathbf{P}(\mathcal{K})$ is solved if and only if one of these items is found. We therefore introduce two more general notions of 'solve'. The first follows:

Definition 6. *The primal problem* $\mathbf{P}(\mathcal{K})$ *is considered* weakly solved *if one finds a cone* $\mathcal{C} \subseteq \mathbb{R}^n$ *for which the optimal values of* $\mathbf{P}(\mathcal{K})$ *and* $\mathbf{P}(\mathcal{C})$ *are equal, and one of the following items for* $\mathbf{P}(\mathcal{C})$*:*

1. *A complementary solution,*
2. *An unboundedness certificate,*
3. *An infeasibility certificate.*

While any instance of $\mathbf{P}(\mathcal{K})$ can be solved in the weak sense of Definition 6, only the optimal value is necessarily obtained. In particular, the definition doesn't require one of the items (1)-(3) for $\mathbf{P}(\mathcal{K})$ when one exists, neither does it require a feasible point of $\mathbf{P}(\mathcal{K})$ attaining the optimal value when such a point exists. Our second notion of solve addresses these issues:

Definition 7. *The primal problem* $\mathbf{P}(\mathcal{K})$ *is considered* strongly solved *if it is solved in the weak sense of Definition 6 and, in addition, the following conditions hold for* $\mathbf{P}(\mathcal{K})$ *and* $\mathbf{P}(\mathcal{C})$*:*

1. *They are equal, i.e.,* $\mathcal{C} = \mathcal{K}$*, if one of the items* (1)-(3) *of Definition 6 exists for* $\mathbf{P}(\mathcal{K})$*.*
2. *Their feasible sets are equal if the optimal value of* $\mathbf{P}(\mathcal{K})$ *is finite and attained.*

We now develop an algorithm for solving any instance of $\mathbf{P}(\mathcal{K})$ in the sense of Definition 7. Before proceeding, we make a few remarks regarding finite certificates and use Definition 6-7 to compare our algorithm to a two-staged approach that first executes a facial reduction algorithm.

Finite certificates. Definition 7 fails to address another weakness of Definition 6. Specifically, it doesn't require a finite certificate that proves $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ have equal optimal values. We introduce (and show how to find) such a certificate in Section 4.1.3. Discussion of this certificate is postponed to simplify presentation.

Comparison with two-staged approach. Consider a two-staged approach for solving $\mathbf{P}(\mathcal{K})$, where one first identifies the minimal face $\mathcal{F}_{min}$ of $\mathbf{P}(\mathcal{K})$ using a facial reduction algorithm [5, 15, 25] and then solves the regularized problem $\mathbf{P}(\mathcal{F}_{min})$. This two-staged approach does *not* always solve $\mathbf{P}(\mathcal{K})$ in the strong sense of Definition 7. In particular, a complementary solution for $\mathbf{P}(\mathcal{K})$ may exist even though Slater's condition fails, implying $\mathcal{F}_{min} \neq \mathcal{K}$. (These remarks also apply to the recent method of Lourenço et al. [11].) Our algorithm, on the other hand, avoids this issue by regularizing only when improving rays or complementary solutions do not exist (exploiting Corollary 1). The mentioned two-staged approach also fails to solve $\mathbf{P}(\mathcal{K})$ even in the weaker sense of Definition 6 if the primal optimal value is finite and unattained (e.g., Section 5.1) or if the primal is unbounded but has no improving ray (e.g., Section 5.5). Our algorithm avoids this issue by regularizing the dual problem when needed, using the certificates described by Corollary 2.

4.1.1 Finding a complementary solution, infeasibility certificate or improving ray

Our method for solving $\mathbf{P}(\mathcal{K})$ is built upon the following procedure (Algorithm 1), which finds a cone $\mathcal{C}$ such that the primal problems $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ have equal optimal values, and a complementary solution, infeasibility certificate, or improving ray for $\mathbf{P}(\mathcal{C})$. The procedure exits with $\mathcal{C} = \mathcal{K}$ if a complementary solution, infeasibility certificate, or improving ray exists for $\mathbf{P}(\mathcal{K})$. Otherwise, it finds $\mathcal{C}$ using facial reduction, using, potentially, both primal and dual regularization. Theorem 4 summarizes its basic properties. Note if $\mathbf{P}(\mathcal{C})$ has an infeasibility certificate and an improving ray, the infeasibility certificate may not be found. Also note an improving ray is not a complete certificate of unboundedness. We will address these issues in Section 4.1.2.

Algorithm 1: Finds a cone $\mathcal{C}$ such the primal problems $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ have equal optimal values, and a complementary solution, infeasibility certificate or improving ray for $\mathbf{P}(\mathcal{C})$.

```
C ← K
repeat
    Find (x, s, y, τ, κ) in the relative interior of H(C)
    if τ = κ = 0 then
        if s ∉ C^⊥ then
            C ← C ∩ s^⊥
        else
            C ← (C* ∩ x^⊥)*
        end
    else
        return ((x, s, y, τ, κ), C)
    end
until algorithm returns;
```

Theorem 4. *Algorithm 1 has the following basic properties, where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$ are the problem data of the primal problem $\mathbf{P}(\mathcal{K})$.*

1. *Algorithm 1 terminates in finitely-many iterations.*
2. *Algorithm 1 terminates after one iteration, with $\mathcal{C} = \mathcal{K}$, if and only if a complementary solution, infeasibility certificate or improving ray exists for $\mathbf{P}(\mathcal{K})$.*
3. *The dual regularization step $\mathcal{C} \leftarrow (\mathcal{C}^* \cap x^\perp)^*$ executes if and only if one of the following statements hold:*
   - *(a) The optimal value of $\mathbf{P}(\mathcal{K})$ is finite and unattained;*
   - *(b) The optimal value of $\mathbf{P}(\mathcal{K})$ is unbounded below and the set of improving rays $\{x \in \mathcal{K} : Ax = 0, c^T x < 0\}$ is empty.*
4. *Suppose the dual regularization step $\mathcal{C} \leftarrow (\mathcal{C}^* \cap x^\perp)^*$ executes, and let $\mathcal{C}'$ and $\mathcal{C}''$ denote $\mathcal{C}$ just before and after execution. The following statements hold:*
   - *(a) The primal problems $\mathbf{P}(\mathcal{C}')$ and $\mathbf{P}(\mathcal{C}'')$ are feasible and satisfy Slater's condition;*
   - *(b) The primal regularization step $\mathcal{C} \leftarrow \mathcal{C} \cap s^\perp$ is not executed at any following iteration.*

In addition, the following statements hold about the output $((x, s, y, \tau, \kappa), \mathcal{C})$ of Algorithm 1, where $\mathcal{A}_p := \{x \in \mathbb{R}^n : Ax = b\}$.

5. *If $\tau > 0$, then $\frac{1}{\tau}(x, s, y)$ is a complementary solution for $\mathbf{P}(\mathcal{C})$.*
6. *If $\kappa > 0$, then (s, y) is an infeasibility certificate for $\mathbf{P}(\mathcal{C})$ and/or x is an improving ray for $\mathbf{P}(\mathcal{C})$. The former holds if $b^T y > 0$ and the latter if $c^T x < 0$.*
7. *The optimal values of the primal problem $\mathbf{P}(\mathcal{K})$ and the regularized problem $\mathbf{P}(\mathcal{C})$ are equal, i.e.,*
$$\inf \left\{ c^T x : x \in \mathcal{A}_p \cap \mathcal{K} \right\} = \inf \left\{ c^T x : x \in \mathcal{A}_p \cap \mathcal{C} \right\}.$$

8. $\mathcal{A}_p \cap \mathcal{K} \subseteq \mathcal{A}_p \cap \mathcal{C}$, and with strict inclusion only if (3a) *or* (3b) *holds.*

Proof. In arguments below, we call $\mathcal{C} \leftarrow \mathcal{C} \cap s^\perp$ the primal regularization step, and $\mathcal{C} \leftarrow (\mathcal{C}^* \cap x^\perp)^*$ the dual regularization step. When one of these steps executes, we let $\mathcal{C}'$ and $\mathcal{C}''$ denote the cone $\mathcal{C}$ before and after execution, respectively.

Statement 3. We will show $\mathbf{P}(\mathcal{C}')$ satisfies (3a) or (3b) when a dual regularization step executes. Using this, we then show $\mathbf{P}(\mathcal{K})$ also satisfies (3a) or (3b). We use the following facts from Theorem 1: when the dual regularization step executes, all points in $\mathbf{H}(\mathcal{C}')$ satisfy $\tau = \kappa = 0$; and, when the dual regularization step executes, no facial reduction certificate exists for $\mathbf{P}(\mathcal{C}')$ (since $s \in \mathcal{C}'^\perp$).

To begin, we first show $\mathbf{P}(\mathcal{C}')$ is feasible, and hence has finite optimal value or is unbounded below. Suppose $\mathbf{P}(\mathcal{C}')$ is infeasible. If $\{x \in \mathbb{R}^n : Ax = b\}$ is empty, then there exists $\hat{y}$ for which $b^T\hat{y} = 1$ and $A^T\hat{y} = 0$. Hence $(0, 0, \hat{y}, 0, b^T\hat{y})$ is a point in $\mathbf{H}(\mathcal{C}')$ with $\kappa > 0$, which is a contradiction. On the other hand, if $\{x \in \mathbb{R}^n : Ax = b\}$ is non-empty, there exists a hyperplane properly seperating $\mathcal{A}_p := \{x \in \mathbb{R}^n : Ax = b\}$ from the relative interior of $\mathcal{C}'$. That is, there exists $\hat{s} \in \mathcal{C}'^*$ for which

$$\hat{s}^T x \leq 0, \quad \forall x \in x_0 + \operatorname{null} A,$$
$$\hat{s}^T z \neq 0 \text{ for some } z \in (x_0 + \operatorname{null} A) \cup \mathcal{C}',$$

where $x_0 \in \mathcal{A}_p$ and $\mathcal{A}_p = x_0 + \operatorname{null} A$. This implies $\hat{s} \in (\operatorname{null} A)^\perp = \operatorname{range} A^T$. Hence, $\hat{s} = -A^T\hat{y}$ for some $\hat{y}$, where, evidently, $\hat{s}^T x = -b^T\hat{y} \leq 0$ for all $x \in \mathcal{A}_p$. If $b^T\hat{y} = 0$, then $\hat{s}^T z \neq 0$ for some $z \in \mathcal{C}'$ by proper separation of the sets. Hence, $\hat{s}$ is a facial reduction certificate which, as mentioned above, cannot exist. On the other hand, if $b^T\hat{y} > 0$, then $(0, -A^T\hat{y}, \hat{y}, 0, b^T\hat{y})$ is a point in $\mathbf{H}(\mathcal{C}')$ with $\kappa > 0$, which is a contradiction. Hence, $\mathbf{P}(\mathcal{C}')$ must be feasible and either has finite optimal value or an optimal value that is unbounded below.

We have established that $\mathbf{P}(\mathcal{C}')$ is feasible and that no facial reduction certificate for $\mathbf{P}(\mathcal{C}')$ exists. Hence, by Proposition 1, $\mathbf{P}(\mathcal{C}')$ is strictly feasible. Now suppose that $\mathbf{P}(\mathcal{C}')$ has a finite optimal value. Then, by Slater's condition, the dual $\mathbf{D}(\mathcal{C}')$ of $\mathbf{P}(\mathcal{C}')$ has equal optimal value that is attained. Hence, if $\mathbf{P}(\mathcal{C}')$ attains its optimal value, $\mathbf{P}(\mathcal{C}')$ has a complementary solution (x, s, y) where $(x, s, y, 1, 0)$ is a point in $\mathbf{H}(\mathcal{C}')$ with $\tau > 0$; a contradiction. Suppose next the optimal value equals $-\infty$. If an improving ray $\hat{x}_{ray}$ exists, then $(\hat{x}_{ray}, 0, 0, 0, -c^T\hat{x}_{ray})$ is a point in $\mathbf{H}(\mathcal{C}')$ with $\kappa > 0$; a contradiction. Hence, $\mathbf{P}(\mathcal{C}')$ satisfies (3a) or (3b).

Now consider the first time the dual regularization step executes. Since the feasible sets of $\mathbf{P}(\mathcal{C}')$ and $\mathbf{P}(\mathcal{K})$ are equal, it trivially follows that $\mathbf{P}(\mathcal{K})$ satisfies (3a) if $\mathbf{P}(\mathcal{C}')$ does. If $\mathbf{P}(\mathcal{C}')$ satisfies (3b), then $\mathbf{P}(\mathcal{K})$ is clearly unbounded. Suppose then $\mathbf{P}(\mathcal{K})$ has an improving ray x_{ray}. Then, for any feasible point x_0 and facial reduction certificate s used by the primal regularization step,

$$0 = s^T(x_0 + x_{ray}) = s^T x_{ray}.$$

Hence, $x_{ray} \in \mathcal{C}'$ and is therefore an improving ray for $\mathbf{P}(\mathcal{C}')$, a contradiction.

For the converse direction, we will argue $\tau = \kappa = 0$ holds at each iteration unless the dual regularization step executes. Since the primal regularization step can execute only finitely many times (since $\mathcal{K}$ is finite-dimensional), the converse direction therefore follows. To begin, suppose the optimal value of $\mathbf{P}(\mathcal{K})$ is finite and unattained, i.e., suppose (3a) holds. Then $\tau = \kappa = 0$ for all $(x, s, y, \tau, \kappa) \in \operatorname{relint} \mathbf{H}(\mathcal{K})$; otherwise, either an infeasibility certificate/improving ray would exist for $\mathbf{P}(\mathcal{K})$, contradicting finiteness, or a complementary solution would exist, contradicting unattainment. Since the feasible sets of $\mathbf{P}(\mathcal{C}')$ and $\mathbf{P}(\mathcal{K})$ are equal unless the dual regularization step executes, repeating this argument shows $\tau = \kappa = 0$ unless the dual regularization step executes. A similar argument shows the claim assuming (3b).

Case	# prim. steps	# dual steps
#1	0	0
#2	d_P^*	n
#3	d_P^*	0

Table 2: Upper bounds on primal and dual regularization steps in three cases. In the first, $\mathbf{P}(\mathcal{K})$ has a complementary solution, infeasibility certificate, or improving ray (denoted Case #1). In the second, $\mathbf{P}(\mathcal{K})$ has a finite-unattained optimal value or is unbounded but has no improving ray (denoted Case #2). The third is simply all other possibilities (denoted Case #3).

Statement 4a. Strict feasibility of $\mathbf{P}(\mathcal{C}')$ was established in the proof of statement 3. This implies $\mathbf{P}(\mathcal{C}'')$ is strictly feasible given that

$$\mathcal{A}_p \cap \operatorname{relint} \mathcal{C}' \subseteq \mathcal{A}_p \cap \operatorname{relint}(\mathcal{C}' + \operatorname{span} x) = \mathcal{A}_p \cap \operatorname{relint}(\mathcal{C}'^* \cap x^\perp)^* = \mathcal{A}_p \cap \operatorname{relint} \mathcal{C}''.$$

Statement 4b. By (4a), if dual regularization is performed, then $\mathbf{P}(\mathcal{C}')$ satisfies Slater's condition, and continues to satisfy Slater's condition at each ensuing iteration. Hence, a facial reduction certificate s cannot exist at any ensuing iteration by Proposition 2.

Statement 1. Since $\mathcal{K}$ is finite dimensional, it trivially follows, using (4b), that both regularization steps can execute only finitely many times. Hence, the algorithm must terminate.

Statement 2 and 5-6. Immediate from Corollary 1.

Statement 7. The optimal values of $\mathbf{P}(\mathcal{C}')$ and $\mathbf{P}(\mathcal{C}'')$ are equal when the primal regularization step executes. Similarly, the optimal values of $\mathbf{D}(\mathcal{C}')$ and $\mathbf{D}(\mathcal{C}'')$ are equal when the dual regularization step executes. Moreover, (4a) and Slater's condition imply the optimal values of $\mathbf{P}(\mathcal{C}')$ and $\mathbf{D}(\mathcal{C}')$ are equal when the dual regularization step executes. Combining these facts with (4b) shows the optimal values of $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ are the same at termination.

Statement 8. When the primal regularization step executes, $\mathbf{P}(\mathcal{C}')$ and $\mathbf{P}(\mathcal{C}'')$ have equal feasible sets since s is a facial reduction certificate for $\mathbf{P}(\mathcal{C}')$. When the dual regularization step executes, the feasible set $\mathcal{A}_p \cap \mathcal{C}'$ of $\mathbf{P}(\mathcal{C}')$ and the feasible set $\mathcal{A}_p \cap \mathcal{C}''$ of $\mathbf{P}(\mathcal{C}'')$ satisfy

$$\mathcal{A}_p \cap \mathcal{C}' \subseteq \mathcal{A}_p \cap \overline{\mathcal{C}' + \operatorname{span} x} = \mathcal{A}_p \cap (\mathcal{C}'^* \cap x^\perp)^* = \mathcal{A}_p \cap \mathcal{C}''.$$

Combining these facts with (4b) shows $\mathcal{A}_p \cap \mathcal{K} \subseteq \mathcal{A}_p \cap \mathcal{C}$. Since dual regularization is performed if and only if (3a) or (3b) hold, the claim follows. ☐

Iteration bounds. Immediate from Theorem 4 and Corollary 1 are a set of upper bounds on regularization steps executed by Algorithm 1. Table 2 summarizes these bounds, where d_P^* denotes the singularity degree of the primal problem $\mathbf{P}(\mathcal{K})$—the maximum number of primal regularization steps possible if optimal facial reduction certificates are used (Section 2.4). The upper bounds equal to d_P^* hold since the facial reduction certificates used by Algorithm 1 are optimal (Corollary 1). Note the singularity degree d_D^* of the dual $\mathbf{D}(\mathcal{K})$ is not used in any of the bounds. We conjecture the trivial dimension bound n (where $\mathcal{K} \subseteq \mathbb{R}^n$) can be improved to d_D^*, but could not find a simple proof since the dual problem—and potentially its singularity degree—changes with primal regularization (Section 2.5).

4.1.2 The complete algorithm

We now give a complete algorithm for solving the primal problem $\mathbf{P}(\mathcal{K})$ using one or two calls to Algorithm 1. Two calls may be needed for the following reasons:

- A certificate of unboundedness requires both a feasible point and an improving ray, which cannot be simultaneously obtained from a single call to Algorithm 1.
- If both $\mathbf{P}(\mathcal{K})$ and its dual are infeasible, the output $((x, s, y, \tau, \kappa), \mathcal{C})$ of Algorithm 1 doesn't necessarily produce an infeasibility certificate (s, y) for $\mathbf{P}(\mathcal{C})$. In particular, $b^T y \leq 0$ may hold if $c^T x > 0$. This problem was illustrated by Example 1.

Both of these issues are resolved by re-executing Algorithm 1 with the cost vector c set to zero. This either produces a feasible point (completing the unboundedness certificate) or an infeasibility certificate (since $c^T x > 0$ can no longer hold). The complete method for solving $\mathbf{P}(\mathcal{K})$ performs this re-execution when needed and appears in Algorithm 2.

Algorithm 2: Solves the primal problem $\mathbf{P}(\mathcal{K})$ in the strong sense of Definition 7.

Execute Algorithm 1 and let $((x, s, y, \tau, \kappa), \mathcal{C})$ denote the output
if $\kappa > 0$ **and** $b^T y \leq 0$ **then**
  $x_{ray} \leftarrow x$
  Re-execute Algorithm 1 with $c = 0$ and let $((x, s, y, \tau, \kappa), \mathcal{C}_{feas})$ denote the output.
  return $\begin{cases} \text{Unboundedness certificate } ((\frac{1}{\tau}x, x_{ray}), \mathcal{C}), & \text{if } \tau > 0, \\ \text{Infeasibility certificate } ((s, y), \mathcal{C}_{feas}), & \text{otherwise.} \end{cases}$
else
  return $\begin{cases} \text{Complementary solution } (\frac{1}{\tau}(x, s, y), \mathcal{C}), & \text{if } \tau > 0, \\ \text{Infeasibility certificate } ((s, y), \mathcal{C}), & \text{otherwise.} \end{cases}$
end

As stated by Theorem 5, Algorithm 2 solves any instance of $\mathbf{P}(\mathcal{K})$ in the strong sense. When Algorithm 1 does not re-execute, this is a relatively straight-forward consequence of Theorem 4. To show it in the general case, we need the following intermediate result:

Lemma 2. *Let $((x, s, y, \tau, \kappa), \mathcal{C}_{feas})$ denote the tuple obtained by re-execution of Algorithm 1 with $c = 0$. The following statements hold.*

1. *If $\tau > 0$, then $\frac{1}{\tau}x$ is a feasible point of $\mathbf{P}(\mathcal{K})$.*
2. *If $\kappa > 0$, then $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C}_{feas})$ are both infeasible, and (s, y) is an infeasibility certificate for $\mathbf{P}(\mathcal{C}_{feas})$*

Proof. Algorithm 1 terminates with a complementary solution if $\tau > 0$, or an infeasibility certificate if $\kappa > 0$ as no improving ray can exist with $c = 0$. As dual regularization will not occur (Theorem 4, statement 3), the feasible sets and optimal values of $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C}_{feas})$ are equal, proving both statements. □

Theorem 5. *Algorithm 2 solves the primal problem $\mathbf{P}(\mathcal{K})$ in the strong sense of Definition 7.*

Proof. We first show the primal problem is solved in the *weak sense* of Definition 6 by establishing outputs are complementary solutions, infeasibility certificates, or unboundedness certificates for a primal problem, $\mathbf{P}(\mathcal{C})$ or $\mathbf{P}(\mathcal{C}_{feas})$, with optimal value equal to $\mathbf{P}(\mathcal{K})$.

If Algorithm 1 is only called once, this follows from Theorem 4. (Specifically, a complementary solution is returned if $\tau > 0$ and an infeasibility certificate is returned otherwise.) Now suppose Algorithm 1 re-executes with $c = 0$ and produces the point $((x, s, y, \tau, \kappa), \mathcal{C}_{feas})$. If $\tau > 0$, the algorithm returns $((\frac{1}{\tau}x, x_{ray}), \mathcal{C})$ containing an unboundedness certificate for $\mathbf{P}(\mathcal{C})$; specifically, x_{ray} is an improving ray for $\mathbf{P}(\mathcal{C})$ and $\frac{1}{\tau}x$ is a feasible point of $\mathbf{P}(\mathcal{K})$, by Lemma 2, and hence also of $\mathbf{P}(\mathcal{C})$, by Theorem 4, statement 8. Moreover, $\mathbf{P}(\mathcal{C})$ and $\mathbf{P}(\mathcal{K})$ have equal optimal values by Theorem 4. If $\kappa > 0$, the algorithm returns $((s, y), \mathcal{C}_{feas})$, where (s, y) is an infeasibility certificate for $\mathbf{P}(\mathcal{C}_{feas})$ by Lemma 2. That $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C}_{feas})$ have equal optimal values also follows from Lemma 2.

Finally, that the outputs solve $\mathbf{P}(\mathcal{K})$ in the *strong sense* of Definition 7 follows from statement 2 and 8 of Theorem 4. $\square$

4.1.3 The regularization certificate

We now introduce a finite certificate, called a *regularization certificate*, that proves $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ have equal optimal values. Using this certificate, one can independently verify Algorithm 2 solves $\mathbf{P}(\mathcal{K})$ in the weak sense of Definition 6. In particular, augmenting this certificate with an infeasibility certificate, complementary solution, or unboundedness certificate for $\mathbf{P}(\mathcal{C})$ provides a finite certificate for the optimal value of $\mathbf{P}(\mathcal{K})$.

The regularization certificate consists of facial reduction certificates (see Definition 2) and, potentially, a Slater point (see Section 2.2). It has the following definition.

Definition 8. *Let $\mathcal{K}, \mathcal{C} \subseteq \mathbb{R}^n$ be non-empty, closed, convex cones. A* regularization certificate *for $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ is a triple (Z_P, Z_D, x_0) consisting of sequences Z_P and Z_D of length $d_P \geq 0$ and $d_D \geq 0$, respectively, and a point x_0 for which:*

1. *$Z_P = s_1, \ldots, s_{d_P}$, where s_i is a facial reduction certificate for $\mathbf{P}(\mathcal{F}_{i-1})$ and $\mathcal{F}_i$ is defined by the recursion:*
$$\mathcal{F}_0 = \mathcal{K}, \quad \mathcal{F}_i = \mathcal{F}_{i-1} \cap s_i^{\perp} \quad i \in \{1, \ldots, d_P\}.$$
2. *$Z_D = x_1, \ldots, x_{d_D}$, where x_i is a facial reduction certificate for $\mathbf{D}(\widehat{\mathcal{F}}_{i-1}^*)$ and $\widehat{\mathcal{F}}_i$ is defined by the recursion:*
$$\widehat{\mathcal{F}}_0 = (\mathcal{F}_{d_P})^*, \quad \widehat{\mathcal{F}}_i = \widehat{\mathcal{F}}_{i-1} \cap x_i^{\perp} \quad i \in \{1, \ldots, d_D\}, \quad \widehat{\mathcal{F}}_{d_D} = \mathcal{C}^*.$$
3. *x_0 is a Slater point of $\mathbf{P}(\mathcal{F}_{d_P})$ if $d_D \geq 1$.*

Combining this definition with Slater's condition and the definition of facial reduction certificates immediately yields the following:

Proposition 5. *Suppose a regularization certificate exists for $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ in the sense of Definition 8. The following statements hold.*

1. *The problems are equal, i.e., $\mathcal{C} = \mathcal{K}$, if $d_P = d_D = 0$.*
2. *The feasible sets and optimal values of $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ are equal if $d_D = 0$.*
3. *The feasible sets and optimal values of $\mathbf{D}(\mathcal{K})$ and $\mathbf{D}(\mathcal{C})$ are equal if $d_P = 0$.*
4. *The optimal values of $\mathbf{P}(\mathcal{K})$ and $\mathbf{P}(\mathcal{C})$ are equal in all cases (given $\mathbf{P}(\mathcal{F}_{d_P})$ and $\mathbf{D}(\mathcal{F}_{d_P})$ have equal optimal values by Slater's condition when $d_D \geq 1$).*

Constructing the certificate. A regularization certificate can be constructed with simple modifications to Algorithm 1. The first changes are mainly book keeping, that is, every time the primal regularization step $\mathcal{C} \leftarrow \mathcal{C} \cap s^{\perp}$ executes, one should append the facial reduction certificate s to Z_P. Similarly, every time the dual regularization step $\mathcal{C} \leftarrow (\mathcal{C}^* \cap x^{\perp})^*$ executes, one should append the facial reduction certificate x to Z_D. Since the algorithm can only switch from primal to dual regularization once (by Statement 4b of Theorem 4), the constructed sequences Z_P and Z_D match Definition 8.

If dual regularization is used, the regularization certificate also requires a Slater point to certify that Slater's condition holds for $\mathbf{P}(\mathcal{F}_{d_P})$. By Theorem 2 and strict feasibility of $\mathbf{P}(\mathcal{F}_{d_P})$ (which holds by Theorem 4, statement 4a), a Slater point for $\mathbf{P}(\mathcal{F}_{d_P})$ can be found by solving an extra feasibility problem just before Algorithm 1 executes its first dual regularization step.

Relationship with other certificates. We remark, finally, that the basic idea of augmenting improving rays and complementary solutions with regularization certificates is *not* new. As explained in [20], a simpler regularization certificate (consisting of only one sequence and no Slater point) augmented with an optimal solution or an improving ray solves Ramana [19]'s exact dual for semidefinite programs. Solutions to the extended-duals of Pataki [15] have similar interpretations. Infeasibility certificates of this type also appear in a recent paper of Liu and Pataki [10]. Note these simpler regularization certificates correspond to only primal regularization; hence, they do not certify optimal values that are finite and unattained (shown in Section 5.1), nor optimal values for problems that are unbounded but have no improving ray (shown in Section 5.5).

4.2 Solving the dual problem

To solve the dual problem $\mathbf{D}(\mathcal{K})$ in a sense analogous to Definition 7, one must modify Algorithm 1 by swapping the order of primal and dual regularization steps. Algorithm 3 makes this modification explicit. A general procedure, analogous to Algorithm 2, would solve $\mathbf{D}(\mathcal{K})$ by calling Algorithm 3 first with $\mathbf{D}(\mathcal{K})$ as input. If necessary, it would then re-execute Algorithm 3 with the cost vector b set to zero, to certify unboundedness or infeasibility. We forgo making this explicit.

Algorithm 3: Finds cone $\mathcal{C}$ such the dual problems $\mathbf{D}(\mathcal{K})$ and $\mathbf{D}(\mathcal{C})$ have equal optimal values, and a complementary solution, infeasibility certificate or improving ray for $\mathbf{D}(\mathcal{C})$.

```
C ← K
repeat
    Find (x, s, y, τ, κ) in the relative interior of H(C)
    if τ = κ = 0 then
        if x ∉ (C*)^⊥ then
            C ← (C* ∩ x^⊥)*
        else
            C ← C ∩ s^⊥
        end
    else
        return ((x, s, y, τ, κ), C)
    end
until algorithm returns;
```

5 Examples

We now analyze relative interior solutions of the homogeneous model for simple conic optimization problems. We consider SOCPs over the quadratic cone $\mathcal{Q}^{1+k} = \{(r,x) \in \mathbb{R}^1_+ \times \mathbb{R}^k \mid r^2 \geq x^T x\}$ and the rotated quadratic cone $\mathcal{Q}_r^{2+k} = \{(r,x) \in \mathbb{R}^2_+ \times \mathbb{R}^k \mid 2r_1 r_2 \geq x^T x\}$. Since these cones and their Cartesian products are closed and convex (and, indeed, self-dual), the developed theory applies. The primal-dual pairs (10) and (11) are taken from [2].

5.1 Unattained, finite optimal values

Consider the primal-dual pair over a rotated quadratic cone:

$$\begin{array}{ll} \text{minimize} & x_2 \\ \text{subject to} & x_3 = \sqrt{2} \\ & x \in \mathcal{Q}_r^3 \end{array} \qquad \begin{array}{ll} \text{maximize} & \sqrt{2}y \\ \text{subject to} & \begin{pmatrix} 0\\1\\0 \end{pmatrix} - \begin{pmatrix} 0\\0\\y \end{pmatrix} = s \\ & s \in \mathcal{Q}_r^3. \end{array} \tag{9}$$

It is easy to verify the optimal value of the primal is unattained and equals $\inf\{1/x_1 : x_1 \in \mathbb{R}_+\} = 0$, and that the dual problem is feasible.

Solutions to the homogeneous model. Since the primal and dual problem are both feasible, but the primal problem has an unattained optimal value, all relative interior solutions to the self-dual homogeneous model (2) must satisfy $\tau = \kappa = 0$ and yield facial reduction certificates. Because $(10, 10, \sqrt{2}) \in \operatorname{relint} \mathcal{Q}_r^3 \cap \{x \in \mathbb{R}^3 : x_3 = \sqrt{2}\}$ is a strictly feasible point of the primal, we also conclude relative interior solutions to (2) yield only dual facial reduction certificates (Theorem 1, (3c) and (3e)). As an example, the solution

$$\hat{x} = (1, 0, 0), \quad \hat{s} = (0, 0, 0), \quad \hat{y} = \hat{\tau} = \hat{\kappa} = 0,$$

yields the dual facial reduction certificate $\hat{x}$.

Facial reduction of the dual. Using $\hat{x}$, we can regularize the dual to obtain a new primal-dual pair. To perform this regularization, we replace $\mathcal{Q}_r^3$ with $\mathcal{Q}_r^3 \cap \hat{x}^\perp = \{0\} \times \mathbb{R}_+ \times \{0\}$ in the dual problem and $\mathcal{Q}_r^3$ with $(\mathcal{Q}_r^3 \cap \hat{x}^\perp)^* = \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R}$ in the primal problem. That is:

$$\begin{array}{ll} \text{minimize} & x_2 \\ \text{subject to} & x_3 = \sqrt{2} \\ & x \in \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R} \end{array} \qquad \begin{array}{ll} \text{maximize} & \sqrt{2}y \\ \text{subject to} & \begin{pmatrix} 0\\1\\0 \end{pmatrix} - \begin{pmatrix} 0\\0\\y \end{pmatrix} = s \\ & s \in \{0\} \times \mathbb{R}_+ \times \{0\}. \end{array}$$

Taking $x = (0, 0, \sqrt{2})$, $s = (0, 1, 0)$ and $y = 0$ yields a primal-dual optimal solution with optimal value equal to zero. Note (s, y) solves the dual problem of (9) and certifies the unattained optimal value of the primal problem of (9).

5.2 Weak infeasibility when one problem is feasible

Consider the following primal-dual pair:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 - x_2 = 0 \\ & x \in \mathcal{Q}^3 \end{array} \qquad \begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & \begin{pmatrix} 0\\0\\1 \end{pmatrix} - \begin{pmatrix} y\\-y\\0 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3. \end{array} \tag{10}$$

Here, the dual problem is weakly infeasible, meaning there exists $s \in \mathcal{Q}^3$ that solve the equation system with arbitrarily small positive error, but no $s \in \mathcal{Q}^3$ that solves it exactly. The primal problem is feasible and has optimal value zero, but becomes unbounded if the equation $x_1 - x_2 = 0$ is replaced with $x_1 - x_2 = \epsilon$ for any $\epsilon > 0$.

Solutions to the homogeneous model. For this problem, no relative interior solution to the self-dual homogeneous model (2) can satisfy $\tau > 0$ or $\kappa > 0$. On the other hand, there is a solution given by

$$\hat{x} = (1, 1, 0), \quad \hat{s} = (1, -1, 0), \quad \hat{y} = -1, \quad \hat{\tau} = \hat{\kappa} = 0,$$

where $\hat{x}$ is a facial reduction certificate for the dual, and $(\hat{s}, \hat{y})$ is a facial reduction certificate for the primal.

Facial reduction of the primal. Using $\hat{s}$, we regularize the primal problem and formulate a new primal-dual pair. To perform this regularization, we replace $\mathcal{Q}^3$ with $\mathcal{Q}^3 \cap \hat{s}^\perp = \left(\begin{smallmatrix}1\\1\end{smallmatrix}\right) \mathbb{R}_+ \times \{0\}$ in the primal problem, where $\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right) \mathbb{R}_+$ denotes the cone generated by $\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)$. We then replace $\mathcal{Q}^3$ with $(\mathcal{Q}^3 \cap \hat{s}^\perp)^* = \{\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)\}^* \times \mathbb{R}$ in the dual problem, where $\{\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)\}^*$ denotes the dual cone of the singleton $\{\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)\}$. That is:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 - x_2 = 0 \\ & x \in \left(\begin{smallmatrix}1\\1\end{smallmatrix}\right) \mathbb{R}_+ \times \{0\} \end{array} \qquad \begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right) - \left(\begin{smallmatrix}y\\-y\\0\end{smallmatrix}\right) = s \\ & s \in \{\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)\}^* \times \mathbb{R}. \end{array}$$

Taking $x = (0, 0, 0)$, $s = (0, 0, 1)$ and $y = 0$ yields a primal-dual optimal solution pair with optimal value equal to zero.

Facial reduction of the dual. Using $\hat{x}$, we can also regularize the dual to obtain a new primal-dual pair. Here, $\mathcal{Q}^3$ is replaced by $(\mathcal{Q}^3 \cap \hat{x}^\perp)^*$ in the primal problem, and by $\mathcal{Q}^3 \cap \hat{x}^\perp$ in the dual. That is:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 - x_2 = 0 \\ & x \in \{\left(\begin{smallmatrix}1\\-1\end{smallmatrix}\right)\}^* \times \mathbb{R} \end{array} \qquad \begin{array}{ll} \text{maximize} & 0 \\ \text{subject to} & \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right) - \left(\begin{smallmatrix}y\\-y\\0\end{smallmatrix}\right) = s \\ & s \in \left(\begin{smallmatrix}1\\-1\end{smallmatrix}\right) \mathbb{R}_+ \times \{0\}. \end{array}$$

The improving ray $x = (0, 0, -1)$ in the primal problem certifies dual infeasibility.

5.3 Finite but nonzero duality gap

The following primal-dual pair of problems are both feasible, but the optimal values of the two problems are different:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 + x_2 + x_4 + x_5 = 0 \\ & -x_3 + x_4 = 1 \\ & x \in \mathcal{Q}^3 \times \mathcal{Q}^2 \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \begin{pmatrix} 0\\0\\1\\0\\0 \end{pmatrix} - \begin{pmatrix} y_1\\y_1\\-y_2\\y_1+y_2\\y_1 \end{pmatrix} = s \\ & s \in \mathcal{Q}^3 \times \mathcal{Q}^2. \end{array} \tag{11}$$

Given their membership in $\mathcal{Q}^3 \times \mathcal{Q}^2$, primal feasible points satisfy the inequalities $x_1 + x_2 \geq 0$ and $x_4 + x_5 \geq 0$. From the equation $x_1 + x_2 + x_4 + x_5 = 0$, we conclude that $x_1 + x_2 = 0$; combining this with the conic constraint $(x_1, x_2, x_3) \in \mathcal{Q}^3$ shows $x_3 = 0$ for all primal feasible points. On the other hand, dual feasible points satisfy $s_1 = s_2$, which in turn implies that $s_3 = 0$. This shows that $y_2 = -1$ for all dual feasible points. Hence, the primal optimal value is zero and the dual optimal value is -1.

Solutions to the homogeneous model. For this problem, no relative interior solution to the self-dual homogeneous model (2) can satisfy $\tau > 0$ because the optimal values of the primal and dual problem differ. Nor can it satisfy $\kappa > 0$ as both problems are feasible. Hence, all relative interior solutions yield facial reduction certificates. Consider

$$\hat{x} = (1, -1, 0, 0, 0), \quad \hat{s} = (1, 1, 0, 1, 1), \quad \hat{y} = (-1, 0), \quad \hat{\tau} = \hat{\kappa} = 0,$$

where $\hat{x}$ and $(\hat{s}, \hat{y})$ are dual and primal facial reduction certificates, respectively.

Facial reduction of the primal. Using $\hat{s}$, we regularize the primal problem to obtain a new primal-dual pair. Here, we replace $\mathcal{Q}^3 \times \mathcal{Q}^2$ with $(\mathcal{Q}^3 \cap \hat{s}_{1:3}^{\perp}) \times (\mathcal{Q}^2 \cap \hat{s}_{4:5}^{\perp})$ in the primal problem and replace $\mathcal{Q}^3 \times \mathcal{Q}^2$ with $(\mathcal{Q}^3 \cap \hat{s}_{1:3}^{\perp})^* \times (\mathcal{Q}^2 \cap \hat{s}_{4:5}^{\perp})^*$ in the dual. That is:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 + x_2 + x_4 + x_5 = 0 \\ & -x_3 + x_4 = 1 \\ & x \in \left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right) \mathbb{R}_+ \times \{0\} \times \left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right) \mathbb{R}_+ \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_1 \\ -y_2 \\ y_1+y_2 \\ y_1 \end{pmatrix} = s \\ & s \in \left\{\left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right)\right\}^* \times \mathbb{R} \times \left\{\left(\begin{smallmatrix} 1 \\ -1 \end{smallmatrix}\right)\right\}^*. \end{array}$$

Taking $x = (0, 0, 0, 1, -1)$, $s = (0, 0, 1, 0, 0)$ and $y = (0, 0)$ yields a primal-dual optimal solution pair with optimal value equal to zero.

Facial reduction of the dual. Using $\hat{x}$, we can also regularize the dual problem to obtain a new primal-dual pair. Here, we replace $\mathcal{Q}^3 \times \mathcal{Q}^2$ with $(\mathcal{Q}^3 \cap \hat{x}_{1:3}^{\perp}) \times (\mathcal{Q}^2 \cap \hat{x}_{4:5}^{\perp})$ in the dual problem and $\mathcal{Q}^3 \times \mathcal{Q}^2$ with $(\mathcal{Q}^3 \cap \hat{x}_{1:3}^{\perp})^* \times (\mathcal{Q}^2 \cap \hat{x}_{4:5}^{\perp})^*$ in the primal problem. That is:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 + x_2 + x_4 + x_5 = 0 \\ & -x_3 + x_4 = 1 \\ & x \in \left\{\left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right)\right\}^* \times \mathbb{R} \times \mathcal{Q}^2 \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 \\ \text{subject to} & \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_1 \\ -y_2 \\ y_1+y_2 \\ y_1 \end{pmatrix} = s \\ & s \in \left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right) \mathbb{R}_+ \times \{0\} \times \mathcal{Q}^2. \end{array}$$

Taking $x = (0, 0, -1, 0, 0)$, $s = (0, 0, 0, 1, 0)$ and $y = (0, -1)$ yields a primal-dual optimal solution pair with optimal value equal to -1.

5.4 Weak infeasibility when both problems are infeasible

Consider the following primal-dual pair:

$$\begin{array}{ll} \text{minimize} & -x_2 \\ \text{subject to} & x_1 = 0 \\ & x_3 = 1 \\ & x_4 = 1 \\ & x \in \mathcal{Q}_r^3 \times \mathbb{R}_+ \end{array} \qquad \begin{array}{ll} \text{maximize} & y_2 + y_3 \\ \text{subject to} & \begin{pmatrix} 0 \\ -1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y_1 \\ 0 \\ y_2 \\ y_3 \end{pmatrix} = s \\ & s \in \mathcal{Q}_r^3 \times \mathbb{R}_+, \end{array} \tag{12}$$

where both the primal and dual problem are infeasible. Any relative interior solution to the homogeneous model will yield a primal improving ray such as $\hat{x} = (0, 1, 0, 0)$ to certify that the dual problem is infeasible, but cannot yield a dual improving ray. This is because all dual rays satisfy $s_2 = 0$ (implying $s_3 = -y_2 = 0$) and $y_3 \leq 0$, showing the objective value to be nonpositive. In response to the unknown primal feasibility status, we follow the same procedure as in Example 1, and construct the primal feasibility problem:

$$
\begin{array}{ll}
\text{minimize} & 0 \\
\text{subject to} & x_1 = 0 \\
& x_3 = 1 \\
& x_4 = 1 \\
& x \in \mathcal{Q}_r^3 \times \mathbb{R}_+
\end{array}
\qquad
\begin{array}{ll}
\text{maximize} & y_2 + y_3 \\
\text{subject to} & \left(\begin{smallmatrix} 0 \\ 0 \\ 0 \\ 0 \end{smallmatrix}\right) - \left(\begin{smallmatrix} y_1 \\ 0 \\ y_2 \\ y_3 \end{smallmatrix}\right) = s \\
& s \in \mathcal{Q}_r^3 \times \mathbb{R}_+,
\end{array}
\tag{13}
$$

All dual rays of (12) are now dual solutions of (13), and thus satisfy $s_2 = 0$ (implying $s_3 = -y_2 = 0$) and $y_3 \leq 0$ as derived.

Solutions to the homogeneous model. The primal problem of (13) is infeasible, but there is no dual improving ray. Hence, all relative interior solutions to the homogeneous model satisfy $\tau = \kappa = 0$. Moreover, since $c^T x = 0$, the equation $b^T y - c^T x - \kappa = 0$ implies $b^T y = 0$ and thus $y_3 = 0$. One example of relative interior solution is given by:

$$\hat{x} = (0, 1, 0, 0), \quad \hat{s} = (1, 0, 0, 0), \quad \hat{y} = (-1, 0, 0), \quad \hat{\tau} = \hat{\kappa} = 0,$$

where $\hat{x}$ is a dual facial reduction certificate and $\hat{s}$ is a primal facial reduction certificate. We highlight the subtlety that $\hat{y}_3 = 0$ for all points in the relative interior of the homogeneous model, even though all relative interior feasible points (s, y) of the dual problem of (13) satisfy $y_3 < 0$.

Facial reduction of the primal. Using $\hat{s}$, we regularize the primal problem of (13) and formulate a new primal-dual pair. That is:

$$
\begin{array}{ll}
\text{minimize} & 0 \\
\text{subject to} & x_1 = 0 \\
& x_3 = 1 \\
& x_4 = 1 \\
& x \in \{0\} \times \mathbb{R}_+ \times \{0\} \times \mathbb{R}_+
\end{array}
\qquad
\begin{array}{ll}
\text{maximize} & y_2 + y_3 \\
\text{subject to} & \left(\begin{smallmatrix} 0 \\ 0 \\ 0 \\ 0 \end{smallmatrix}\right) - \left(\begin{smallmatrix} y_1 \\ 0 \\ y_2 \\ y_3 \end{smallmatrix}\right) = s \\
& s \in \mathbb{R} \times \mathbb{R}_+ \times \mathbb{R} \times \mathbb{R}_+.
\end{array}
$$

The improving ray $y = (0, 1, 0)$ and $s = (0, 0, -1, 0)$ in the dual problem certifies primal infeasibility. Hence, the primal problem of (12) is also infeasible.

5.5 Unboundedness with no improving ray

Consider the following primal-dual pair:

$$
\begin{array}{ll}
\text{minimize} & x_3 \\
\text{subject to} & x_1 = 1 \\
& x \in \mathcal{Q}_r^3
\end{array}
\qquad
\begin{array}{ll}
\text{maximize} & y \\
\text{subject to} & \left(\begin{smallmatrix} 0 \\ 0 \\ 1 \end{smallmatrix}\right) - \left(\begin{smallmatrix} y \\ 0 \\ 0 \end{smallmatrix}\right) = s \\
& s \in \mathcal{Q}_r^3,
\end{array}
\tag{14}
$$

where the primal problem is unbounded and the dual problem is infeasible. Though unbounded, the primal problem has no improving ray, since all primal rays satisfy $x_1 = 0$ (implying $x_3 = 0$).

Solutions to the homogeneous model. For this problem, all relative interior solutions to (2) satisfy $\tau = \kappa = 0$. Hence, since all primal rays satisfy $x_3 = 0$, the equation $b^T y - c^T x - \kappa = 0$ implies $y = 0$. A relative interior solution is thus given by:

$$\hat{x} = (0, 1, 0), \quad \hat{s} = (0, 0, 0), \quad \hat{y} = 0, \quad \hat{\tau} = \hat{\kappa} = 0,$$

where $\hat{x}$ is a facial reduction certificate for the dual problem. By Theorem 4, the absence of a facial reduction certificate for the primal implies the primal is unbounded with no improving ray or has a finite optimal value that is unattained. This fact, along with Theorem 1-(3c), also implies it is strictly feasible. Hence, the duality gap between the primal-dual pair of (14) is zero.

Facial reduction of the dual. Using $\hat{x}$, we can regularize the dual problem to obtain a new primal-dual pair, where both the primal and dual problems have optimal values equal to the primal problem of (14):

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_1 = 1 \\ & x \in \mathbb{R}_+ \times \mathbb{R}^2 \end{array} \qquad \begin{array}{ll} \text{maximize} & y \\ \text{subject to} & \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} - \begin{pmatrix} y \\ 0 \\ 0 \end{pmatrix} = s \\ & s \in \mathbb{R}_+ \times \{0\}^2. \end{array} \tag{15}$$

For this primal-dual pair, relative interior solutions of the homogeneous model find the primal improving ray $x = (0, 0, -1)$. Since we have already established primal feasibility, we can conclude the primal of (15) is unbounded. Nevertheless, to complete the unboundedness certificate of Definition 6, we need a feasible point. To find this, we solve a primal feasibility problem.

The primal feasibility problem. Fixing the primal objective of (14) to zero gives:

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & x_1 = 1 \\ & x \in \mathcal{Q}_r^3 \end{array} \qquad \begin{array}{ll} \text{maximize} & y \\ \text{subject to} & \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y \\ 0 \\ 0 \end{pmatrix} = s \\ & s \in \mathcal{Q}_r^3. \end{array}$$

Taking $x = (0, 1, 0)$, $s = (0, 0, 0)$ and $y = 0$ yields a primal-dual optimal solution, and a feasible point x to both the primal of (14) and (15). This completes the unboundedness certificate for the primal of (15).

6 Conclusions

We have unified the facial reduction algorithm of Borwein and Wolkowicz with the self-dual embedding technique of Goldman and Tucker, bringing together both techniques to, in principle, solve arbitrary conic optimization problems over non-empty, closed, convex cones. Our method assumes one can produce relative interior solutions to homogeneous models. Such points are found from relative interior solutions to extended embeddings which, in the case of semidefinite programming, are produced by central-path-following techniques. Indeed, for SDP, implementing our method involves only conceptually-simple modifications to solvers such as SeDuMi [23], and these modifications only affect solver execution when both complementary solutions and improving rays do not exist. Addressing practical issues of numerical accuracy is a topic for future research, as is formal complexity analysis.

References

[1] R. A. Abrams. Projections of convex programs with unattained infima. *SIAM Journal on Control*, 13(3):706–718, 1975.

[2] E. Andersen, C. Roos, and T. Terlaky. Notes on Duality in Second Order and p-Order Cone Optimization. *Optimization*, 51(4):627–643, 2002.

[3] D. P. Bertsekas. *Convex optimization theory.* Athena Scientific Belmont, MA, 2009.

[4] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar. *Convex analysis and optimization.* Athena Scientific Belmont, 2003.

[5] J. Borwein and H. Wolkowicz. Regularizing the abstract convex program. *Journal of Mathematical Analysis and Applications*, 83(2):495–530, 1981.

[6] Y.-L. Cheung, S. Schurr, and H. Wolkowicz. Preprocessing and regularization for degenerate semidefinite programs. In *Computational and Analytical Mathematics*, pages 251–303. Springer, 2013.

[7] E. de Klerk, C. Roos, and T. Terlaky. Infeasible-start semidefinite programming algorithms via self-dual embeddings. *Fields Institute Communications*, 18:215–236, 1998.

[8] A. J. Goldman and A. W. Tucker. Theory of linear programming. *Linear inequalities and related systems*, 38:53–97, 1956.

[9] M. Halická, E. de Klerk, and C. Roos. On the convergence of the central path in semidefinite optimization. *SIAM Journal on Optimization*, 12(4):1090–1099, 2002.

[10] M. Liu and G. Pataki. Exact duals and short certificates of infeasibility and weak infeasibility in conic linear programming. 2015.

[11] B. F. Lourenço, M. Muramatsu, and T. Tsuchiya. Solving SDP completely with an interior point oracle. 2015.

[12] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Duality and self-duality for conic convex programming. Technical Report 9620/A, Econometric Institute, Erasmus University Rotterdam, 1996.

[13] Y. Nesterov. Infeasible start interior-point primal-dual methods in nonlinear programming. CORE Discussion Papers 1995067, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 1995.

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

[15] G. Pataki. Strong duality in conic linear programming: facial reduction and extended duals. *Computational and Analytical Mathematics*, pages 613–634, 2013.

[16] F. Permenter and P. A. Parrilo. Basis selection for SOS programs via facial reduction and polyhedral approximations. In *Proceedings of the IEEE Conference on Decision and Control*, 2014.

[17] F. Permenter and P. A. Parrilo. Partial facial reduction: simplified, equivalent SDPs via inner approximations of the PSD cone. http://arxiv.org/abs/1408.4685, 2014.

[18] F. A. Potra and R. Sheng. On homogeneous interrior-point algorithms for semidefinite programming. *Optimization Methods and Software*, 9(1-3):161–184, 1998.

[19] M. V. Ramana. An exact duality theory for semidefinite programming and its complexity implications. *Mathematical Programming*, 77(1):129–162, 1997.

[20] M. V. Ramana, L. Tunçel, and H. Wolkowicz. Strong duality for semidefinite programming. *SIAM Journal on Optimization*, 7(3):641–662, 1997.

[21] J. Renegar. Incorporating Condition Measures into the Complexity Theory of Linear Programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995. ISSN 1052-6234. doi: 10.1137/0805026.

[22] R. T. Rockafellar. *Convex analysis*, volume 28. Princeton University Press, 1997.

[23] 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.

[24] J. F. Sturm. Error bounds for linear matrix inequalities. *SIAM Journal on Optimization*, 10 (4):1228–1248, 2000.

[25] H. Waki and M. Muramatsu. Facial reduction algorithms for conic optimization problems. *Journal of Optimization Theory and Applications*, pages 1–28.

[26] Y. Ye, M. J. Todd, and S. Mizuno. An $\mathcal{O}(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research*, 19(1):53–67, 1994.

Appendix D

[Friberg 38]
A relaxed-certificate facial reduction algorithm based on subspace intersection

A relaxed-certificate facial reduction algorithm based on subspace intersection

Henrik A. Friberg[a,b]

[a] *Department of Wind Energy, Technical University of Denmark, DK-4000 Roskilde, Denmark.*
[b] *MOSEK ApS, Fruebjergvej 3, DK-2100 Copenhagen O, Denmark.*

Abstract

A "facial reduction"-like regularization algorithm is established for conic optimization problems by relaxing requirements on the reduction certificates. It requires only a linear number of reduction certificates from a constant time-solvable auxiliary problem, but is challenged by representational issues of the exposed reductions. A condition for representability is presented, analyzed for Cartesian product cones, and shown satisfiable for all exposed reductions of a single second-order cone. Should the representational condition fail at any iteration, a partially regularized problem is still obtained. Work on representing the exposed reductions, i.e., subspace intersections of conic sets, is ongoing.

Keywords: Facial reduction, subspace intersection, conic optimization, second-order cones.

1. Introduction

For $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$, consider the following primal-dual pair of conic optimization problems over the non-empty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^n$ and its dual cone $\mathcal{K}^* \subseteq \mathbb{R}^n$:

$$\begin{aligned} \theta_P = \text{infimum} \quad & c^T x \\ \text{subject to} \quad & Ax = b, \\ & x \in \mathcal{K}, \end{aligned} \tag{P}$$

$$\begin{aligned} \theta_D = \text{supremum} \quad & b^T y \\ \text{subject to} \quad & c - A^T y = s, \\ & s \in \mathcal{K}^*,\ y \in \mathbb{R}^m, \end{aligned} \tag{D}$$

where $\theta_P, \theta_D \in \mathbb{R}\cup\{-\infty, +\infty\}$ is the (possibly unattained) optimal values of (P) and (D) respectively. In contrast to linear optimization, valid reformulations of (P) may change the feasible set and optimal value of (D), and vice versa. This is manifested by the fact that whenever strong duality fails (i.e., $\theta_P > \theta_D$), it is possible to reformulate (P) (resp. (D)) such that its dual changes to establish strong duality. Such a reformulation is revealed by the facial reduction algorithms of [2, 8, 10].

Facial reduction algorithms progress iteratively driven by facial reduction certificates which are solutions of conic auxiliary problems [3, 6]. Alternatives to these certificates can also be used however. A remark in [3] notes that, by restricting the certificate definition, only a single certificate is needed for regularization albeit this being difficult to compute. In contrast, this paper relax the certificate definition to obtain a trivially solvable auxiliary problem from which a linear number of certificates is needed for regularization.

This relaxed-certificate facial reduction algorithm is presented in Section 3. Executing it on the primal problem (P), gives an equivalent problem with $\mathcal{K}$ replaced by the subspace intersected conic set

$$\mathcal{K} \cap \operatorname{span}(z_1, \ldots, z_k)^\perp = \mathcal{K} \cap z_1^\perp \cap \ldots \cap z_k^\perp, \tag{1}$$

where k is the number of iterations. This is like any other facial reduction algorithm except that subspace intersected conic set may not be a face of $\mathcal{K}$. Hence, to utilize this algorithm, classic representational results (e.g., [4] for faces of the semidefinite cone) have to be extended to non-facially exposing subspace intersections. Section 4 formalizes a representational condition for (1) to be supported by standard optimization software, and analyses its extendability from a single cone to Cartesian products. The condition is then satisfied by characterizing all subspace intersections of a second-order cone in Section 5. Note that cones for which all subspace intersections have not been characterized still allow for partial regularization by excluding all problematic exposing vectors z_i from (1).

2. Preliminaries

The image of a set under a function, i.e., $f(\mathcal{C}) := \{f(x) : x \in \mathcal{C}\}$, is used heavily. A subset $\mathcal{C} \subseteq \mathbb{R}^n$ is thus a *cone* if $\lambda\mathcal{C} = \mathcal{C}$ for any $\lambda > 0$, and the *dual cone* of $\mathcal{C}$ is denoted and defined by $\mathcal{C}^* := \{y \in \mathbb{R}^n : y^T\mathcal{C} \subseteq \mathbb{R}_+\}$. This paper is limited to non-empty, closed, convex cones, whereby $\mathcal{C}$ equals $(\mathcal{C}^*)^*$ and contains the origin [9].

A *face* $\mathcal{F}$ of $\mathcal{C}$, denoted $\mathcal{F} \trianglelefteq \mathcal{C}$, is a subset for which any line segment in $\mathcal{C}$, with a midpoint in $\mathcal{F}$, has both endpoints in $\mathcal{F}$ [9]. This generalizes extreme points and other faces of polyhedra. A *proper face* of $\mathcal{C}$ is a face which is non-empty and not equal to $\mathcal{C}$.

Email address: metronware@gmail.com (Henrik A. Friberg)

A *subspace intersection* of $\mathcal{C}$ is the intersection of a linear subspace and $\mathcal{C}$. Let $z^\perp := \{x \in \mathbb{R}^n : x^T z = 0\}$. For $z \in \mathcal{C}^*$, the subspace intersection $\mathcal{C} \cap z^\perp$ contains the origin and is a face of $\mathcal{C}$ as it maximizes $-z^T x$ over $x \in \mathcal{C}$ [9]. Hence, if $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$, then $\mathcal{C} \cap z^\perp$ is a proper face of $\mathcal{C}$.

A *facial reduction* of $\mathcal{K}$ in (P), is a proper face-revealing subspace intersection $\mathcal{K} \cap z^\perp$ for which $z^T x = 0$ is valid in (P). Similarly, a facial reduction of $\mathcal{K}^*$ in (D), is a proper face-revealing subspace intersection $\mathcal{K}^* \cap z^\perp$ for which $z^T s = 0$ is valid in (D). Note, if these equations $z^T x = 0$ (resp. $z^T s = 0$) is implied by $Ax = b$ in (P) (resp. $c - A^T y = s$ in (D)), then z is called a *facial reduction certificate*. Namely, taking (D) as example, $z^T s = 0$ is implied by $c - A^T y = s$ if and only if $z^\perp \supseteq \{c - A^T y : y \in \mathbb{R}^m\}$ (i.e., $z^\perp$ is a relaxation of the feasible set of s), if and only if $c^T z = 0$ and $A^T z = 0$, in agreement with [3, 6].

3. Primal regularization in $\mathcal{O}(m)$ time

Clearly, any equation of the form $z^T x = 0$, valid in (P) for some $z \in \mathbb{R}^n \setminus \{0\}$, justify reformulation from cone $\mathcal{K}$ to the subspace intersected cone $\mathcal{K} \cap z^\perp$. This is hence denoted a *valid subspace intersection* of $\mathcal{K}$ exposed by z, although validity is only with respect to the considered problem (P) and may change its dual. The task of finding such exposing vectors is the *auxiliary problem* of (P) for the relaxed-certificate facial reduction algorithm, and turns out to be trivially solvable.

Proposition 1 (Solving the auxiliary problem)**.** *Consider the m rows of $Ax = b$ from* (P)*. An equation of the form $z^T x = 0$ is found by $z^T := \omega^T A$ for the row weighting*

$$(\omega_1, \omega_2, \omega_{3:m}) = \begin{cases} (1,0,0) & \text{if } b_1 = 0, \\ (0,1,0) & \text{if } b_2 = 0, \\ (b_1^{-1}, -b_2^{-1}, 0) & \text{otherwise}, \end{cases}$$

chosen to satisfy $\omega^T b = 0$. If $z = \omega^T A = 0$, then a linear dependent row is found in $Ax = b$ and removing it leads to another row weighting. Otherwise, z is the exposing vector of a valid subspace intersection in (P) *by definition.*

With slight abuse of notation, Proposition 1 only fails to solve the auxiliary problem for (P) if $m = 0$ or $m = 1$ and $b_1 \neq 0$. In these cases there can be no facial reduction certificates for (P) by definition, implying the problem is regularized in the usual facial reduction sense. Conversely, when it succeeds, an exposing vector z is returned where $z^T x = 0$ is implied by $Ax = b$. The reformulation from $\mathcal{K}$ to $\mathcal{K} \cap z^\perp$ is thus valid and acts to enforce $z^T x = 0$ implicitly. That is, any one of the rows with $\omega_i \neq 0$ is made redundant. Removing it and repeating, leads to Algorithm 1. After k iterations, k rows is removed and the subspace intersected cone has been reformulated to $\mathcal{K} \cap z_1^\perp \cap \ldots \cap z_k^\perp = \mathcal{K} \cap \operatorname{span}(z_1, \ldots, z_k)^\perp$. The algorithm terminates when Proposition 1 fails to solve the auxiliary problem, that is, in $\mathcal{O}(m)$ iterations.

Algorithm 1: Primal regularization through the use of valid subspace intersections.

```
Let k ← 1
repeat
    Let z_k ← A^T ω be the solution to the auxiliary
    problem obtained from of Proposition 1.
    Remove the i'th row of Ax = b for some ω_i ≠ 0.
    Increment k ← k + 1.
until a solution z_k is not found by Proposition 1;
K ← K ∩ span(z_1, ..., z_k)^⊥.
```

A similar procedure can be made to regularize (D) at a slightly higher computation cost. In particular, Gaussian elimination can be carried out on the matrix $\left(c, \ -A^T \right)$ in order to systematically find weightings z of rows from this matrix that cancel out. That it, weightings z of rows from $c - A^T y = s$ of the form $z^T s = 0$.

What remains to be shown is that the rapidly obtained subspace intersected cones are useful in practice. This, however, remains obstructed by representational issues.

4. Representational issues

To solve the primal-dual pair (P) and (D) efficiently, a certain amount of information is needed about the cones $\mathcal{K}$ and $\mathcal{K}^*$. Good barrier functions are for example needed to deploy a primal-dual interior-point method [7]. Hence, when applying valid subspace intersections to the cones of a problem, it is important to ensure that optimization can also take place over the reduced cones.

The following definition materialize this property by giving a sufficient condition for the intersection $\mathcal{K} \cap z^\perp$ to be representable within the same class of cones as $\mathcal{K}$. This class, formally denoted Ω, is arbitrary but could be the symmetric cones for which optimization is efficient.

Definition 1. *Let $z \in \mathbb{R}^n$ and suppose* (P) *and* (D) *has cones $\mathcal{K}$ and $\mathcal{K}^*$ of class Ω. For some cones $\hat{\mathcal{K}}$ and $\hat{\mathcal{K}}^*$ of class Ω, and a matrix $H \in \mathbb{R}^{n \times r}$ for $1 \leq r \leq n$, a subspace intersection in* (P) *is said to be Ω-representable if*

$$\mathcal{K} \cap z^\perp = H\hat{\mathcal{K}}; \tag{2}$$

and a subspace intersection in (D) *is Ω^n-representable if*

$$s \in \mathcal{K}^* \cap z^\perp \quad \Leftrightarrow \quad H^T s \in \hat{\mathcal{K}}^*. \tag{3}$$

These conditions allow subspace intersections without leaving the chosen class of cones as claimed.

Proposition 2. *When the representational requirement of* (2) *or* (3) *is satisfied in Definition 1, the reduced problem can be represented by the primal-dual pair:*

$$\hat{\theta}_P = \inf_{\hat{x}} \{(H^T c)^T \hat{x} : (AH)\hat{x} = b, \, \hat{x} \in \hat{\mathcal{K}}\}, \tag{$\hat{P}$}$$

$$\hat{\theta}_D = \sup_{\hat{s}, \hat{y}} \{b^T \hat{y} : (H^T c) - (AH)^T \hat{y} = \hat{s}, \, \hat{s} \in \hat{\mathcal{K}}^*\}. \tag{$\hat{D}$}$$

Moreover, if the subspace intersection is a valid for the considered problem, the following statements hold:

1. *If* (2) *is satisfied, the problems* (P) *and* $(\hat{P})$ *have equal optimal values and equivalent feasible sets related by* $x = H\hat{x}$*;*
2. *If* (3) *is satisfied, the problems* (D) *and* $(\hat{D})$ *have equal optimal values and equivalent feasible sets related by* $y = \hat{y}$ *and* $s = c - A^T\hat{y}$*.*

Proof. $(\hat{P})$ and $(\hat{D})$ forms a primal-dual pair. Hence, the claims follow since (2) leads to $(\hat{P})$ as seen by

$$\begin{aligned}\theta_P &= \inf_x\{c^Tx : Ax = b,\, x \in \mathcal{K} \cap z^\perp\},\\ &= \inf_x\{c^Tx : Ax = b,\, x \in H\hat{\mathcal{K}}\},\\ &= \inf_x\{c^T(H\hat{x}) : A(H\hat{x}) = b,\, \hat{x} \in \hat{\mathcal{K}}\},\end{aligned}$$

and (3) leads to $(\hat{D})$ as seen by

$$\begin{aligned}\theta_D &= \sup_{s,y}\{b^Ty : c - A^Ty = s,\, s \in \mathcal{K}^* \cap z^\perp\},\\ &= \sup_{s,y}\{b^Ty : c - A^Ty = s,\, H^Ts \in \hat{\mathcal{K}}^*\},\\ &= \sup_{s,y}\{b^Ty : H^T(c - A^Ty) = \hat{s},\, \hat{s} \in \hat{\mathcal{K}}^*\}.\end{aligned}$$

□

Suppose the conditions of Definition 1 can be satisfied for all subspace intersections of two cones, $\mathcal{K}_1$ and $\mathcal{K}_2$, respectively. Even then, only a subset of all subspace intersections of $\mathcal{K}_1 \times \mathcal{K}_2$ may necessarily satisfy the conditions. This is a corollary (generalizing [10, Lemma 2.9] to non-facially exposing vectors) of the following proposition applied later, concerning the split of certificates.

Proposition 3. *Suppose* $z = \sum_{j=1}^k z_j \in \mathbb{R}^n$ *is an exposing vector for a valid subspace intersection of* $\mathcal{K} \subseteq \mathbb{R}^n$*. Then* $\mathcal{K}$ *can be reduced by each addend independently,*

$$\mathcal{K} \cap z^\perp = \mathcal{K} \cap z_1^\perp \cap \ldots \cap z_k^\perp,$$

if and only if one of the following conditions hold:

1. $\mathcal{K} \cap z_j^\perp \trianglelefteq \mathcal{K}$ *for all addends where either*

$$\begin{aligned} &z_j \in \mathcal{K}^*, && \text{for all } j \in \{1,\ldots,k\}, \quad \text{or}\\ &z_j \in -\mathcal{K}^*, && \text{for all } j \in \{1,\ldots,k\},\end{aligned}$$

   such that $\mathcal{K} \cap z^\perp = \mathcal{K} \cap (-z)^\perp \trianglelefteq \mathcal{K}$*;*
2. $\mathcal{K} \cap z_j^\perp = \mathcal{K}$ *for all but one addend. That is,*

$$z_j \in \mathcal{K}^\perp, \quad \text{for all } j \in \{1,\ldots,k\} \setminus \{i\},$$

   where $i \in \{1,\ldots,k\}$*.*

Proof. For any $z \in \mathbb{R}^n$, the subspace intersection contains the addend-wise subspace intersection. That is,

$$\begin{aligned}\mathcal{K} \cap z^\perp &= \{x \in \mathcal{K} : x^Tz = 0\},\\ &\supseteq \{x \in \mathcal{K} : x^Tz_j = 0 \text{ for } j = 1,\ldots,k\},\\ &= \mathcal{K} \cap z_1^\perp \cap \ldots \cap z_k^\perp.\end{aligned}$$

Equality holds if and only if $x^Tz = 0$ implies $x^Tz_j = 0$ for all $j \in \{1,\ldots,k\}$ over the set $x \in \mathcal{K}$. As $x^Tz = \sum_{j=1}^k x^Tz_j$, this requires all terms x^Tz^j to have the same sign (statement 1), or that only a single term x^Tz^j can take nonzero values (statement 2). □

Corollary 1. *Suppose* $z = (z^1,\ldots,z^k) \in \mathbb{R}^n$ *is an exposing vector for a valid subspace intersection of* $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_k \subseteq \mathbb{R}^n$*. Then* $\mathcal{K}$ *can be reduced in each Cartesian factor independently,*

$$\mathcal{K} \cap z^\perp = \left(\mathcal{K}_1 \cap (z^1)^\perp\right) \times \cdots \times \left(\mathcal{K}_k \cap (z^k)^\perp\right),$$

if and only if $\mathcal{K} \cap z^\perp \trianglelefteq \mathcal{K}$ *or* $\mathcal{K}_j \cap (z^j)^\perp = \mathcal{K}_j$ *for all but one Cartesian factor of* $\mathcal{K}$*.*

Proof. Define $z_j^T = (0,\ldots,0,(z^j)^T,0,\ldots,0)$, nonzero only on the support of K_j, and use Proposition 3. □

The consequences of Corollary 1 is a setback for general purpose usage of Algorithm 1. In particular, it leaves all non-facially exposing subspace intersections of $\mathcal{K}_1 \times \mathcal{K}_2$, properly reducing both cones at the same time, to be characterized separately from those of $\mathcal{K}_1$ and $\mathcal{K}_2$. Whether this result can be improved for any particular class of cones is left open. Instead, a characterization of all subspace intersections of a second-order cone is now derived and formalized in Theorem 1. This is followed by an efficient subspace intersection shortcut in Theorem 2.

5. Subspace intersections of a second-order cone

The quadratic cone $\mathcal{Q}^n$ and rotated quadratic cone $\mathcal{Q}_r^n$ (see, e.g., [1]) is defined and closely related by

$$\begin{aligned}\mathcal{Q}^n &= \left\{x \in \mathbb{R}^n : x_1^2 \geq \textstyle\sum_{j=2}^n x_j^2, \text{ and } x_1 \geq 0\right\},\\ &= W\mathcal{Q}_r^n \text{ for } W = \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0\\ 0 & 0 & \mathrm{I}\end{pmatrix}.\end{aligned}$$

This relation means that the subspace intersections of one leads to the subspace intersections of the other.

Proposition 4. *The representational conditions for* $\mathcal{Q}^n$ *from Definition 1, implies the conditions for* $\mathcal{Q}_r^n$*:*

1. $\mathcal{Q}_r^n \cap z^\perp = W(\mathcal{Q}^n \cap (Wz)^\perp)$*;*
2. $s \in \mathcal{Q}_r^n \cap z^\perp \;\Leftrightarrow\; Ws \in \mathcal{Q}^n \cap (Wz)^\perp$*.*

Proof. The matrix W is orthogonal. Hence, $f(x) = Wx$ is injective such that $f(X \cap Y) = f(X) \cap f(Y)$ for all sets X and Y. Moreover, $W\mathcal{Q}_r^n = \mathcal{Q}^n$ and $Wz^\perp = (W^{-T}z)^\perp = (Wz)^\perp$. Finally, $WW = \mathrm{I}$ by symmetry. □

In the following derivation of subspace intersections for the quadratic cone $\mathcal{Q}^n$, many results follow by definition. Note that x_1 in this definition is sometimes called the radius entry, and $\hbar = x_{2:n}$ the hyperball entries, since the quadratic cone correspond to an $(n-1)$-dimensional hyperball with radius x_1 centered around the origin. First, the elimination of zero-valued entries is formalized.

Proposition 5. *Consider $(x_1, \hbar) \in \mathcal{Q}^n$.*

1. *Given $\hbar_i = 0$, the membership is equivalent to*
$$\begin{pmatrix} x_1 \\ \hbar_{1:(i-1)} \\ \hbar_{(i+1):n} \end{pmatrix} \in \mathcal{Q}^{n-1}.$$
2. *Given $x_1 = 0$, the membership is equivalent to*
$$\hbar \in \{0\}^{n-1}.$$

Next, the elimination of scaled duplicates is formalized by describing an aggregation of squares in the definition of $\mathcal{Q}^n$. In particular, scaled hyperball-hyperball duplicates are eliminated by aggregating a sum of two squares, while scaled radiues-hyperball duplicates are eliminated by aggregating a difference of two squares.

Proposition 6. *Consider $(x_1, \hbar) \in \mathcal{Q}^n$.*

1. *Given $\hbar_i = \alpha\hbar_j$, where $j < i$ is assumed without loss of generality, the membership is equivalent to*
$$\begin{pmatrix} x_1 \\ \hbar_{1:(j-1)} \\ \sqrt{1+\alpha^2}\hbar_j \\ \hbar_{(j+1):(i-1)} \\ \hbar_{(i+1):n} \end{pmatrix} \in \mathcal{Q}^{n-1}.$$
2. *Given $\hbar_i = \alpha x_1$, the aggregation depends on α^2 and further allows use of Proposition 5.2 when $\alpha^2 \geq 1$.*
   (a) *For $\alpha^2 < 1$, the membership is equivalent to*
   $$\begin{pmatrix} \sqrt{1-\alpha^2}x_1 \\ \hbar_{1:(i-1)} \\ \hbar_{(i+1):n} \end{pmatrix} \in \mathcal{Q}^{n-1}.$$
   (b) *For $\alpha^2 = 1$, the membership is equivalent to*
   $$\begin{pmatrix} \hbar_{1:(i-1)} \\ \hbar_{(i+1):n} \end{pmatrix} \in \{0\}^{n-1} \quad \text{and} \quad x_1 \geq 0.$$
   (c) *For $\alpha^2 > 1$, the membership is equivalent to*
   $$\begin{pmatrix} \hbar_{1:(i-1)} \\ \sqrt{\alpha^2-1}\,x_1 \\ \hbar_{(i+1):n} \end{pmatrix} \in \{0\}^{n-1},$$
   where $x_1 \geq 0$ is redundant.

The applicability of these eliminations are greatly widened by the fact that the hyperball entries can be modified by orthogonal transformations [11].

Proposition 7. *The hyperball entries are invariant to orthogonal transformations. That is, given $H = H^{-T}$, then*
$$\left(\begin{smallmatrix} 1 & 0 \\ 0 & H \end{smallmatrix}\right) \mathcal{Q}^n = \mathcal{Q}^n.$$

The Householder matrix defines a particularly useful orthogonal transformation matrix able to rotate any nonzero vector to the main axis $\mathrm{e}_1 = (1, 0, \ldots, 0)^T \in \mathbb{R}^n$. This follows from the results of [5].

Proposition 8. *The Householder matrix solving $H\lambda = \|\lambda\|_2\mathrm{e}_1$ for nonzero $\lambda \in \mathbb{R}^n$ is given by*
$$\begin{aligned} H &= (\, \lambda/\|\lambda\|_2,\, V\,) = \begin{pmatrix} \lambda^T/\|\lambda\|_2, \\ V^T \end{pmatrix} = \mathrm{I} - 2uu^T, \\ u &= \begin{cases} \frac{\lambda/\|\lambda\|_2 - \mathrm{e}_1}{\|\lambda/\|\lambda\|_2 - \mathrm{e}_1\|_2} & \text{if } \lambda/\|\lambda\|_2 \neq \mathrm{e}_1, \\ 0 & \text{otherwise,} \end{cases} \end{aligned}$$
where H is symmetric and orthogonal by definition.

Finally, all results are in position to fully characterize the subspace intersections of the quadratic cone.

Theorem 1. *Suppose $z = (z_1, \lambda^T)^T \in \mathbb{R}^n$ is nonzero. Further, when $\lambda \neq 0$, consider $\alpha = -z_1/\|\lambda\|_2$ and the submatrix V of H from Proposition 8 solving $H\lambda = \|\lambda\|_2\mathrm{e}_1$. The conditions of Definition 1 is satisfied for* (P) *by:*

1. $\mathcal{Q}^n \cap z^\perp = \{0\}^n$, *for* $z_1^2 > \|\lambda\|_2^2$;
2. $\mathcal{Q}^n \cap z^\perp = \left(\begin{smallmatrix} 1 \\ \alpha\lambda/\|\lambda\|_2 \end{smallmatrix}\right) \mathbb{R}_+$, *for* $0 \neq z_1^2 = \|\lambda\|_2^2$;
3. $\mathcal{Q}^n \cap z^\perp = \begin{pmatrix} \frac{1}{\sqrt{1-\alpha^2}} & 0 \\ \frac{\alpha}{\sqrt{1-\alpha^2}}\lambda/\|\lambda\|_2 & V \end{pmatrix} \mathcal{Q}^{n-1}$, *for* $z_1^2 < \|\lambda\|_2^2$,

and for (D) *by:*

4. $x \in \mathcal{Q}^n \cap z^\perp \;\Leftrightarrow\; x \in \{0\}^n$, *for* $z_1^2 > \|\lambda\|_2^2$;
5. $x \in \mathcal{Q}^n \cap z^\perp \;\Leftrightarrow\; \begin{pmatrix} 1 & 0 \\ 0 & V^T \\ -\alpha & \lambda^T/\|\lambda\|_2 \end{pmatrix} x \in \mathbb{R}_+ \times \{0\}^{n-1}$, *for* $0 \neq z_1^2 = \|\lambda\|_2^2$;
6. $x \in \mathcal{Q}^n \cap z^\perp \;\Leftrightarrow\; \begin{pmatrix} \sqrt{1-\alpha^2} & 0 \\ 0 & V^T \\ -\alpha & \lambda^T/\|\lambda\|_2^2 \end{pmatrix} x \in \mathcal{Q}^{n-1} \times \{0\}$, *for* $z_1^2 < \|\lambda\|_2^2$.

Proof. If $\lambda = 0$ (a subcase of statement 1), the claim follows from Proposition 5-(2). Otherwise $\lambda \neq 0$, and the subspace intersections of $\mathcal{Q}^n$ are characterized by
$$\begin{aligned} \mathcal{Q}^n \cap z^\perp &= W\big((W\mathcal{Q}^n) \cap (Wz)^\perp\big), \\ &= W\left(\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2\mathrm{e}_1 \end{smallmatrix}\right)^\perp\right), \end{aligned}$$
for the symmetric and orthogonal matrix $W = \left(\begin{smallmatrix} 1 & 0 \\ 0 & H \end{smallmatrix}\right)$. This follows firstly by arguments for the proof of Proposition 4, and secondly by Proposition 7 and the definition of H. The set $\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2\mathrm{e}_1 \end{smallmatrix}\right)^\perp = \{x \in \mathcal{Q}^n : z_1x_1 + \|\lambda\|_2x_2 = 0\}$ is characterized below, and the claims follow from left-multiplication by W. If $z_1 = 0$ (a subcase of statement 3), the claim hence follows from Proposition 5-(1) as
$$\begin{aligned} \mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ |\lambda\|_2\mathrm{e}_1 \end{smallmatrix}\right)^\perp &= \{x \in \mathbb{R}^n : \left(\begin{smallmatrix} x_1 \\ x_{3:n} \end{smallmatrix}\right) \in \mathcal{Q}^{n-1},\ x_2 = 0\}, \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & \mathrm{I} \end{pmatrix} \mathcal{Q}^{n-1}. \end{aligned}$$

Otherwise $z_1 \neq 0$, and Proposition 6-(2) is used to eliminate the dependency $x_2 = \alpha x_1$ where $\alpha = -z_1/\|\lambda\|_2$. If $\alpha^2 > 1$ (the last of statement 1), the claim follows by
$$\begin{aligned} &\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2\mathrm{e}_1 \end{smallmatrix}\right)^\perp \\ &= \{x \in \mathbb{R}^n : \begin{pmatrix} \sqrt{\alpha^2-1}\,x_1 \\ x_{3:n} \end{pmatrix} \in \{0\}^{n-1},\ x_2 = \alpha x_1\}, \\ &= \{0\}^n. \end{aligned}$$

If $\alpha^2 = 1$ (statement 2), the claim follows by

$$\begin{aligned} &\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2 e_1 \end{smallmatrix}\right)^\perp \\ &\quad = \{x \in \mathbb{R}^n : x_{3:n} \in \{0\}^{m-2},\ x_1 \geq 0,\ x_2 = \alpha x_1\}, \\ &\quad = \begin{pmatrix} 1 \\ \alpha \\ \{0\}^{m-2} \end{pmatrix} \mathbb{R}_+. \end{aligned}$$

If $\alpha^2 < 1$ (the last of statement 3), the claim follows by

$$\begin{aligned} &\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2 e_1 \end{smallmatrix}\right)^\perp \\ &\quad = \{x \in \mathbb{R}^n : \left(\begin{smallmatrix} \sqrt{1-\alpha^2}\, x_1 \\ x_{3:n} \end{smallmatrix}\right) \in \mathcal{Q}^{n-1},\ x_2 = \alpha x_1\}, \\ &\quad = \begin{pmatrix} \frac{1}{\sqrt{1-\alpha^2}} & 0 \\ \frac{\alpha}{\sqrt{1-\alpha^2}} & 0 \\ 0 & \mathrm{I} \end{pmatrix} \mathcal{Q}^{n-1}. \end{aligned}$$

The dual statements 4-6, are characterized from the above derivations using that $x \in \mathcal{Q}^n \cap z^\perp$ is equivalent to

$$\begin{aligned} PWx \in PW(\mathcal{Q}^n \cap z^\perp) &= P\big((W\mathcal{Q}^n) \cap (Wz)^\perp\big), \\ &= P\left(\mathcal{Q}^n \cap \left(\begin{smallmatrix} z_1 \\ \|\lambda\|_2 e_1 \end{smallmatrix}\right)^\perp\right), \end{aligned}$$

for full rank matrices P and $W = \left(\begin{smallmatrix} 1 & 0 \\ 0 & H \end{smallmatrix}\right)$ as above. The statements are obtained using $P = \mathrm{I}$ for statement 1, $P = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & \mathrm{I} \\ -\alpha & 1 & 0 \end{pmatrix}$ for statement 2, and $P = \begin{pmatrix} \sqrt{1-\alpha^2} & 0 & 0 \\ 0 & 0 & \mathrm{I} \\ -\alpha & 1 & 0 \end{pmatrix}$ for statement 3. ☐

5.1. Efficient higher-dimensional subspace intersections

Consider the subspace intersection

$$\mathcal{Q}^n \cap \operatorname{span}(z_1, z_2, \ldots)^\perp = \mathcal{Q}^n \cap z_1^\perp \cap \ldots \cap z_k^\perp.$$

One may safely ignore any $z_j \in (\mathcal{Q}^n)^\perp = \{0\}$ and use Proposition 3 to aggregate, by summation, the subset of facially exposing vectors $z_j \in (\mathcal{Q}^n)^* = \mathcal{Q}^n$. Another technique, specific to the quadratic cone, allows several subspace intersections to be computed in a single step.

Theorem 2. *Let $\mathcal{Z} = \operatorname{span}(z_1, \ldots, z_k) \subseteq \mathbb{R}^n$ for nonzero $z_j = (0, \lambda_j^T)^T \in \mathbb{R}^n$, and consider the QR-decomposition with pivoting $(Q_1, Q_2)\left(\begin{smallmatrix} R_1 \\ 0 \end{smallmatrix}\right) = (\lambda_1, \ldots, \lambda_k)P$ for full row rank $R_1 \in \mathbb{R}^{r \times k}$. The conditions of Definition 1 is satisfied for* (P) *by:*

1. $\mathcal{Q}^n \cap \mathcal{Z}^\perp = \left(\begin{smallmatrix} 1 & 0 \\ 0 & Q_2 \end{smallmatrix}\right) \mathcal{Q}^{n-r}$,

and for (D) *by:*

2. $x \in \mathcal{Q}^n \cap \mathcal{Z}^\perp \ \Leftrightarrow\ \begin{pmatrix} 1 & 0 \\ 0 & Q_2^T \\ 0 & Q_1^T \end{pmatrix} x \in \mathcal{Q}^{n-r} \times \{0\}^r.$

Proof. The subspace reduction of $\mathcal{Q}^n$ is characterized by

$$\mathcal{Q}^n \cap \mathcal{Z}^\perp = W^T\big(\mathcal{Q}^n \cap (W\mathcal{Z})^\perp\big),$$

for nonsymmetric but orthogonal $W = \begin{pmatrix} 1 & 0 \\ 0 & Q_2^T \\ 0 & Q_1^T \end{pmatrix}$, following arguments for the proof of Proposition 4. Moreover, in terms of the column space operator $\mathcal{C}(\cdot)$, then

$$W\mathcal{Z} = W\mathcal{C}\left(\begin{smallmatrix} 0, & \ldots, & 0 \\ \lambda_1, & \ldots, & \lambda_k \end{smallmatrix}\right) = W\mathcal{C}\left(\begin{smallmatrix} 0 \\ Q_1 \end{smallmatrix}\right) = \mathcal{C}\left(\begin{smallmatrix} 0 \\ \left(\begin{smallmatrix} 0 \\ \mathrm{I} \end{smallmatrix}\right) \end{smallmatrix}\right),$$

by definition, where $Q_1^T Q_1 = \mathrm{I} \in \mathbb{R}^{r \times r}$ is the identity matrix. Hence, by Proposition 5-(1),

$$\mathcal{Q}^n \cap (W\mathcal{Z})^\perp = \mathcal{Q}^{n-r} \times \{0\}^r,$$

showing statement 1 after left-multiplication by W^T. Statement 2 is shown from the above derivation using that $x \in \mathcal{Q}^n \cap \mathcal{Z}^\perp$ is equivalent to $Wx \in \mathcal{Q}^n \cap (W\mathcal{Z})^\perp$. ☐

6. Final comments

The relaxed-certificate facial reduction algorithm shows a potential for rapid regularization. Nevertheless, it is likely doomed to partial regularization for all interesting applications unless the representational conditions of Definition 1 can be weakened. This is due to the consequences of Corollary 1. Finally, it remains unknown to what extend subspace intersections of other cones, such as the semidefinite cone, can be characterized. The reader is pointed to [4, Corollary 2.2.15] for a similar characterization of the facially exposing intersections of the semidefinite cone.

Acknowledgments. This paper was supervised by Erling D. Andersen, Mathias Stolpe and Frank Permenter. The author was funded by MOSEK ApS and the Danish Ministry of Higher Education and Science through the Industrial PhD project "Combinatorial Optimization over Second-Order Cones and Industrial Applications".

References

[1] F. Alizadeh and G. Goldfarb. Second-order cone programming. *Mathematical programming*, 95(1):3–51, 2003.

[2] J. Borwein and H. Wolkowicz. Regularizing the abstract convex program. *Journal of Mathematical Analysis and Applications*, 83(2):495–530, 1981.

[3] Y. Cheung, S. Schurr, and H. Wolkowicz. Preprocessing and Regularization for Degenerate Semidefinite Programs. In *Computational and Analytical Mathematics*. Springer New York, 2013.

[4] Y.-L. Cheung. *Preprocessing and Reduction for Semidefinite Programming via Facial Reduction: Theory and Practice*. PhD thesis, University of Waterloo, 2013.

[5] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

[6] B. F. Lourenço, M. Muramatsu, and T. Tsuchiya. Solving SDP Completely with an Interior Point Oracle. *Optimization Online*, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/06/4982.html`.

[7] Y. Nesterov and M. J. Todd. Primal-Dual Interior-Point Methods for Self-Scaled Cones. *SIAM Journal on Optimization*, 8 (2):324–364, 1998.

[8] G. Pataki. Strong duality in conic linear programming: facial reduction and extended duals. *Springer Proceedings in Mathematics and Statistics*, 50:613–634, 2013.

[9] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[10] H. Waki and M. Muramatsu. Facial reduction algorithms for conic optimization problems. *Journal of Optimization Theory and Applications*, 158(1):188–215, 2013.

[11] H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors. *Handbook of semidefinite programming: theory, algorithms, and applications*, volume 27 of *International Series in Operations Research & Management Science*. Springer US, 2000.

Appendix E

[Friberg 39]
Facial reduction heuristics and the motivational example of mixed-integer conic optimization

Facial reduction heuristics and the motivational example of mixed-integer conic optimization

Henrik A. Friberg *

February 24, 2016

Abstract

Facial reduction heuristics are developed in the interest of added performance and reliability in methods for mixed-integer conic optimization. Specifically, the process of branch-and-bound is shown to spawn subproblems for which the conic relaxations are difficult to solve, and the objective bounds of linear relaxations are arbitrarily weak. While facial reduction algorithms already exist to deal with these issues, heuristic variants represent a very potent supplement due to their inherent speed and accuracy. The paper covers a family of heuristics based on linear optimization, subgradient matching, single-cone analysis, and cone factorization.

1 Introduction

For $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$, consider the following primal-dual pair of conic optimization problems over the nonempty, closed, convex cone $\mathcal{K} \subseteq \mathbb{R}^n$ and its dual cone $\mathcal{K}^* \subseteq \mathbb{R}^n$:

$$(P): \quad \theta_P = \inf_x \{c^T x : Ax = b,\, x \in \mathcal{K}\}, \qquad (D): \quad \theta_D = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{K}^*\}. \tag{1}$$

By careful construction, the primal-dual pair can always be formulated such that the problem of interest, (P) or (D), is either *strongly feasible*[1] or *strongly infeasible*[2] [29, 41]. These properties serve in the interest of a successful solve in both theory and practice (see, e.g., [40, 20, 42]), and problems not satisfying either are denoted *ill-posed* following Renegar [35, 16].

Careful construction is unfortunately a false premise in many cases, such as when problem formulations are the product of preprocessing, cut generation or other automated modifications. As a result, ill-posed formulations may occur naturally in some of these cases (albeit perhaps rarely), and optimization software needs robust countermeasures for dealing with them to function reliably. This is the first reason for looking at the example of mixed-integer conic optimization, namely because it serves as a particularly great motivation for these countermeasures. Specifically, any solver would potentially fail from time to time if neglecting them, due to the ill-posed conic relaxations constructed by its own branch-and-bound procedure, as elaborated in Section 2.

Facial reduction algorithms [8, 32, 41] automate the process of careful construction by iteratively

*H. A. Friberg is employed at MOSEK ApS and affiliated with the Department of Wind Energy, Technical University of Denmark, DK-4000 Roskilde, Denmark. E-mail: metronware@gmail.com.

[1] Existence of a feasible point in the relative interior of all non-polyhedral cones (i.e., the generalized Slater's condition) qualifying use of the usual KKT-type optimality conditions [9].

[2] Existence of a dual improving ray qualifying use of the usual Farkas-type theorem of alternatives [29]; if this property is not satisfied for an infeasible problem, the infimum distance to feasibility is zero!

applying reformulations until either strong feasibility or strong infeasibility is satisfied. At every iteration until this happens, there exist so-called facial reduction certificates to verify and drive the process of these reformulations [41]. At termination, the problem is said to be *regularized*. Even so, the algorithm can still be continued in dual space as one way of dealing with unattainment [34, 28]. In any case, and for any of this to work in practice, core decisions have to be made regarding the generation of the facial reduction certificates.

First of all, the immediate conic formulation of the feasible set of certificates may itself be ill-posed and hence problematic to solve. Alternatives are given in [13] and [28], where the feasible set is lifted (for strong feasibility) by a nonnegative artificial variable which is then minimized. More recently, [34] shows that the self-dual embedding of (1) also contains the feasible set of certificates. Specifically, these certificates make up the optimal set if and only if (1) has no primal-dual optimal solutions or improving rays. Regularization and optimization can thus be interleaved with expected computational advantages (yet to be verified in practice), and the appealing property that facial reduction-induced reformulation is used only as needed.

This leads to the second point. If the certificates are generated by solving the conic optimization problems described above (from [13], [28] or [34]) using floating-point-based algorithms, the applied reformulations are only approximately valid. In fact, given the property of backwards stability shown in [13], this corresponds to reformulating a perturbed primal-dual pair (1). Given that there are many sources of inaccuracy in conic optimization software (see, e.g., [45, 4]), it may occur that the perturbation becomes too big for a continuation of the facial reduction algorithm to make sense. The very process of solving conic optimization problems—the size of the original problem—in each iteration of the facial reduction algorithm can, of course, also be very time consuming.

These concerns may to some extent be resolved by *facial reduction heuristics* attempting to construct accurate facial reduction certificates (or even exact, in rational arithmetic) within a short period of time. Specifically, with the right choice of heuristics (e.g., inspired by the application-specific reduction techniques of [14, 27, 11, 5, 47, 43]), heuristics alone may turn out to be sufficient in many cases. Partial regularization can also be useful, however, as shown by the example of mixed-integer conic optimization. Specifically, as elaborated in Section 2, objective bounds computed from linear relaxations can be arbitrarily weak if the conic relaxation is ill-posed, and only rapid regularization techniques make sense to include in such simple bound computations.

This paper introduces the relevance of facial reduction in mixed-integer conic optimization, and extends upon previous work on facial reduction heuristics. A family of heuristics based on *linear optimization*, inspired by [33], is presented in Section 4, along with strengthened versions of heuristics from [21, 13], denoted *subgradient matching* and *cone factorization*. This section also introduces a new heuristic based on *single-cone analysis*, inspired by the ability to solve single second-order cone problems analytically [19].

2 Motivational example: Mixed-integer conic optimization

Presolve, cut generation and branching are standard elements of mixed-integer optimization [1], and all act to modify either the feasible set of a problem or its formulation. This poses a risk for the procedure, as the conic relaxations solved in the nodes of the branch-and-bound search tree may become ill-posed at any time during the branch-and-bound algorithm.

This sudden occurrence of ill-posedness is illustrated to occur in [23] for the conic representation of the conditional constraint, $x \geq 4$ if $z = 0$, where many solvers are shown to give the wrong answer or produce errors. The conic relaxations in the search tree for this example are also unattained, however, motivating the following somewhat "cleaner" example where primal and dual optimal

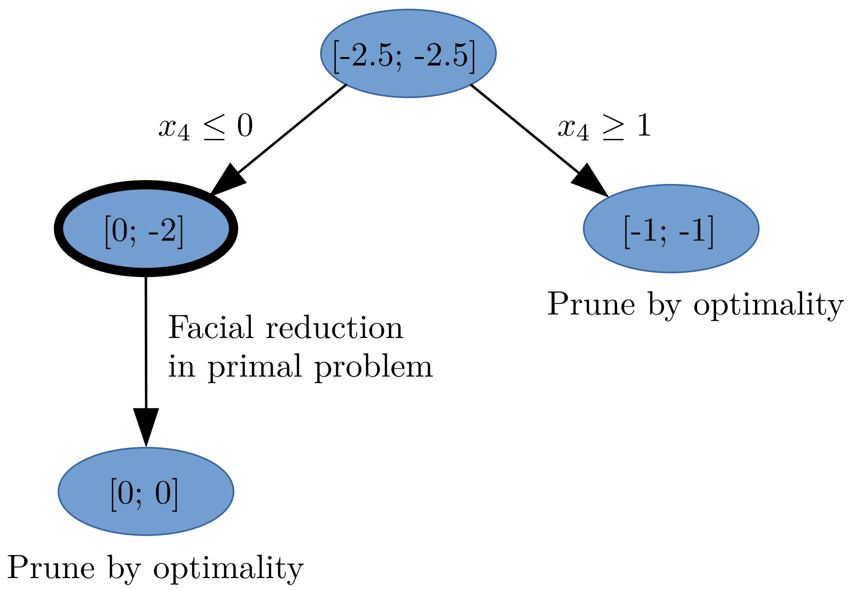


Figure 1: The branch-and-bound search tree for the problem of Example 1. Each node shows the range from the primal optimal value of the continuous relaxation to its dual optimal value.

values are attained throughout the search tree. The paper returns to analyze the conditional constraint of [23] in Section 4.3.

Example 1. *Consider the mixed-integer second-order cone optimization problem,*

$$\begin{array}{rclll} \theta_{MIP} & = & \inf\limits_{x} & 2x_3 + 2x_4 - x_5 & \\ & & \text{s.t.} & x_1 + x_2 - x_4 & \leq & 0, \\ & & & 4x_4 - x_5 & \geq & 0, \\ & & & x_3 & \geq & -1, \\ & & & x_5 & \leq & 1, \\ & & & x \in \mathcal{Q}^3 \times \mathbb{R}^2_+ & & \\ & & & x_4 \in \mathbb{Z}, & & \end{array} \tag{2}$$

where $\mathcal{Q}^n := \{x \in \mathbb{R}^n : x_1 \geq \|x_{2:n}\|_2\}$ *is the quadratic cone. Solving this problem gives* $\theta_{MIP} = -1$ *with the branch-and-bound search tree of Figure 1. The solutions to the node relaxations of this search tree can be found analytically as follows.*

root node *Assume the relaxation can be solved without the constraint* $x_1 + x_2 - x_4 \leq 0$*. Omitting it, two separate subproblems are obtained which can be solved by inspection, namely*

$$\begin{array}{ll} \inf\limits_{x_1,x_2,x_3} & 2x_3 \\ \text{s.t.} & x_3 \geq -1, \\ & (x_1, x_2, x_3) \in \mathcal{Q}^3, \end{array} \qquad \textit{and} \qquad \begin{array}{llll} \inf\limits_{x_4,x_5} & 2x_4 - x_5 & & \\ \text{s.t.} & 4x_4 - x_5 & \geq & 0, \\ & x_5 & \leq & 1, \\ & x_4, x_5 & \geq & 0. \end{array}$$

Solutions are, respectively, $(x_1, x_2, x_3) = (3, -2.75, -1)$ *and* $(x_4, x_5) = (0.25, 1)$*. Given that the omitted constraint is not violated by these values, the assumption was correct, and an optimal solution of the root node relaxation is given by* $x = (x_1, x_2, x_3, x_4, x_5)$ *with value* -2.5*. This is also the dual optimal value, since strong feasibility is certified by the identified solution, noting* $(x_1, x_2, x_3) \in \operatorname{relint} \mathcal{Q}^3$*, and implies zero duality gap [29].*

$[x_4 \geq 1]$**-node** *The same analysis as before applies with* $(x_4, x_5) = (1, 1)$ *being the new optimal solution of the second subproblem. Hence, since the omitted constraint is not violated, an optimal solution is given by* $x = (3, -2.75, -1, 1, 1)$ *with value* -1*, and this is also the dual optimal value.*

$[x_4 \leq 0]$**-node** *The till now omitted constraint can no longer be satisfied by optimal solutions of the separate subproblems. Instead, ocular inspection of* (2) *reveals the fixations* $x_4 = x_5 = 0$*, whereby this node relaxation reduces to the analytically solvable primal-dual pair,*

$$
\begin{array}{rcll}
\theta_P & = & \inf\limits_{x_1,x_2,x_3} & 2x_3 \\
 & & \text{s.t.} & x_1 + x_2 \leq 0, \\
 & & & x_3 \geq -1, \\
 & & & (x_1, x_2, x_3) \in \mathcal{Q}^3,
\end{array}
\qquad
\begin{array}{rcll}
\theta_D & = & \sup\limits_{s,y} & -y_2 \\
 & & \text{s.t.} & y_1 = -s_1, \\
 & & & y_1 = -s_2, \\
 & & & y_2 - 2 = -s_3, \\
 & & & y_1 \leq 0,\ y_2 \geq 0,\ s \in \mathcal{Q}^3.
\end{array}
\tag{3}
$$

Given $x_1 + x_2 \leq 0$*, let* $x_2 = -x_1 - t$ *for some* $t \geq 0$ *such that the conic constraint* $(x_1, x_2, x_3) \in \mathcal{Q}^3$ *becomes* $x_1^2 \geq (-x_1 - t)^2 + x_3^2$ *for* $x_1 \geq 0$*. This shows that* $t = 0$ *and* $x_3 = 0$*. Hence, the primal optimal value is* $\theta_P = 0$ *as attained, e.g., by* $(x_1, x_2, x_3) = (0, 0, 0)$*. Similarly, the conic constraint* $s \in \mathcal{Q}^3$ *becomes* $y_1^2 \geq y_1^2 + (y_2 - 2)^2$ *from which* $y_2 = 2$ *follows. The dual optimal value is hence* $\theta_D = -2$ *as attained, e.g., by* $s = (0, 0, 0)$ *and* $y = (0, 2)$*.*

The $[x_4 \leq 0]$*-node relaxation has positive duality gap and is therefore ill-posed. This ill-posedness is single-handedly caused by* $x_1 + x_2 \leq 0$*, as shown by* $x_1 + x_2 \geq 0$ *being a supporting halfspace [38] of* $(x_1, x_2, x_3) \in \mathcal{Q}^3$ *(compare to the unsatisfied definition of strong feasibility, requiring a feasible relative interior point of the cone). Hence, the conic relaxation is problematic to optimize without regularization and the method of facial reduction is now exemplified.*

$[x_4 \leq 0]$**-node using facial reduction** *The previous analysis of this node concluded* $x_4 = x_5 = 0$*, as well as* $x_1 + x_2 = t = 0$ *and* $x_3 = 0$ *for the primal-dual pair* (3)*. We can regularize the primal problem of* (3) *using either one of the latter two equations, albeit only the first one corresponds to a facial reduction. Indeed, taking* $x_1 + x_2 = 0$ *as example, it is possible to reformulate the conic constraint* $\left(\begin{smallmatrix} x_1 \\ x_2 \\ x_3 \end{smallmatrix}\right) \in \mathcal{Q}^3$ *as*

$$
\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in \mathcal{Q}^3 \cap \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}^{\perp}, \tag{4}
$$

because $\left(\begin{smallmatrix} x_1 \\ x_2 \\ x_3 \end{smallmatrix}\right) \in \left(\begin{smallmatrix} 1 \\ 1 \\ 0 \end{smallmatrix}\right)^{\perp}$ *is just another way of writing* $\left(\begin{smallmatrix} 1 \\ 1 \\ 0 \end{smallmatrix}\right)^T \left(\begin{smallmatrix} x_1 \\ x_2 \\ x_3 \end{smallmatrix}\right) = x_1 + x_2 = 0$*. In turn, given* $\mathcal{Q}^3 \cap \left(\begin{smallmatrix} 1 \\ 1 \\ 0 \end{smallmatrix}\right)^{\perp} = \left(\begin{smallmatrix} 1 \\ -1 \\ 0 \end{smallmatrix}\right) \mathbb{R}_+$ *(see [17]), this reformulation actually changes the conic constraint into the linear constraints* $x_1 \geq 0$*,* $x_2 = -x_1$ *and* $x_3 = 0$*. That is, the regularized conic relaxation becomes a linear relaxation for which strong duality always holds, in this case at value* $\theta_P = 0$*.*

Example 1 shows that branching operations can cause ill-posed relaxations to occur in the search tree of a branch-and-bound algorithm. In particular, by branching, the property of strong feasibility was lost when the feasible set was reduced in a way that kept the relaxation feasible without intersecting the relative interior of non-polyhedral cones. This specific type of reduction could as well have been caused by presolving or cut generation. Moreover, it is not even necessary for the non-polyhedral cone in question to be part of the original problem formulation, as it could be added later on as a conic cut [6, 25].

The implications of not being able to solve a relaxation is, in general, that there is a part of the search tree which cannot be pruned and hence prevent conclusions such as infeasibility or optimality for the mixed-integer problem. This fact motivates the integration of facial reduction techniques in the solution process of conic relaxations. Of course, turning attention to linear relaxations, these will not require regularization for solvability given that they can be solved symbolically assuming rational coefficients [26]. It will now be shown, however, that linear relaxations may need regularization for another reason, namely to strengthen the objective bounds they compute.

2.1 Objective bounds from linear relaxations

Cutting plane methods iteratively refining and resolving a linear relaxation of (P), formed by outer approximation of the non-polyhedral cones (e.g., [31]), converge with refinement towards the optimal value of (D)—not (P)!—and vice versa. That is, unless strong duality fails for the linear relaxation. Formally:

Proposition 1. *Let $\theta_P, \theta_D \in \mathbb{R} \cup \{-\infty, +\infty\}$ be the respective (and possibly unattained) optimal values of the primal-dual pair* (1)*, denoted (P) and (D). Suppose $\mathcal{C} \subseteq \mathbb{R}^n$ is a polyhedral set replacing $\mathcal{K}$ in (P) and consider the new primal-dual pair:*

$$\widehat{\theta}_P = \inf_x \{c^T x : Ax = b,\, x \in \mathcal{C}\}, \qquad \widehat{\theta}_D = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{C}^*\}. \tag{5}$$

Strong duality either fails for the primal-dual pair (5)*, or the following statements hold:*

1. *If $\mathcal{C} \supseteq \mathcal{K}$, then $\theta_P \geq \theta_D \geq \widehat{\theta}_D = \widehat{\theta}_P$;*
2. *If $\mathcal{C}^* \supseteq \mathcal{K}^*$, then $\widehat{\theta}_D = \widehat{\theta}_P \geq \theta_P \geq \theta_D$,*

Proof. If strong duality holds for the linear relaxation ($\widehat{\theta}_P = \widehat{\theta}_D$), the statements follow by weak duality in conic optimization ($\theta_P \geq \theta_D$). In particular, restricting the dual feasible set in Statement 1 (i.e., $\mathcal{C}^* \subseteq \mathcal{K}^*$) implies $\theta_D \geq \widehat{\theta}_D$, contra restricting the primal feasible set in Statement 2 which implies $\widehat{\theta}_P \geq \theta_P$. □

Remarkably, the subtlety of Proposition 1 is real as strong duality may fail for the primal-dual pair (5) even if it satisfied for the primal-dual pair (1). This is seen, e.g., by [34, Example 1] when the domain $x \in \mathbb{R}^2_+$ is taken to be an outer approximation of $x \in \{0\}^2$. Just as remarkable is the opposite case when strong duality holds for the linear relaxation. In particular, independently from the level of refinement used to construct the linear relaxation, the approximation error (as measured in terms of the optimal value) remains bounded by the duality gap of the conic relaxation.

Corollary 1. *Consider the primal-dual pairs* (1) *and* (5) *and suppose strong duality holds for* (5)*. The following statements hold:*

1. *If $\mathcal{C} \supseteq \mathcal{K}$, then $\theta_P - \widehat{\theta}_P \geq \theta_P - \theta_D$;*
2. *If $\mathcal{C}^* \supseteq \mathcal{K}^*$, then $\widehat{\theta}_D - \theta_D \geq \theta_P - \theta_D$,*

where $\theta_P - \theta_D \geq 0$ is the duality gap of the primal-dual pair (1)*.*

As the duality gap can be arbitrarily large for ill-posed conic relaxations (see, e.g., [40]), the objective bounds computed from linear relaxations can be arbitrarily weak. For branch-and-bound algorithms guided solely by the value of linear relaxations, this may cause a far greater number of nodes to be explored than actually needed. One remedy is to strengthen the objective bounds by using facial reduction algorithms. The expense alone, however, of checking for ill-posedness (by solving the conic optimization problems from [13], [28] or [34]), totally ruins the advantage of the linear relaxations which can be solved efficiently. This motivates the use of facial reduction heuristics to heuristically detect and fix ill-posedness and therethrough strengthen the objective bounds computed from linear relaxations.

3 Background

3.1 Cones, faces and facial reduction

Cones are subsets $\mathcal{C} \subseteq \mathbb{R}^n$ closed under positive scaling, i.e., $\lambda x \in \mathcal{C}$ holds for any $\lambda > 0$ and $x \in \mathcal{C}$. One such cone is the *dual cone* of any subset $\mathcal{C} \subseteq \mathbb{R}^n$, defined as $\mathcal{C}^* := \{y \in \mathbb{R}^n : y^T x \geq 0, \;\; \forall x \in \mathcal{C}\}$. This paper only concerns nonempty, closed, convex cones $\mathcal{C}$. In this case, the dual cone $\mathcal{C}^*$ is also a nonempty, closed, convex cone, $\mathcal{C}$ equals $(\mathcal{C}^*)^*$ and the origin is contained [36].

The conical hull of any set $S \subseteq \mathbb{R}^n$, denoted $\operatorname{cone}(S)$, is defined as the union of the origin and all finite conical combinations, $\lambda_1 s_1 + \ldots + \lambda_k s_k$ for $\lambda \in \mathbb{R}^k_{++}$, of points $s_1, \ldots, s_k \in S$. By definition, it is a nonempty, convex cone [36].

Polyhedral cones are conical hulls, $\operatorname{cone}(S)$, of any finite set of extreme rays S. By definition, it is a nonempty, closed, convex cone [36, Theorem 19.1].

Proper cones are nonempty, closed and convex, besides being *solid* and *pointed* [10]. Specifically, *solid cones* have nonempty interior, $\operatorname{int} \mathcal{C} \neq \emptyset$, or, equivalently, are full-dimensional in the sense that $\operatorname{span} \mathcal{C} = \mathbb{R}^n$ and $\mathcal{C}^\perp = (\operatorname{span} \mathcal{C})^\perp = \{0\}^n$. *Pointed cones*, on the other hand, contain no lines, i.e., $\mathcal{C} \cap (-\mathcal{C}) = \{0\}^n$.

Self-dual cones satisfy $\mathcal{C} = \mathcal{C}^*$. The self-dual and proper cones used in the examples of this paper are given by the nonnegative orthant $\mathbb{R}^n_+$, the quadratic cone $\mathcal{Q}^n := \{x \in \mathbb{R}^n : x_1 \geq \|x_{2:n}\|_2\}$, the rotated quadratic cone $\mathcal{Q}^n_r := \{(x_1, x_2, x_{3:n}) \in \mathbb{R}^2_+ \times \mathbb{R}^{n-2} \mid 2x_1 x_2 \geq \|x_{3:n}\|_2^2\}$, and the semidefinite cone $\mathcal{S}^N_+ := \left\{VV^T : V \in \mathbb{R}^{N \times N}\right\}$ (see, e.g., [44] for more properties).

Faces of a set $S \subseteq \mathbb{R}^N$ are subsets $\mathcal{F} \subseteq \mathcal{S}$ for which any line segment in $\mathcal{S}$, with a midpoint in $\mathcal{F}$, has both endpoints in $\mathcal{F}$ [36]. A *proper face* of $\mathcal{S}$ is a face which is nonempty and not equal to $\mathcal{S}$. These definitions generalize extreme points and other faces from polyhedra, and will now be related to the nonempty, closed, convex cones $\mathcal{C}$.

Let $z^\perp := \{x \in \mathbb{R}^n : x^T z = 0\}$. For $z \in \mathcal{C}^*$, the intersection $\mathcal{C} \cap z^\perp$ contains the origin and is a face of $\mathcal{C}$ as it maximizes $-z^T x$ over $x \in \mathcal{C}$ [36]. Hence, if $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$, the intersection $\mathcal{C} \cap z^\perp$ cannot equal $\mathcal{C}$ and hence represent a proper face of $\mathcal{C}$.

In these terms, a *facial reduction* can finally be defined as a valid problem reformulation in which a cone $\mathcal{C}$ is replaced by one of its proper faces, e.g., $\mathcal{C} \cap z^\perp$ for some $z \in \mathcal{C}^* \setminus \mathcal{C}^\perp$.

3.2 Facial reduction certificates

Consider again the primal-dual pair (1), restated here for the convenience of the reader:

$$(P): \quad \theta_P = \inf_x \{c^T x : Ax = b, \, x \in \mathcal{K}\}, \qquad (D): \quad \theta_D = \sup_{s,y} \{b^T y : c - A^T y = s, \, s \in \mathcal{K}^*\}.$$

where $\mathcal{K}, \mathcal{K}^* \subseteq \mathbb{R}^n$ are nonempty, closed, convex cones. Any equation of the form $z^T x = 0$, valid in (P) for some $z \in \mathcal{K}^* \setminus \mathcal{K}^\perp$, justifies the facial reduction from cone $\mathcal{K}$ to its proper face $\mathcal{K} \cap z^\perp$. If the equation is implied by the equation system $Ax = b$, formally denoted $z^\perp \supseteq \{x \in \mathbb{R}^n : Ax = b\}$, the exposing vector z is called a *facial reduction certificate* for (P) [34]. Similarly, the exposing vector $z \in \mathcal{K} \setminus (\mathcal{K}^*)^\perp$ of a facial reduction in (D), is called a facial reduction certificate for (D) if the required equation $z^T s = 0$ is implied by the equation system $c - A^T y = s$. The immediate conic formulation of the feasible set of facial reduction certificates for (P) and (D), respectively, can thus be stated as follows.

Proposition 2 ([34])**.** *The following statements hold:*

1. $z \in \mathbb{R}^n$ is a facial reduction certificate for (P) if the exists a $\omega \in \mathbb{R}^m$ for which

$$b^T\omega = 0, \qquad z = -A^T\omega, \qquad z \in \mathcal{K}^* \setminus \mathcal{K}^\perp,$$

and all facial reduction certificates are of this form if $\{x \in \mathbb{R}^n : Ax = b\} \neq \emptyset$.

2. $z \in \mathbb{R}^n$ is a facial reduction certificate for (D) if and only if

$$c^Tz = 0, \qquad Az = 0, \qquad z \in \mathcal{K} \setminus (\mathcal{K}^*)^\perp.$$

The conic formulations in statement 1 and 2 of Proposition 2 are called *auxiliary problems* of the primal-dual pair (1). Note that a constraint of the form $x \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp$ is satisfied if and only if $x \in \mathcal{C}$ and x has nonzero inner-product with any point in $\operatorname{relint} \mathcal{C}^*$. Hence, $x \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp$ can be reformulated as $x \in \mathcal{C}$ and $\hat{p}^Tx = 1$ for $\hat{p} \in \operatorname{relint} \mathcal{C}^*$ in the statements of Proposition 2, thereby normalizing the solutions to the homogeneous auxiliary problems.

3.3 Subgradient-based outer approximation

Let $\mathcal{C} \subseteq \mathbb{R}^n$ be a nonempty, closed, convex cone with membership indicator function

$$\chi_{\mathcal{C}}(x) = \begin{cases} 0 & \text{if } x \in \mathcal{C}, \\ +\infty & \text{otherwise.} \end{cases}$$

The indicator function is convex, since the set $\mathcal{C}$ is convex [36], and offers the useful description $C = \{x \in \mathbb{R}^n : \chi_{\mathcal{C}}(x) \leq 0\}$. In particular, valid inequalities of $\{x \in \mathbb{R}^n : \chi_{\mathcal{C}}(x) \leq 0\}$ are readily provided by the subgradient inequality

$$\chi_{\mathcal{C}}(\hat{x}) + \overline{\xi}^T(x - \hat{x}) \ \leq \ 0, \tag{6}$$

holding for any $\hat{x} \in \mathbb{R}^n$, and all subgradients $\overline{\xi}$ of the corresponding subdifferential set $\partial\chi_{\mathcal{C}}(\hat{x}) \subseteq \mathbb{R}^n$. This is a direct consequence of the inequality $\chi_{\mathcal{C}}(x) \leq 0$, combined with the fact that subgradients $\overline{\xi} \in \partial\chi_{\mathcal{C}}(\hat{x})$ are defined (e.g., in [36]) as solutions to

$$\chi_{\mathcal{C}}(\hat{x}) + \overline{\xi}^T(x - \hat{x}) \ \leq \ \chi_{\mathcal{C}}(x) \text{ for all } x \in \mathbb{R}^n. \tag{7}$$

For the subset of points $\hat{x} \notin \mathcal{C}$, the system (7) has no solution as $\chi_{\mathcal{C}}(\hat{x}) = +\infty$. For all other points, $\hat{x} \in \mathcal{C}$, the system is solved by $\overline{\xi} \in -\mathcal{C}^* \cap \hat{x}^\perp$ [36, Corollary 23.5.4]. This allows us to characterize the family of subgradient inequalities (6) in a much simpler way.

Theorem 1. *For nonempty, closed, convex cones $\mathcal{C} \subseteq \mathbb{R}^n$, the set obtained by intersecting all subgradient inequalities of the form (6), is concisely described by*

$$\xi^Tx \geq 0, \qquad \text{for } \xi \in \Omega,$$

where $\Omega \subseteq \mathbb{R}^n$ is any set satisfying $\operatorname{cone}(\Omega) = \mathcal{C}^*$.

Proof. The subdifferential set is the solutions to (7) given by

$$\partial\chi_{\mathcal{C}}(\hat{x}) = \begin{cases} \emptyset & \text{if } \hat{x} \notin \mathcal{C}, \\ -\mathcal{C}^* \cap \hat{x}^\perp & \text{otherwise,} \end{cases}$$

as argued. Hence, to satisfy $\overline{\xi} \in \partial\chi_{\mathcal{C}}(\hat{x})$ as needed in (6), both $\chi_{\mathcal{C}}(\hat{x}) = 0$ and $\overline{\xi}^T \hat{x} = 0$ is required. This simplifies (6) to $\overline{\xi}^T x \leq 0$, holding for any $\hat{x} \in \mathcal{C}$ and all $\overline{\xi} \in -\mathcal{C}^* \cap \hat{x}^{\perp}$. The claim is hence shown for $\Omega = \cup_{\hat{x}\in\mathcal{C}}(\mathcal{C}^* \cap \hat{x}^{\perp}) = \mathcal{C}^*$ (noting that $\mathcal{C}$ contains the origin), in terms of the negated subgradient $\xi = -\overline{\xi}$. Finally, if $\operatorname{cone}(\Omega) = \mathcal{C}^*$, then any $\xi \in \mathcal{C}^*$ has a finite conical factorization $\xi = \sum_{j=1}^{k} \lambda_j \xi_j^T$ for $\lambda \in \mathbb{R}^k_{++}$ and $\xi_j \in \Omega$. This shows $\xi^T x = \lambda_1 \xi_1^T x + \ldots + \lambda_k \xi_k^T x \geq 0$ to be redundant given $\xi_j^T x \geq 0$ for $j = 1, \ldots, k$. □

Needless to say, an outer approximation of $\mathcal{C}$ is obtained by Theorem 1 from any finite subset of $\Omega \subseteq \mathbb{R}^n$, or less pedantic, from any finite subset of $\mathcal{C}^*$.

Corollary 2. *A subgradient-based outer approximation of a nonempty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$ is given by $\hat{\Omega}^* = \{x : \xi^T x \geq 0$ for $\xi \in \hat{\Omega}\}$, that is $\hat{\Omega}^* \supseteq \mathcal{C}$, for any finite subset $\hat{\Omega} \subseteq \mathcal{C}^*$.*

The rather trivial corollary above does not exploit the distinction between Ω and $\mathcal{C}^*$ in Theorem 1, and may hence lead to formulations with redundancies. What Theorem 1 actually suggests is that Ω can be chosen as any minimal set of conically independent points generating $\mathcal{C}^*$ in the sense that $\operatorname{cone}(\Omega) = \mathcal{C}^*$. This notion can be clarified in case $\mathcal{C}^*$ is a pointed, nonempty, closed, convex cone not equal to $\{0\}^n$. Specifically, in this case, a necessary and sufficient condition for $\operatorname{cone}(\Omega) = \mathcal{C}^*$ is that Ω contains a relative interior point from all one-dimensional faces of $\mathcal{C}^*$. This follows by [36, Corollary 18.5.2.], noting that *extreme rays* are used to denote the set of half-line faces, i.e., the set of one-dimensional faces for pointed, nonempty, closed, convex cones. This leads to the following stricter definition of an outer approximation obtained by Theorem 1.

Corollary 3. *Suppose Ω is any minimal set of conically independent points generating $\mathcal{C}^*$ in the sense that* $\operatorname{cone}(\Omega) = \mathcal{C}^*$. *A subgradient-based outer approximation of a nonempty, closed, convex cone $\mathcal{C} \subseteq \mathbb{R}^n$ is then given by $\hat{\Omega}^* = \{x : \xi^T x \geq 0$ for $\xi \in \hat{\Omega}\}$ for any finite subset $\hat{\Omega} \subseteq \Omega$.*

Outer approximations on the form of Corollary 3 appears for second-order cones in [7], and for semidefinite cones in [30, 33]. These outer approximations are constructed either statically as in [33], or dynamically as in [7, 30] by iteratively separating violated points. This indicates that the implications of Theorem 1 are folklore, although never formalized to the author's knowledge. The paper returns to show the static outer approximations of [33] in the context of facial reduction heuristics in Section 4.1.

4 Facial reduction heuristics

The auxiliary problems of Proposition 2 describe the feasible set of facial reduction certificates for (P) and (D), respectively. Solving simplifications of these sets are hence a straightforward way to design facial reduction heuristics. This is done for the family of heuristics based on linear optimization in Section 4.1, and for the heuristics based on cone factorization in Section 4.4. Alternatively, one may search for equations of the form $z^T x = 0$ in (P) (or $z^T s = 0$ in (D)) and detect whenever the exposing vector $z \in \mathbb{R}^n$ exposes a proper face of the respective cone. This is done in the subgradient matching heuristic of Section 4.2 and in the single-cone analysis of Section 4.3.

4.1 A family of heuristics based on linear optimization

This family of facial reduction heuristics have in common that they solve the auxiliary problems of Proposition 2 using linear inner approximations of the conic variable domains. This allows the partially represented set of facial reduction certificates to be explored fairly efficient and to high accuracy using simplex methods. In fact, the use of simplex methods allow *exact certificates* in rational arithmetic to be computed for this family of heuristics, within reasonable computational efforts [26]. This motivates the use of heuristics based on linear optimization.

To begin, note that a partial set of facial reduction certificates is clearly obtained through inner approximation of the auxiliary problems in Proposition 2. One example of this is given in [33], where inner approximations of the conic variable domain are obtained from span-invariant outer approximations of the cone considered for facial reduction. Taking the auxiliary problem of Proposition 2-(1) as example, an inner approximation of the conic variable domain,

$$\mathcal{C}^* \setminus \mathcal{C}^\perp \subseteq \mathcal{K}^* \setminus \mathcal{K}^\perp,$$

clearly holds if $\mathcal{C}^* \subseteq \mathcal{K}^*$ and $\mathcal{C}^\perp \supseteq \mathcal{K}^\perp$ is satisfied. The former containment shows $\mathcal{C}$ to be an outer approximation of $\mathcal{K}$ by its contrapositive, $\mathcal{C} \supseteq \mathcal{K}$, and the latter containment establish span-invariance. In particular, the latter containment and its contrapositive, $\operatorname{span} \mathcal{C} \subseteq \operatorname{span} \mathcal{K}$, must hold with equality given $\mathcal{C} \supseteq \mathcal{K}$. This characterization leads to the following family of inner approximated auxiliary problems.

Proposition 3. *A partial set of facial reduction certificates, $z \in \mathbb{R}^n$, is obtained from any span-invariant outer approximation of the cone from the considered problem. Specifically:*

1. *Given $\mathcal{K} \subseteq \mathcal{C} \subseteq \mathbb{R}^n$ where $\operatorname{span} \mathcal{K} = \operatorname{span} \mathcal{C}$, a partial description of the facial reduction certificates for (P) is given by*

$$b^T\omega = 0, \qquad z = -A^T\omega, \qquad z \in \mathcal{C}^* \setminus \mathcal{C}^\perp.$$

2. *Given $\mathcal{K}^* \subseteq \mathcal{C}^* \subseteq \mathbb{R}^n$ where $\operatorname{span} \mathcal{K}^* = \operatorname{span} \mathcal{C}^*$, a partial description of the facial reduction certificates for (D) is given by*

$$c^Tz = 0, \qquad Az = 0, \qquad z \in \mathcal{C} \setminus (\mathcal{C}^*)^\perp.$$

Restricting attention to primal-dual pairs (1) where the cone considered for facial reduction is solid, the span-invariance of Proposition 3 is automatically satisfied. This makes it possible to apply the subgradient-based outer approximation of Corollary 2 directly. A partial description of the facial reduction certificates for (P) is thus obtained from Proposition 3-(1), using

$$\mathcal{C} = \mathcal{C}_1 \times \cdots \times \mathcal{C}_k, \text{ where } \mathcal{C}_j = \{x \in \mathbb{R}^n : \xi^T x \geq 0,\ \forall \xi \in \hat{\Omega}_j\}, \tag{8}$$

given any finite subset of negated subgradients, $\hat{\Omega}_j \subseteq \mathcal{K}_j^*$, for all factors of the Cartesian product $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_k$. It is worth noting that $C_j = \hat{\Omega}_j^*$ by definition and hence $\mathcal{C}_j^* = \operatorname{cone}(\hat{\Omega}_j)$ by the bipolar theorem [36]. Two examples of this inner approximation are now given, satisfying the stricter requirements of Corollary 3 on the selection of $\hat{\Omega}_j$.

Example 2. *The semidefinite cone $\mathcal{S}_+^N$ is outer approximated in accordance with Corollary 3, by a finite subset $\hat{\Omega}$ of rank-one matrices of the form $\omega\omega^T$ for $\omega \in \mathbb{R}^N$, representing the set of extreme rays for $\mathcal{S}_+^N$ [24]. Two examples from [33] are given by:*

1. *The non-negative diagonal approximation chooses ω as any permutation of $(1, 0, \ldots, 0)^T$. Hence, $\mathcal{C} = \{X \in \mathcal{S}^n : X_{ii} \geq 0\}$ and $\mathcal{C}^* = \{X \in \mathcal{S}^n : X_{ii} \geq 0,\ X_{ij} = 0 \text{ for } i \neq j\}$.*

2. *The diagonally-dominant approximation chooses ω as any permutation of either $(1, 0, \dots, 0)^T$, $(1, 1, 0, \dots, 0)^T$ or $(1, -1, 0, \dots, 0)^T$. Hence, $\mathcal{C} = \{X \in \mathcal{S}^n : X_{ii} \geq 0, X_{ii} + X_{jj} \geq 2|X_{ij}|\}$ and $\mathcal{C}^* = \{X \in \mathcal{S}^n : X_{ii} \geq \sum_{j \neq i} |X_{ij}|\}$.*

These realizations of $\mathcal{C}$ give rise to inner approximated auxiliary problems through Proposition 3, that are shown effective in [33] for a wide range of ill-posed semidefinite optimization instances.

With the right choice of $\hat{\Omega}_j$ in Corollary 2, any facial reduction certificate can be found using the subgradient-based inner approximated auxiliary problems of Proposition 3. The only catch is that the right choice of $\hat{\Omega}_j$ literally have to guess which facial reductions are possible, in order for this family of heuristics to recognize them.

Proposition 4. *Let $z = (z_1, \dots, z_k)$ be a facial reduction certificate for (P), where $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_k$ is a solid cone. Suppose further that this z is feasible in the auxiliary problem of Proposition 3-(1) for a cone $\mathcal{C} = \mathcal{C}_1 \times \cdots \times \mathcal{C}_k$ defined as in (8). Then $\mathcal{C}^* = \operatorname{cone}(\hat{\Omega}_1) \times \cdots \times \operatorname{cone}(\hat{\Omega}_k)$ for finite subsets $\hat{\Omega}_j \subseteq \mathcal{K}_j^*$, giving rise to finite conical factorizations $z = (\sum_{i=1}^{p_1} \lambda_1^i \xi_1^i, \dots, \sum_{i=1}^{p_k} \lambda_k^i \xi_k^i)$ for $\lambda_j^i > 0$ and nonzero $\xi_j^i \in \hat{\Omega}_j$. It now follows that the reformulation from $\mathcal{K}_j$ to $\mathcal{K}_j \cap (\xi_j^i)^\perp$ is a valid facial reduction in (P) for all $j = 1, \dots, k$ and $i = 1, \dots, p_j$.*

Proof. A facial reduction certificate for (P) satisfy $z \in \mathcal{K}^*$ by Proposition 2. Hence,

$$\mathcal{K} \cap z^\perp = (\mathcal{K}_1 \cap z_1^\perp) \times \cdots \times (\mathcal{K}_k \cap z_k^\perp)$$

as seen, e.g., by [17, Corollary 1]. In turn, each Cartesian factor can be rewritten as

$$\mathcal{K}_j \cap z_j^\perp = \mathcal{K}_j \cap (\textstyle\sum_{i=1}^{p_j} \lambda_j^i \xi_j^i)^\perp = \mathcal{K}_j \bigcap_{i=1}^{p_j} (\lambda_j^i \xi_j^i)^\perp = \mathcal{K}_j \bigcap_{i=1}^{p_j} (\xi_j^i)^\perp,$$

where the first equality follows by definition. The second equality is a consequence of [17, Proposition 3], given $\lambda_j^i \xi_j^i \in \mathcal{K}_j^*$ as implied by $\lambda_j^i > 0$ and $\xi_j^i \in \hat{\Omega}_j \subseteq \mathcal{K}_j^*$. The last equality is from invariance of the orthogonal complement to positive scaling. The claim hence follows from $\mathcal{K}_j \cap (\xi_j^i)^\perp$ being a relaxation of $\mathcal{K} \cap z^\perp$, and from noting that $\mathcal{K}_j \cap (\xi_j^i)^\perp$ defines a proper face of the solid cone $\mathcal{K}_j$. $\square$

The equivalent statement for the subgradient-based inner approximated auxiliary problem of Proposition 3-(2) is shown similarly. There is another and possibly more intuitive way to think of this. In particular, note that the partial set of certificates for the primal-dual pair (1), is the complete set of certificates for the linear primal-dual pair:

$$\widehat{\theta}_P = \inf_x \{c^T x : Ax = b,\, x \in \mathcal{C}\}, \qquad \widehat{\theta}_D = \sup_{s,y} \{b^T y : c - A^T y = s,\, s \in \mathcal{C}^*\}, \tag{9}$$

as seen by comparing Proposition 2 with Proposition 3. Hence, you have to guess the facial reductions in order to find them in (9), since all facial reductions of linear problems are inequalities holding as implied equalities [13] and the only inequalities of (9) are those with normal vectors from $\hat{\Omega}_j$. As example, if $\mathcal{C}$ is defined by the non-negative diagonal approximation of Example 2-(1), the only facial reductions that can be found are from diagonal entries fixed to zero. Simpler variable bound analysis may also yield these conclusions and hence might enjoy much of the success reported for the non-negative diagonal approximation in [33]. This motivates subgradient matching.

4.2 Subgradient matching

In contrast to the previous family of heuristics that require one to solve an optimization problem, albeit linear, the subgradient matching technique integrates with domain propagation to provide a faster facial reduction heuristic. This integration has synergistic effects too, even when no facial reductions are identified, as subgradient matching acts to strengthen the variable and constraint activity bounds derived through domain propagation as will be shown.

Consider the activity bounds of an affine expression $a^T x$ for some $a \in \mathbb{R}^n$, given by

$$L_{\min} = \sum_{j:a_j>0} a_j l_j + \sum_{j:a_j<0} a_j u_j \quad \text{and} \quad L_{\max} = \sum_{j:a_j>0} a_j u_j + \sum_{j:a_j<0} a_j l_j, \tag{10}$$

where $l, u \in (\mathbb{R} \cup \{-\infty, +\infty\})^n$ holds the domain propagated lower and upper variable bounds. These are called the simplest, but also the weakest, activity bounds by Savelsbergh [37]. A simple advancement of these bounds is to include subgradient information from the conic variable domain $x \in \mathcal{K}$ as shown in Algorithm 1. The correctness of this algorithm is proven in Proposition 5.

Algorithm 1: Computing activity bounds of $a^T x$ using subgradient matching.

Data: Suppose $x = (x^1, \ldots, x^k) \in \mathcal{K}_1 \times \cdots \times \mathcal{K}_k$, dictating similar partitioning of $a = (a^1, \ldots, a^k)$, $l = (l^1, \ldots, l^k)$ and $u = (u^1, \ldots, u^k)$.

1 **for all** $i \in \{1, \ldots, k\}$ **do**

2   Compute the simple activity bounds of $(a^i)^T x^i$:

$$L^i_{\min} \leftarrow \sum_{j:a^i_j>0} a^i_j l^i_j + \sum_{j:a^i_j<0} a^i_j u^i_j \quad \text{and} \quad L^i_{\max} \leftarrow \sum_{j:a^i_j>0} a^i_j u^i_j + \sum_{j:a^i_j<0} a^i_j l^i_j.$$

3   **if** $L^i_{\min} \leq 0$ **and** $a^i \in \mathcal{K}^*_i$ **then**

4   Update $L^i_{\min} \leftarrow 0$.

5   **end**

6   **if** $L^i_{\max} \geq 0$ **and** $a^i \in -\mathcal{K}^*_i$ **then**

7   Update $L^i_{\max} \leftarrow 0$.

8   **end**

9 **end**

10 **return** $L_{\min} \leftarrow \sum_{i=1}^k L^i_{\min}$ and $L_{\max} \leftarrow \sum_{i=1}^k L^i_{\max}$.

Proposition 5. *The activity bounds computed by Algorithm 1 are valid and stronger than those computed by* (10)*.*

Proof. If line 4 (resp. line 7) of Algorithm 1 never executes, then $L_{\min}$ (resp. $L_{\max}$) equals the value computed by (10). Otherwise, when line 4 executes, then $(a^i)^T x^i \geq 0$ holds by definition of dual cones and strengthens the current lower bound if $L^i_{\min} < 0$. Similarly, when line 7 executes, then $(a^i)^T x^i \leq 0$ holds and strengthens the current upper bound if $L^i_{\max} > 0$. ☐

The attentive reader may question the use of non-strict inequalities on line 4 and 7 of Algorithm 1, as the updating of $L^i_{\min}$ and $L^i_{\max}$ is needless if they are already zero. This is done intentionally, however, to stress that one should pay attention to all cases in which the subgradient matching bound is the tightest bound computed. In particular, whenever this holds, valid facial reductions can be identified from *forcing* constraints on the affine expression $a^T x$.

Definition 1. *Suppose* $\sum_{i=1}^{p} L^i = b$. *A constraint is considered forcing if*

1. $\sum_{i=1}^{p} (a^i)^T x^i \leq b$ *for lower bounded terms* $(a^i)^T x^i \geq L^i$ *for all* $i \in \{1, \ldots, p\}$;
2. $\sum_{i=1}^{p} (a^i)^T x^i \geq b$ *for upper bounded terms* $(a^i)^T x^i \leq L^i$ *for all* $i \in \{1, \ldots, p\}$;
3. $\sum_{i=1}^{p} (a^i)^T x^i = b$ *for lower bounded terms (resp. upper bounded terms) as above,*

whereby $(a^i)^T x^i = L^i$ *for all* $i \in \{1, \ldots, p\}$ *is implied.*

Forcing constraints appear already in [3], but the idea that they may be used to identify valid facial reductions is new. This idea is formalized in the following proposition.

Proposition 6. *Consider a forcing constraint from Definition 1. If any of the implied equations* $(a^i)^T x^i = L^i$ *holds with* $L^i = 0$ *and* $a^i \in \mathcal{K}^* \setminus \mathcal{K}^\perp$ *for some* $i \in \{1, \ldots, p\}$, *the facial reduction from* $x^i \in \mathcal{K}$ *to* $x^i \in \mathcal{K} \cap (a^i)^\perp$ *is justified.*

Comparing Proposition 6 to Algorithm 1, it can now be verified that line 4 and 7 captures the sought conditions for a facial reduction whenever $a^i \notin \mathcal{K}_i^\perp$. Note that this is equivalent to $a^i \neq 0$ for solid cones. In an attempt to evaluate the usefulness of subgradient matching without an implementation, the reader is invited to verify that it is capable of regularizing the $[x_4 \leq 0]$-node of Example 1. It can furthermore be shown to recognize all primal facial reductions of all examples in [34], including the following less trivial reduction.

Example 3. *Consider the primal-dual pair originating with [2], given by*

$$\begin{array}{ll} \inf\limits_{x} & x_3 \\ \text{s.t.} & x_1 + x_2 + x_4 + x_5 = 0, \\ & -x_3 + x_4 = 1, \\ & x \in \mathcal{Q}^3 \times \mathcal{Q}^2, \end{array} \qquad \begin{array}{ll} \sup\limits_{s,y} & y_2 \\ \text{s.t.} & \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_1 \\ -y_2 \\ y_1 + y_2 \\ y_1 \end{pmatrix} = s, \\ & s \in \mathcal{Q}^3 \times \mathcal{Q}^2. \end{array} \tag{11}$$

Simple activity bounds for the equation $x_1 + x_2 + x_4 + x_5 = 0$ *leads to* $L_{\min} = -\infty$ *and* $L_{\max} = \infty$ *since* x_2 *and* x_5 *are free variables. This equation is nevertheless forcing according to Definition 1, as shown by the following partition into Cartesian factors,*

$$a^T x = (a^1)^T x^1 + (a^2)^T x^2 = (x_1 + x_2) + (x_4 + x_5) = 0,$$

where $a^1 = (1, 1, 0)^T$ *and* $a^2 = (1, 1)^T$. *In particular, subgradient matching finds* $L^1_{\min} = 0$ *since* $a^1 \in \mathcal{Q}^3$, *as well as* $L^2_{\min} = 0$ *since* $a^2 \in \mathcal{Q}^2$, *to conclude* $L_{\min} = 0$ *which matches the value* $a^T x$ *is constrained to take. Hence,* $x_1 + x_2 = 0$ *and* $x_4 + x_5 = 0$ *by Proposition 6, exposing two facial reductions. That is, from* $x^1 \in \mathcal{Q}^3$ *to* $x^1 \in \mathcal{Q}^3 \cap (a^1)^\perp$ *and from* $x^2 \in \mathcal{Q}^2$ *to* $x^2 \in \mathcal{Q}^2 \cap (a^2)^\perp$.

Finally, note that simpler forms of subgradient matching have already appeared in literature. Gruber et al. [21] (and latter Cheung et al. [13]) notice that the existence of an equation $a^T x = 0$ in the considered problem, for a semidefinite coefficient and variable $a, x \in \mathcal{S}_+^n$, implies validity of the facial reduction from $x \in \mathcal{S}_+^n$ to $x \in \mathcal{S}_+^n \cap a^\perp$. This special case of subgradient matching was shown useful for the side chain positioning problem in [12].

Another far less apparent appearance of subgradient matching is given by the widely known facial reduction made possible by diagonal entries of a semidefinite cone fixed to zero (see, e.g., [39, page 535]). We elaborate on this specific usecase in the following paragraph.

Subgradient matching on variable bounds For a conic variable domain $x^i \in \mathcal{K}$, consider the set of variables selected by $a^T x$ for all permutations of $a = (1, 0, \ldots, 0)$ belonging to $\mathcal{K}^* \setminus \mathcal{K}^\perp$ (resp. to $-\mathcal{K}^* \setminus \mathcal{K}^\perp$). The variables of this set all have lower (resp. upper) bounds of zero by definition of dual cones. Suppose then that an upper bound $x_j \leq 0$ (resp. a lower bound $x_j \geq 0$) was derived for a variable of this set, e.g., using domain propagation. This would then show that the variable was forced to zero (as the execution of Algorithm 1 on this bound constraint would also reveal), whereby the facial reduction of Proposition 6 can be used on the conic variable domain.

Subgradient matching with general conic constraints One may confirm that Definition 1 and Proposition 6 can be generalized from expressions $\sum_{i=1}^p (a^i)^T x^i$ following the variable partition $x = (x^1, \ldots, x^p)$, to any finite list of variable-overlapping expressions of the form $\sum_{i=1}^p (a^i)^T x$. As a consequence, Algorithm 1 can be extended to include subgradient information from general conic constraints, $Dx - d \in \mathcal{K}$ for some $D \in \mathbb{R}^{p \times n}$ and $b \in \mathbb{R}^p$, using the partitioning strategy

$$a^T x = (a^1)^T x + (a^2)^T x = \Big(a^T x - \lambda^T (Dx - d)\Big) + \lambda^T (Dx - d),$$

for any $\lambda \in \mathbb{R}^p$. The term $(a^2)^T x = \lambda^T (Dx - d)$ is nonnegative and invariant to the computation of $L_{\min}$ if $\lambda \in \mathcal{K}^*$, or nonpositive and invariant to the computation of $L_{\max}$ if $\lambda \in -\mathcal{K}^*$. Hence, by restricting the domain of λ, one can disregard $(a^2)^T x$ in the respective bound computation and actively choose λ to strengthen the $L_{\min}$ (resp. $L_{\max}$) bound of $(a^1)^T x$, e.g., by eliminating unbounded variables. If the constraint on $a^T x$ is shown forcing, the expression $(a^2)^T x$ should of course be used to check for facial reductions in Proposition 6. In particular, if $\lambda \in \mathcal{K} \setminus (\mathcal{K}^*)^\perp$ (resp. $\lambda \in -\mathcal{K} \setminus (\mathcal{K}^*)^\perp$) holds, then the facial reduction from $Dx - d \in \mathcal{K}$ to $Dx - d \in \mathcal{K} \cap \lambda^\perp$ is justified. An exact strategy for subgradient matching with general conic constraints is left unexplored.

4.3 Single-cone analysis

As opposed to subgradient matching, which integrates information from conic constraints into the analysis of a single linear constraint, single-cone analysis acts to integrate information from linear constraints into the analysis of a single conic constraint. This is realized as a facial reduction heuristic for the second-order cones in this section. In motivation of this heuristic, the following example from [23] is now presented.

Example 4. *The non-standard mixed-integer optimization problem,*

$$\begin{array}{ll} \inf\limits_{x,t} & x^2 + t \\ \text{s.t.} & x - 4 \geq 0 \text{ if } t = 0, \\ & x \in \mathbb{R}_+, \\ & t \in \{0, 1\}, \end{array}$$

is solved by $(x, t) = (0, 1)$ and representable as a conic optimization problem by

$$\begin{array}{ll} \inf\limits_{x,t,\omega,y,\gamma} & \omega + t \\ \text{s.t.} & x - y - 4 \geq 0, \\ & \begin{pmatrix} 1/2 \\ \omega \\ x \end{pmatrix} \in \mathcal{Q}_r^3, \quad \begin{pmatrix} \gamma + t \\ \gamma - t \\ 2y \end{pmatrix} \in \mathcal{Q}^3, \\ & x \in \mathbb{R}_+, \\ & t \in \{0, 1\}. \end{array}$$

To give a feeling for this conic formulation, the first conic constraint models $\omega \geq x^2$ to represent the squared objective contribution. The second conic constraint models $y = 0$ if $t = 0$ and $\sqrt{\gamma} \geq |y|$ if $t = 1$ (for unbounded γ) to represent the conditional constraint (see [23] for more details).

Branching on $t = 0$, the relaxation is ill-posed as there is no way to satisfy $\begin{pmatrix} \gamma+t \\ \gamma-t \\ 2y \end{pmatrix} \in \operatorname{relint} \mathcal{Q}^3$ as needed for strong feasibility. In this case, however, there is also a linear dependency exposed by $z^T \begin{pmatrix} \gamma+t \\ \gamma-t \\ 2y \end{pmatrix} = 0$ for $z = \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$. That is, the conic constraint $\begin{pmatrix} \gamma+t \\ \gamma-t \\ 2y \end{pmatrix} \in \mathcal{Q}^3$ can be restated as

$$\begin{pmatrix} \gamma+t \\ \gamma-t \\ 2y \end{pmatrix} \in \mathcal{Q}^3 \cap \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}^{\perp} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \mathbb{R}_+ \text{ (see [17]).}$$

This simplifies to $\gamma \geq 0$ and $2y = 0$, and regularizes the relaxation as the reader may confirm by verifying the property of strong feasibility.

In order to systematize the search, exemplified above, for linear dependencies between the entries of conic constraints, let $D \in \mathbb{R}^{p \times n}$, $d \in \mathbb{R}^p$ and $\mathcal{K} \subseteq \mathbb{R}^p$ and consider

$$Dx - d = \left(D, \, d \right) \left(\begin{smallmatrix} x \\ -1 \end{smallmatrix} \right) \in \mathcal{K}.$$

Any linear dependency $z^T \left(D, \, d \right) = 0$, for nonzero $z \in \mathbb{R}^p$, justifies reformulation to the subspace intersected conic constraint $Dx - d \in \mathcal{K} \cap z^{\perp}$. For z to expose a proper face of a $\mathcal{K}$, as needed for a facial reduction, we require a solution to the system

$$z^T \left(D, \, d \right) = 0, \quad z \in \mathcal{K}^* \setminus \mathcal{K}^{\perp},$$

which is comparable to Proposition 2-(2) and equivalent to the system

$$\left(D, \, d \right)^T z = 0, \quad \hat{p}^T z = 1, \quad z \in \mathcal{K}^*, \tag{12}$$

for some relative interior point $\hat{p} \in \operatorname{relint} \mathcal{K}$ [33]. Finally, if $\mathcal{K}^*$ is the image under linear mapping of another set, i.e., $\mathcal{K}^* = H\hat{\mathcal{K}}^*$, then (12) with $z = H\hat{z}$ is equivalent to the system

$$\left(H^T D, \, H^T d \right)^T \hat{z} = 0, \quad (H^T \hat{p})^T \hat{z} = 1, \quad \hat{z} \in \hat{\mathcal{K}}^*. \tag{13}$$

These systems are now shown analytically solvable for the second-order cones. The first step in this direction is to rewrite (12) as a least-norm optimization problem for the quadratic cone.

Proposition 7. *Let $D = \left(\begin{smallmatrix} \alpha^T \\ A \end{smallmatrix} \right) \in \mathbb{R}^{p \times n}$ and $d = \left(\begin{smallmatrix} \beta \\ b \end{smallmatrix} \right) \in \mathbb{R}^p$. The system (12), with $\mathcal{K} = \mathcal{Q}^p$ and the relative interior point $\hat{p} = (1, 0, \ldots, 0)^T \in \operatorname{relint} \mathcal{Q}^p$, is feasible if and only if $\theta_P \leq 1$ for the least-norm optimization problem*

$$\begin{array}{rcll} \theta_P & = & \inf\limits_{\lambda} & \|\lambda\|_2 \\ & & \text{s.t.} & \left(A, \, b \right)^T \lambda = -\left(\begin{smallmatrix} \alpha \\ \beta \end{smallmatrix} \right), \\ & & & \lambda \in \mathbb{R}^{p-1}. \end{array} \tag{14}$$

Specifically, a feasible point of (12) is given by $z = (1, \hat{\lambda}^T)^T$ for any feasible point $\hat{\lambda}$ of (14) with an objective value less than or equal to one.

Proof. Let $z \in \mathbb{R}^p$ satisfy (12) such that $\hat{p}^T z = z_1 = 1$. Then $z = (1, \lambda^T)^T$ for some $\lambda \in \mathbb{R}^{p-1}$. The claim follows by $z \in \mathcal{Q}^p \iff \|\lambda\|_2 \leq 1$ and $z^T \left(D \; d \right) = \left(\alpha^T \; \beta \right) + \lambda^T \left(A \; b \right)$. □

Algorithm 2: Single-cone analysis for a second-order cone.

Data: Let $D \in \mathbb{R}^{p \times n}$, $b \in \mathbb{R}^p$ and $\mathcal{K} \subseteq \mathbb{R}^p$, such that $Dx - d \in \mathcal{K}$ is a second-order cone constraint (or variable domain) of the considered problem.

while $Dx - d$ contains a singleton variable **do**

  Substitute the singleton variable in $Dx - d$ with its unique definition in the equation system of the considered problem.

end

return a facial reduction certificate, if one is found following the instructions of Corollary 4.

The least-norm optimization problem of Proposition 7 is widely documented elsewhere and can, although alternatives exists, be solved to a fair balance between speed and accuracy by means of QR-decomposition [22]. For completeness, this approach is now established.

Proposition 8. *Consider the least-norm optimization problem of Proposition 7, and the QR-decomposition with pivoting* $(Q_1, Q_2) \left(\begin{smallmatrix} R_1 \\ 0 \end{smallmatrix} \right) = \left(\begin{smallmatrix} A, & b \end{smallmatrix} \right) P$ *for full row rank* $R_1 \in \mathbb{R}^{r \times p}$*. Then* $\lambda = Q \left(\begin{smallmatrix} \lambda' \\ 0 \end{smallmatrix} \right)$ *is a solution to* (14) *if and only if* λ' *is a solution to*

$$(R_1)^T \lambda' = -P^T \left(\begin{smallmatrix} \alpha \\ \beta \end{smallmatrix} \right), \tag{15}$$

which can be determined by forward substitution.

Proof. The system (15) is equivalent to $PR^T \left(\begin{smallmatrix} \lambda' \\ 0 \end{smallmatrix} \right) = P(R_1)^T \lambda' = -\left(\begin{smallmatrix} \alpha \\ \beta \end{smallmatrix} \right)$. Hence, by definition of λ and $Q^T Q = \mathrm{I}$, the system implies feasibility of λ in (14) as shown by $PR^T Q^T \lambda = (A, b)^T \lambda = -\left(\begin{smallmatrix} \alpha \\ \beta \end{smallmatrix} \right)$. To show that λ is the unique optimal solution of (14), note that any other feasible point $\hat{\lambda}$ of (14) satisfy $(\hat{\lambda} - \lambda)^T \lambda = (\hat{\lambda} - \lambda)^T Q \left(\begin{smallmatrix} \lambda' \\ 0 \end{smallmatrix} \right) = (\hat{\lambda} - \lambda)^T QRP^T \omega = ((A, b)^T \hat{\lambda} - (A, b)^T \lambda)^T \omega = 0$ where $R(P^T \omega) = \left(\begin{smallmatrix} \lambda' \\ 0 \end{smallmatrix} \right)$ is solvable because R_1 has full row rank. Thus, $\|\hat{\lambda}\|_2^2 = \|\hat{\lambda} - \lambda + \lambda\|_2^2 = \|\hat{\lambda} - \lambda\|_2^2 + \|\lambda\|_2^2 + 2(\hat{\lambda} - \lambda)^T \lambda = \|\hat{\lambda} - \lambda\|_2^2 + \|\lambda\|_2^2 > \|\lambda\|_2^2$ for $\lambda \neq \hat{\lambda}$. ☐

These results are finally be combined in Corollary 4 and realized in the facial reduction heuristic of Algorithm 2, allowing heuristic regularization of the second-order cones in (P) and (D).

Corollary 4. *Let* $D \in \mathbb{R}^{p \times n}$ *and* $b \in \mathbb{R}^p$ *such that* $Dx - d \in \mathcal{K}$ *is a second-order cone constraint. Any linear dependency between the conic entries, exposing a facial reduction, can be found analytically.*

1. *If* $\mathcal{K} = \mathcal{Q}^p$*, the system* (12) *can be solved using* $\hat{p} = (1, 0, \ldots, 0)^T \in \operatorname{relint} \mathcal{Q}^p$ *as shown by Proposition 7 and Proposition 8.*

2. *If* $\mathcal{K} = \mathcal{Q}_r^p$*, the system* (13) *can be solved as in statement 1, using* $H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 \\ 0 & 0 & \mathrm{I} \end{pmatrix}$*, such that* $(\mathcal{Q}_r^p)^* = H(\mathcal{Q}^p)^*$*, and* $\hat{p} = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \ldots, 0)^T \in \operatorname{relint} \mathcal{Q}_r^p$*, such that* $H^T \hat{p} = (1, 0, \ldots, 0)$*.*

Note that the substitution mechanism of Algorithm 2 is rather unsophisticated, simply purging singleton variables, and improvements could possibly be made in this direction. The merits of additional substitutions are hard to quantify without executing the computations of Corollary 4 repeatedly, however, and the idea is hence left unexplored.

4.4 Heuristics based on cone factorization

Heuristics based on cone factorization exploits the fact that the equations of the auxiliary problems of Proposition 2 are homogeneous. In particular, if a cone can be put on product-form, then inner approximations of the auxiliary problems are simply obtained by dropping factors of that product-form. A simple analogy of this relation is that $ax = 0$ implies $axy = 0$ for variables $x, y \in \mathbb{R}$. Taking the semidefinite cone as example, this approach leads to the facial reduction heuristic from [13, Algorithm 1.0.2].

Example 5. *The semidefinite cone from Section 3.1 was defined in its product-form as*

$$\mathcal{S}_+^N := \left\{ VV^T : V \in \mathbb{R}^{N\times N} \right\} \subseteq \mathbb{R}^{N\times N}. \tag{16}$$

Substituting this definition into the auxiliary problem from Proposition 2-(2), and taking advantage of $VV^T \neq 0 \Longleftrightarrow V \neq 0$, one obtain in matrix notation

$$\langle C, VV^T\rangle = \langle CV, V\rangle = 0, \qquad \begin{pmatrix} \langle A_1, VV^T\rangle \\ \vdots \\ \langle A_m, VV^T\rangle \end{pmatrix} = \begin{pmatrix} \langle A_1V, V\rangle \\ \vdots \\ \langle A_mV, V\rangle \end{pmatrix} = 0, \qquad V \in \mathbb{R}^{N\times N} \setminus \{0\},$$

where $\langle X, Y\rangle = \mathrm{Tr}(Y^TX)$ is the trace inner product. An inner approximation is thus obtained by

$$CV = 0, \qquad \begin{pmatrix} A_1V \\ \vdots \\ A_mV \end{pmatrix} = 0, \qquad V \in \mathbb{R}^{N\times N} \setminus \{0\},$$

from which facial reduction certificates for (D) can be computed as VV^T, for nonzero matrices V composed of columns from the common nullspace of C and A_i for $i = 1, \ldots, m$.

More generally, one may consider all proper cones factorisable into a bilinear matrix-vector product of the form

$$\mathcal{K} = \{L(v)v : v \in \mathbb{R}^n\} \subseteq \mathbb{R}^n. \tag{17}$$

The semidefinite cone from Example 5 is included in vectorized form by this characterization as will become clear, and a facial reduction heuristic for (D), based on the auxiliary problem from Proposition 2-(2), can similarly be derived.

Theorem 2. *Consider the product-form* (17) *of a Cartesian product of proper cones,*

$$\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_k = \left\{ L(v)v = \begin{pmatrix} L_1(v_1) & & \\ & \ddots & \\ & & L_k(v_k) \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_k \end{pmatrix} : v \in \mathbb{R}^n = \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_k} \right\} \subseteq \mathbb{R}^n.$$

Substituting this definition into the auxiliary problem from Proposition 2-(2), an inner approximation can be obtained, as proven, by

$$c^T \begin{pmatrix} L_1(v^1) & & \\ & \ddots & \\ & & L_k(v^k) \end{pmatrix} = 0, \qquad A \begin{pmatrix} L_1(v^1) & & \\ & \ddots & \\ & & L_k(v^k) \end{pmatrix} = 0, \qquad v \in \mathbb{R}^n \setminus \{0\}.$$

Let $c^T = (c_1^T, \ldots, c_k^T)$ and $A = (A_1, \ldots, A_k)$ according to the Cartesian product. Facial reduction certificates for (D) can then be computed as vectors $z = L(v)v$, for nonzero $v = (v^1, \ldots, v^k) \in \mathbb{R}^n$ satisfying $c_i^T L_i(v^i) = 0$ and $A_iL_i(v^i) = 0$ for $i = 1, \ldots, k$.

Proof. The only difficult part is

$$\mathcal{K} \setminus (\mathcal{K}^*)^{\perp} = \mathcal{K} \setminus \{0\} = \{L(v)v : v \in \mathbb{R}^n \setminus \{0\}\}.$$

The first step holds since $\mathcal{K}^*$ is solid, as shown by the closure of $\mathcal{K}$ being pointed [10]. The second step holds since $L(v)v = 0$ only if $v = 0$, as can be shown by $\mathcal{K}$ being solid. In particular, assume that $L(\hat{v})\hat{v} = 0$ for some $\hat{v} \neq 0$. Then $L(\lambda\hat{v})(\lambda\hat{v}) = \lambda^2 L(\hat{v})\hat{v} = 0$ for all $\lambda \in \mathbb{R}$ by bilinearity. Hence, $\operatorname{span} \mathcal{K} \neq \mathbb{R}^n$ contradicting $\mathcal{K}$ being solid. ☐

Heuristics for (D) based on the cone factorization approach of Theorem 2 can be realized for the class of symmetric cones [44]. Formally:

Proposition 9. *The product-form* (17) *is achieved by the following cones:*

1. *The nonnegative orthant* $\mathbb{R}_+$, *representable by* $L(v) = v$;
2. *The quadratic cone* $\mathcal{Q}^n$, *representable by* $L(v) = \begin{pmatrix} v_1 & v_{2:n}^T \\ v_{2:n} & v_1 I \end{pmatrix}$;
3. *The semidefinite cone* $\operatorname{svec}(\mathcal{S}_+^N)$, *representable by* $L(v) = I \otimes_s \operatorname{smat}(v)$, *where* $v \in \mathbb{R}^{N(N+1)/2}$;
4. *The semidefinite cone* $\operatorname{vec}(\mathcal{S}_+^n)$, *representable by* $L(v) = I \otimes \operatorname{mat}(v)^T$, *where* $v \in \mathbb{R}^{N^2}$.

Proof. Statements 1-3 follow from the fact that all symmetric cones have the product-form,

$$\mathcal{K} = \{v \circ v : v \in \mathbb{R}^n\} \subseteq \mathbb{R}^n,$$

for some Euclidean Jordan algebra with associated Jordan product '$\circ$'. The Jordan product is bilinear such that $v \circ w = L(v)w$ for some symmetric linear mapping $L(v) : \mathbb{R}^n \to \mathcal{S}^{n \times n}$, and the definitions of $L(v)$ for the cones above are well known [15, 44]. The symmetric Kronecker product '$\otimes_s$' as found, e.g., in [44], satisfies $L(v)w = \operatorname{svec}(\operatorname{smat}(v)\operatorname{smat}(w) + \operatorname{smat}(w)\operatorname{smat}(v))/2$.

Statement 4 follows from the product-form (16) and the Kronecker product '$\otimes$' for which $L(v)w = \operatorname{vec}(\operatorname{mat}(v)^T \operatorname{mat}(w))$. ☐

It is unclear whether Theorem 2 represents any advancement from the special case of Example 5. First of all, Statement 1 is an idempotent as $A_i L(v)v = 0 \iff A_i L(v) = 0$ in Theorem 2 for all $L(v) = \lambda v$ where $\lambda > 0$. Secondly, Statement 2 and Statement 3 are restricted in usability by symmetry of $L(v)$, and Statement 4 leads to the characterization shown in Example 5. Finally, the author was unable to use the product-form (17) to construct an inner approximation for Proposition 2-(1) that was not trivially infeasible. Similar heuristics for (P), based on cone factorization, might thus not be possible.

5 Conclusion

Facial reduction is theoretically established as a useful countermeasure against ill-posedness in mixed-integer optimization which affects both conic and linear relaxations. Speedy and/or accurate facial reduction heuristics are further motivated by the slow and inaccurate alternative of having to solve conic optimization problems in each iteration of the facial reduction algorithm. This led to the development of the heuristics based on linear optimization, subgradient matching and single-cone analysis, which were shown useful in various scenarios. A fourth type of heuristic based on cone factorization may also prove useful, although it seems to be limited in applicability to dual facial reductions of semidefinite cones.

Further work is needed to test how these heuristics compare and performs in practice, such as on the instances of CBLIB [18]. Also, it remains to be investigated how these heuristics should be integrated with the brand-and-bound algorithm for greatest accuracy, greatest speed, or a balance thereof. In this regard, it is likely that local information on branching decisions, presolve changes and generated cuts can be used to guide the employment of heuristics. In a sense, subgradient matching already achieves this by integrating with domain propagation to capture facial reductions only from the changes propagated out.

Acknowledgments The author, Henrik A. Friberg, owe a special thanks to Erling D. Andersen and Mathias Stolpe who supervised the work on this paper. The author was funded by MOSEK ApS and the Danish Ministry of Higher Education and Science through the Industrial PhD project "Combinatorial Optimization over Second-Order Cones and Industrial Applications".

References

[1] T. Achterberg. *Constraint Integer Programming.* PhD thesis, Technische Universität Berlin, 2008.

[2] E. Andersen, C. Roos, and T. Terlaky. Notes on Duality in Second Order and p-Order Cone Optimization. *Optimization: A Journal of Mathematical Programming and Operations Research*, 51(4):627–643, 2002.

[3] E. D. Andersen and K. D. Andersen. Presolving in linear programming. *Mathematical Programming*, 71(2):221–245, 1995.

[4] M. F. Anjos and S. Burer. On handling free variables in interior-point methods for conic linear optimization. *SIAM Journal on Optimization*, 18(4):1310–1325, 2007.

[5] M. F. Anjos and H. Wolkowicz. Strengthened semidefinite relaxations via a second lifting for the Max-Cut problem. *Discrete Applied Mathematics*, 119(1-2):79–106, 2002.

[6] P. Belotti, J. C. Goez, I. Pólik, T. K. Ralphs, and T. Terlaky. A conic representation of the convex hull of disjunctive sets and conic cuts for integer second order cone optimization. In *Numerical Analysis and Optimization*, volume 134 of *Springer Proceedings in Mathematics & Statistics*, pages 1–35. Springer International Publishing, 2015.

[7] H. Y. Benson and Ü. Sağlam. Mixed-integer second-order cone programming: A survey. In *Tutorials in Operations Research*, INFORMS, Minneapolis, 2013.

[8] J. Borwein and H. Wolkowicz. Regularizing the abstract convex program. *Journal of Mathematical Analysis and Applications*, 83(2):495–530, 1981.

[9] J. M. Borwein and A. S. Lewis. Partially finite convex programming, Part II: Explicit lattice models. *Mathematical Programming*, 57(1-3):49–83, 1992.

[10] S. Boyd and L. Vandenberghe. *Convex optimization.* Cambridge University Press, 2004.

[11] F. Burkowski, Y.-L. Cheung, and H. Wolkowicz. Efficient Use of Semidefinite Programming for Selection of Rotamers in Protein Conformations. *INFORMS Journal on Computing*, 26(4):748–766, 2014.

[12] Y.-L. Cheung. *Preprocessing and Reduction for Semidefinite Programming via Facial Reduction: Theory and Practice.* PhD thesis, University of Waterloo, 2013. URL `http://orion.uwaterloo.ca/~hwolkowi/henry/reports/thesisvrisnov13.pdf`.

[13] Y.-L. Cheung, S. Schurr, and H. Wolkowicz. Preprocessing and Regularization for Degenerate Semidefinite Programs. In *Computational and Analytical Mathematics*, volume 50 of *Springer Proceedings in Mathematics & Statistics*, pages 251–303. Springer New York, 2013.

[14] D. Drusvyatskiy, G. Pataki, and H. Wolkowicz. Coordinate shadows of semi-definite and euclidean distance matrices. *arXiv*, 2015. URL `http://arxiv.org/abs/1405.2037`.

[15] J. Faraut and A. Korányi. *Analysis on symmetric cones.* 1994.

[16] R. M. Freund and J. R. Vera. Some characterizations and properties of the "distance to ill-posedness" and the condition measure of a conic linear system. *Mathematical Programming*, 86(2):225–260, 1999.

[17] H. A. Friberg. A relaxed-certificate facial reduction algorithm based on subspace intersection. *Optimization Online*, 2016. URL `http://www.optimization-online.org/DB_HTML/2016/01/5283.html`.

[18] H. A. Friberg. CBLIB 2014: a benchmark library for conic mixed-integer and continuous optimization. *Mathematical Programming Computation*, Online 2015.

[19] F. Glineur. Second-order cone optimization with a single second-order cone. In *18th International Symposium on Mathematical Programming (ISMP)*, 2003.

[20] G. Gruber and F. Rendl. Computational experience with ill-posed problems in semidefinite programming. *Computational Optimization and Applications*, 21(2):201–212, 2002.

[21] G. Gruber, S. Kruk, F. Rendl, and H. Wolkowicz. Presolving for Semidefinite Programs Without Constraint Qualifications. Technical Report CORR 98-32, University of Waterloo, Waterloo, Ontario, 1998.

[22] N. J. Higham. Underdetermined Systems. In *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, 2nd edition, 2002.

[23] H. Hijazi and L. Liberti. Constraint Qualification Failure in Second-Order Cone Formulations of Unbounded Disjunctions. Technical report, NICTA, Canberra ACT Australia, 2014.

[24] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

[25] F. Kılınç-Karzan and S. Yıldız. Two-Term Disjunctions on the Second-Order Cone. In *Integer Programming and Combinatorial Optimization*, volume 8494 of *Lecture Notes in Computer Science*, pages 345–356. Springer International Publishing, 2014.

[26] T. Koch. The final NETLIB-LP results. *Operations Research Letters*, 32(2):138–142, 2004.

[27] N. Krislock and H. Wolkowicz. Explicit sensor network localization using semidefinite representations and clique reductions. *SIAM Journal on Optimization*, 20(5):2679–2708, 2010.

[28] B. F. Lourenço, M. Muramatsu, and T. Tsuchiya. Solving SDP Completely with an Interior Point Oracle. *Optimization Online*, 2015. URL `http://www.optimization-online.org/DB_HTML/2015/06/4982.html`.

[29] Z.-Q. Luo, J. F. Sturm, and S. Zhang. Duality Results for Conic Convex Programming. Technical Report 9719/A, Econometric Institute, Erasmus University Rotterdam, The Netherlands, 1997.

[30] S. Mars and L. Schewe. An SDP-Package for SCIP. Technical Report August, TU Darmstadt, 2012. URL `http://www.opt.tu-darmstadt.de/~smars/scip_sdp.html`.

[31] A. Nemirovski. Efficient methods in convex programming. Technion - The Israel Institute Of Technology Faculty Of Industrial Engineering & Management, 1994.

[32] G. Pataki. Strong duality in conic linear programming: facial reduction and extended duals. In *Springer Proceedings in Mathematics and Statistics*, volume 50, pages 613–634. Springer New York, 2013.

[33] F. Permenter and P. A. Parrilo. Partial facial reduction: simplified, equivalent SDPs via approximations of the PSD cone. *arXiv*, 2014. URL `http://arxiv.org/abs/1408.4685`.

[34] F. Permenter, H. A. Friberg, and E. D. Andersen. Solving conic optimization problems via self-dual embedding and facial reduction : a unified approach. Technical report, 2015. `http://www.optimization-online.org/DB_HTML/2015/09/5104.html`.

[35] J. Renegar. Incorporating Condition Measures into the Complexity Theory of Linear Programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995.

[36] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[37] M. W. P. Savelsbergh. Preprocessing and probing techniques for mixed integer programming problems. *ORSA Journal on Computing*, 6(4):445–454, 1994.

[38] J. Stoer and C. Witzgall. *Convexity and Optimization in Finite Dimensions I*. Springer Verlag, Berlin, 1970.

[39] M. J. Todd. Semidefinite optimization. *Acta Numerica*, 10(C):515–560, 2001.

[40] L. Tunçel and H. Wolkowicz. Strong duality and minimal representations for cone optimization. *Computational Optimization and Applications*, 53(2):619–648, 2012.

[41] H. Waki and M. Muramatsu. Facial reduction algorithms for conic optimization problems. *Journal of Optimization Theory and Applications*, 158(1):188–215, 2013.

[42] H. Waki, M. Nakata, and M. Muramatsu. Strange behaviors of interior-point methods for solving semidefinite programming problems in polynomial optimization. *Computational Optimization and Applications*, 53(3):823–844, 2011.

[43] H. Wolkowicz and Q. Zhao. Semidefinite programming relaxations for the graph partitioning problem. *Discrete Applied Mathematics*, 96-97:461–479, 1999.

[44] H. Wolkowicz, R. Saigal, and L. Vandenberghe. Handbook of Semidefinite Programming. 2000.

[45] M. Yamashita, K. Fujisawa, K. Nakata, M. Nakata, M. Fukuda, K. Kobayashi, and K. Goto. A high-performance software package for semidefinite programs: SDPA 7. Technical report, 2010.

[46] Y. Ye, M. J. Todd, and S. Mizuno. An $\mathcal{O}(\sqrt{n}L)$-Iteration Homogeneous and Self-Dual Linear Programming Algorithm, 1994.

[47] Q. Zhao, S. E. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite Programming Relaxations for the Quadratic Assignment Problem. *Journal of Combinatorial Optimization*, 2(1):71–109, 1998.