ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Disjunctive cuts in Mixed-Integer Conic Optimization

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:
Lodi A., Tanneau M., Vielma J.-P. (2022). Disjunctive cuts in Mixed-Integer Conic Optimization. MATHEMATICAL PROGRAMMING, 199(1-2), 671-719 [10.1007/s10107-022-01844-1].

Availability:
This version is available at: https://hdl.handle.net/11585/905151 since: 2024-03-01

Published:
DOI: http://doi.org/10.1007/s10107-022-01844-1

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

15 September 2026

Noname manuscript No.
(will be inserted by the editor)

Disjunctive cuts in Mixed-Integer Conic Optimization

Andrea Lodi · Mathieu Tanneau · Juan-Pablo Vielma

the date of receipt and acceptance should be inserted later

Abstract This paper studies disjunctive cutting planes in Mixed-Integer Conic Programming. Building on conic duality, we formulate a cut-generating conic program for separating disjunctive cuts, and investigate the impact of the normalization condition on its resolution. In particular, we show that a careful selection of normalization guarantees its solvability and conic strong duality. Then, we highlight the shortcomings of separating conic-infeasible points in an outer-approximation context, and propose conic extensions to the classical lifting and monoidal strengthening procedures. Finally, we assess the computational behavior of various normalization conditions in terms of gap closed, computing time and cut sparsity. In the process, we show that our approach is competitive with the internal lift-and-project cuts of a state-of-the-art solver.

Keywords Mixed-Integer Convex Optimization · Disjunctive Cutting Planes · Conic Optimization.

1 Introduction

Mixed-Integer Convex Optimization (MI-CONV) is a fundamental class of Mixed-Integer Non-Linear Optimization problems with applications such as risk management, non-linear physics (e.g., power systems and chemical engineering) and logistics, just to mention a few. Because of such a relevance, classical algorithms for Mixed-Integer Linear Optimization (MILP) have been successfully extended to MI-CONV, like Branch and Bound [14] or Benders decomposition [26]; others like the Outer Approximation scheme [22] have been designed specifically for MI-CONV. In addition, several software tools

A. Lodi · M. Tanneau
CERC, Polytechnique Montréal - Canada Excellence Research Chair,
C.P. 6079, Succ. Centre-ville, Montral, H3C 3A7 QC, Canada

J.P. Vielma
MIT Sloan School of Management, Cambridge, MA

are available for solving general MI-CONV problems, see, e.g., the recent comparison in [35]. Finally, some specific classes of MI-CONV problems, like Mixed-Integer (Convex) Quadratically Constrained Quadratic Optimization (MIQCQP) problems are now supported by the major commercial solvers.

Conic optimization is viewed as a more numerically stable and tractable alternative to general convex optimization [11]. Both classes are equivalent: conic optimization problems are convex, and any convex optimization problem can be written as a conic optimization problem [34]. Modeling tools such as disciplined convex optimization [27] can provide conic formulations for most –if not all– convex optimization problems that arise in practice [44]. In particular, [39] recently showed that all convex instances in MINLPLib can be formulated as Mixed-Integer Conic Optimization (MI-CONIC) problems using only a handful of cones.

Nevertheless, the intrinsic difference between convex and conic optimization lies in a problem's algebraic description: in the former, constraints are formulated as $f(x) \leq 0$, where f is a convex function, whereas, in the latter, they are expressed using conic inequalities of the form $Ax - b \in \mathcal{K}$, where A is a matrix, b is a vector and $\mathcal{K}$ is a cone (see [11] and Section 2). In particular, conic formulations enable the use of conic duality theory, which underlies a number of theoretical insights and practical tools. Major commercial solvers have supported Mixed-Integer Second Order Cone Programming (MISOCP) for some time, and more general MI-CONIC problems are now supported by a number of solvers, e.g., Mosek and Pajarito [19,38,39].

This paper builds on two specific aspects that we consider fundamental for solving MI-CONV problems. First, given that cutting planes are instrumental to solving MILP, a number of authors have looked at various approaches to compute cuts for MI-CONV problems and, nowadays, linear cutting planes are part of the arsenal of some MI-CONV solvers. Despite this (partial) success, some fundamental questions in this area are left unanswered. Second, recent experience has shown that *conic* formulations of MI-CONV problems display enviable properties that make them preferable, from the solving viewpoint, to generic MI-CONV formulations [39,19].

In that context, motivated by the success of disjunctive cuts in MILP, the paper focuses on computational aspects of disjunctive cuts for MI-CONIC problems. In addition, we answer the (somehow) natural question of what one can gain in terms of cutting planes by using a problem's *conic* structure, as well as several questions left open by previous works on the topic. In the remainder of this section, we review the literature on the subject and outline our main contributions.

1.1 Disjunctive cuts: the MILP case

Disjunctive cuts in MILP date back to Balas' seminal work on disjunctive programming [4] in the 70s, and became widely popular as their integration into branch-and-cut frameworks [5,6] proved effective. Remarkably, disjunctive

cuts, split cuts in particular, encompass several classes of cutting planes, e.g., Chvatal-Gomory, Gomory Mixed-Integer and Mixed-Integer Rounding cuts.

A general approach for separating disjunctive cuts in MILP is the so-called Cut-Generating Linear Program (CGLP) proposed by Balas [4,5]. The CGLP leverages a characterization of valid inequalities for disjunctive sets using Farkas multipliers, see Theorem 3.1 in [4]. Thus, it is formulated in a higher-dimensional space, whose size is proportional to the number of disjunctive terms: for split cuts, which are two-term disjunctions, the CGLP is roughly double the size of the original problem.

Computational aspects of the CGLP have been studied extensively, some of which we mention here. Given a fractional point $\bar{x}$ to separate, one can project the CGLP onto the support of $\bar{x}$, thereby reducing its size, and recover a valid cut by lifting [5,6]. Split cuts obtained from solving the CGLP can be improved upon using monoidal strengthening [7,6]. The normalization condition in the CGLP has been shown to have a major impact on the quality of the obtained cuts, and on overall performance [23,13,16,47,20]. In particular, Balas and Perregard [9], and later Bonami [13], show that, in the case of split disjunctions, the CGLP can in fact be solved in the space of original variables only, yielding substantial computational gains. Recent developments include the efficient separation of cuts from multiple disjunctions [45,30].

1.2 Disjunctive cuts: the MI-CONV case

The work on disjunctive cutting planes for MI-CONV (re)started already in the late 90s with two fundamental contributions [17,48]. More precisely, Ceria and Soares [17] show that disjunctive convex problems can be formulated as a single convex problem in a higher dimensional space, and hint that this could serve to generate cutting planes using sub-gradient information at the optimum. Around the same time, Stubbs and Mehrotra [48] make the separation of disjunctive cuts for MI-CONV explicit by *(i)* solving one Non-Linear Programming (NLP) problem, and *(ii)* identifying a sub-gradient that yields a violated cut. The latter is done by taking a gradient (under regularity assumptions), or by solving a linear system (under the assumption that the objective function of the former problem is polyhedral). Those assumptions and the use of perspective functions lead to differentiability issues that made the results of the computational investigation in [48] numerically disappointing (according to the authors themselves).

The numerical difficulties encountered in [48] have slowed down the development of the area for a number of years –with the exception of [52]– until the renewed interest and the practical approaches of the last decade [12,31]. More precisely, Kilinc et al. [31] note that *"A simple strategy for generating lift-and-project cuts for a MINLP problem is to solve a CGLP [...] based on a given polyhedral outer approximation of the relaxed feasible region [...]. The key question to be answered [...] is which points to use to define the polyhedral relaxation."* (from [31], Sec. 3).

The distinction in how to answer the above question is the difference between [52], [12], and [31]. Namely, Zhu and Kuno [52] build an outer-approximation through the current fractional solution, and derive a cut by solving the associated CGLP. However, this approach is not guaranteed to find a violated cut if one exists, see Example 1 in [31]. Bonami [12] solves one auxiliary NLP, and uses the solution to get an outer approximation that provably yields a violated cut if any exists. Instead, Kilinc et al. [31] iteratively refine an outer approximation by solving a sequence of LPs until a violated cut, if any, is separated by solving the associated CGLP.

In a recent paper, Kronqvist and Misener [36] present a disjunctive-based cut-strengthening technique for MI-CONV. Given an initial valid inequality, and an "exclusive" selection constraint (i.e., $x_1 + ... + x_k = 1,\ x \in \{0,1\}^k$), the procedure solves k convex problems to tighten the cut's right-hand side and the coefficients of the k binary variables. This approach relates to the generalized disjunctive programming framework [49], for which cutting-plane algorithms based on [48] have been proposed, see, e.g., [50].

The outer approximation approaches in [12,31] are, to the best of our knowledge, the state of the art for the implementation of disjunctive cuts for MI-CONV and, especially, for MIQCQPs, see e.g., their implementation in CPLEX starting from version 12.6.2. However, despite the impressive practical improvements with respect to the early attempts [48], questions were left on the table, which we answer in the present paper.

1.3 Disjunctive cuts: the MI-CONIC case

Following the support of MISOCP problems by major commercial solvers in the 2000s (MISOCP support appeared in CPLEX 9.0 and in Gurobi 5.0), the last decade has seen a flourishing literature on cuts for MI-CONIC problems.

A large share of these works focus on cuts for MISOCP, or, equivalently, for convex MIQCQP problems. Atamturk and Narayanan [2] introduce conic Mixed-Integer Rounding (MIR) cuts for MISOCP problems. Modaresi et al later show in [41,40] that conic MIR cuts are in fact linear split cuts in an extended space, and compare the strength of families of conic MIR cuts to that of non-linear split cuts. In a related work, Andersen and Jensen [1] study intersection cuts in the MISOCP context, and obtain a closed-form formula for the conic quadratic intersection cut. Belotti et al. [10] study the intersection of a convex set and a two-term disjunction. They show that the convex hull is described by a single conic inequality, for which an explicit formula is derived in the conic quadratic case. In a similar fashion, two-term disjunctions on the second-order cone are investigated in [33], and this approach is later extended in [51].

More general approaches, i.e., not restricted to convex quadratic constraints, include [18,3,21,34,42]. In [18], the authors study classes of cutting planes in the MI-CONIC setting, including Chvatal-Gomory cuts and lift-and-project cuts, and report limited experiments on mixed 0-1 semi-definite programming

instances. A generic lifting procedure for conic cuts is described in [3]. Dadush et al. [21] show that the split closure of a strictly convex body is defined by a finite number of disjunction, but is not necessarily polyhedral. Minimal valid inequalities are introduced in [34], and are shown to be sufficient to describe the convex hull of a disjunctive conic set. The lack of tractable algebraic representation for minimal inequalities then leads the author to consider the broader class of sublinear inequalities, which are further studied in [32]. Finally, intersection cuts for non-polyhedral sets and certain classes of disjunctions are studied in [42].

Nevertheless, for the most part, these works remain theoretical contributions. Indeed, with the exception of [18,2,40], no computational results were reported for any of these techniques and, to the best of our knowledge, none has been implemented in optimization solvers. In fact, neither Mosek nor Gurobi[1] generate cuts from non-linear information.

1.4 Contribution and outline

In this paper, we study linear disjunctive cutting planes for MI-CONIC problems. Our objective is to derive practical and numerically robust tools for the separation of those cuts, and we show that conic formulations allow us to achieve it. Specifically, we do so by extending Balas' CGLP into a Cut-Generating Conic Program (CGCP) (see also [18]). Our contributions are:

1. We study the role of the normalization condition in the CGCP, and propose conic normalizations that guarantee strong duality. In doing so, we answer some concerns that were raised in previous works. Namely,
   - With respect to [12,18,36], we can select the right normalization to overcome issues associated with potential lack of constraint qualification.
   - With respect to [31,36], since we use conic formulations, we do not need *(i)* to pay attention at avoiding generating linearization cuts at points outside the domain where the non-linear functions are known to be convex, *(ii)* to deal with non-differentiable functions, and *(iii)* boundedness assumptions on the value of the constraints and their gradients.
2. We draw attention to some limitations of separating conic-infeasible points in an outer-approximation context, and propose algorithmic strategies to alleviate them.
3. We introduce conic extensions of the lifting procedure for disjunctive cuts, and of monoidal strengthening for split cuts.
4. We provide computational results on the effectiveness of the proposed approach, thereby showing the benefits of the conic representation, and compare the practical effectiveness of several normalization conditions.
5. We make our implementation available[2] under an open-source license.

[1] Personal communication with Gurobi and Mosek developers.

[2] Our code is released at `https://github.com/mtanneau/CLaP`

The remainder of the paper is structured as follows. In Section 2, we introduce some required notation and background material on conic optimization, and state a number of theoretical results on the characterization of valid inequalities for conic and disjunctive conic sets. Section 3 formalizes the CGCP and its dual, and the theoretical properties of several normalization conditions for the CGCP are discussed in Section 4. The separation of conic-infeasible points is further investigated in Section 5, while classical lifting and strengthening techniques from MILP are extended to the conic setting in Section 6. In Section 7, we analyze the practical behavior of different normalizations, and show that our approach is competitive with CPLEX internal lift-and-project cuts. Some concluding remarks are presented in Section 8.

2 Background

In this section, we introduce some notations, and recall a number of results that are needed for our approach. We refer to [46] for a thorough overview of convex analysis, and to [11] and [4] for results on conic optimization and disjunctive programming, respectively.

For $\mathcal{X} \subseteq \mathbb{R}^n$, we denote by $\text{int}(\mathcal{X})$, $\partial(\mathcal{X})$, $\text{cl}(\mathcal{X})$, and $\text{conv}(\mathcal{X})$ the *interior*, *boundary*, *closure*, and *convex hull* of $\mathcal{X}$, respectively. The Minkowski sum of $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^n$ is defined by

$$\mathcal{X} + \mathcal{Y} = \{x + y \mid x \in \mathcal{X}, y \in \mathcal{Y}\}.$$

If $\|\cdot\|$ is a norm on $\mathbb{R}^n$, its *dual norm* $\|\cdot\|_*$ is defined by

$$\forall y \in \mathbb{R}^n,\ \|y\|_* = \sup\left\{y^T x \;\middle|\; \|x\| \leq 1\right\}. \tag{1}$$

In all that follows, $\|\cdot\|_2$ denotes the Euclidean norm on $\mathbb{R}^n$. Finally, we denote by e a vector of all ones, and by e_j a vector whose j^{th} coordinate is 1 and all others are 0; the dimension of e and e_j is always obvious from context.

2.1 Cones and conic duality

The set $\mathcal{K} \subseteq \mathbb{R}^n$ is a *cone* if $\forall (x, \lambda) \in \mathcal{K} \times \mathbb{R}_+,\ \lambda x \in \mathcal{K}$, and it is *irreducible* if it cannot be written as a cartesian product of cones. The *dual cone* of $\mathcal{K} \subseteq \mathbb{R}^n$ is

$$\mathcal{K}^* = \left\{u \in \mathbb{R}^n \;\middle|\; u^T x \geq 0, \forall x \in \mathcal{K}\right\}, \tag{2}$$

and $\mathcal{K}$ is *self-dual* if $\mathcal{K} = \mathcal{K}^*$. A cone $\mathcal{K} \subseteq \mathbb{R}^n$ is *pointed* if $\mathcal{K} \cap (-\mathcal{K}) = \{0\}$, i.e., if it does not contain a line that passes through the origin. *Proper* cones are closed, convex, pointed cones with non-empty interior. If $\mathcal{K}$ is a proper cone, then $\mathcal{K}^*$ is also a proper cone, and any $\rho \in \text{int}\,\mathcal{K}$ induces a norm on $\mathcal{K}^*$, denoted by $|\cdot|_\rho$ and defined by

$$|u|_\rho = \rho^T u, \quad \forall u \in \mathcal{K}^*. \tag{3}$$

Examples of proper cones include the non-negative orthant

$$\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \mid x \geq 0\},$$

the second-order cone (SOC)

$$\mathcal{L}_n = \left\{ x \in \mathbb{R}^n \;\middle|\; x_1 \geq \sqrt{x_2^2 + ... + x_n^2} \right\},$$

the positive semi-definite (PSD) cone

$$\mathbb{S}^n_+ = \left\{ X \in \mathbb{R}^{n \times n} \;\middle|\; X = X^T, \lambda_{min}(X) \geq 0 \right\},$$

where $\lambda_{min}(X)$ is the smallest eigenvalue of X, and the exponential cone

$$\mathcal{E} = \mathrm{cl}\left\{ (x, y, z) \in \mathbb{R}^3 \;\middle|\; x \exp(x/y) \leq z, y > 0 \right\}.$$

The non-negative orthant, SOC and SDP cone are also self-dual, while the exponential cone is not.

A proper cone $\mathcal{K}$ induces a partial (resp. strict partial) ordering on $\mathbb{R}^n$, denoted $\succeq_{\mathcal{K}}$ (resp. $\succ_{\mathcal{K}}$) and defined by

$$\forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n,\ x \succeq_{\mathcal{K}} y \Leftrightarrow x - y \in \mathcal{K}, \tag{4}$$

$$\forall (x, y) \in \mathbb{R}^n \times \mathbb{R}^n,\ x \succ_{\mathcal{K}} y \Leftrightarrow x - y \in \mathrm{int}(\mathcal{K}). \tag{5}$$

In all that follows, we refer to $Ax \succeq_{\mathcal{K}} b$ (resp. $Ax \succ_{\mathcal{K}} b$) as a *conic* (resp. *strict conic*) inequality. Consider the system

$$Ax \succeq_{\mathcal{K}} b, \tag{6}$$

where $A \in \mathbb{R}^{m \times n}$ and $\mathcal{K} = \mathcal{K}_1 \times ... \times \mathcal{K}_N$ with $\mathcal{K}_i \subset \mathbb{R}^{m_i}$; correspondingly, for $y \in \mathbb{R}^m$, we write $y = (y_1, ..., y_N)$. We follow the terminology of [25], and say that system (6) is *feasible* if there exists $x \in \mathbb{R}^n$ such that $Ax - b \in \mathcal{K}$, and *strongly feasible* if there exists $x \in \mathbb{R}^n$ such that $Ax - b \in \mathcal{K}$ and $(Ax - b)_i \in \mathrm{int}(\mathcal{K}_i)$ for all non-polyhedral cones $\mathcal{K}_i$, i.e., such that all non-polyhedral conic inequalities are strictly satisfied. Similarly, system (6) is *infeasible* if it does not admit any feasible solution, and *strongly infeasible* if, in addition, there exists $y \in \mathcal{K}^*$ such that $A^T y = 0$ and $b^T y > 0$. Furthermore, we say that system (6) is *weakly feasible* if it is feasible but not strongly feasible, and *weakly infeasible* if it is infeasible but not strongly infeasible. Finally, a system is *well-posed* if it is either strongly feasible or strongly infeasible, and *ill-posed* otherwise.

Let us emphasize that well-posedness is an algebraic property, i.e., it is not associated to a geometric set but to its algebraic representation through conic inequalities. For instance, for $n \geq 3$, both $0 \succeq_{\mathbb{R}^n_+} x \succeq_{\mathbb{R}^n_+} 0$ and $0 \succeq_{\mathcal{L}_n} x \succeq_{\mathcal{L}_n} 0$ describe the same set $\{0\}$, however, the former is well-posed and the latter is not. Nevertheless, for brevity, we will refer to the well-posedness of a set $\mathcal{X}$, only if there no ambiguity in its description with conic inequalities.

A conic optimization problem writes, in standard form,

$$(P) \quad \min_{x} \quad c^T x \tag{7a}$$
$$s.t. \quad Ax = b, \tag{7b}$$
$$x \in \mathcal{K}, \tag{7c}$$

where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $\mathcal{K}$ is a cone. The strong/weak (in)feasibility and well-posedness of (P) refers to that of the system $(Ax = b, x \in \mathcal{K})$. The optimal value of (P) is $\text{opt}(P) = \inf\left\{c^T x \;\middle|\; Ax = b, x \in \mathcal{K}\right\}$, and we say that (P) is *bounded* if $\text{opt}(P) \in \mathbb{R}$ and *solvable* if, in addition, there exists a feasible solution x^* such that $c^T x^* = \text{opt}(P)$. The dual of (P) is

$$(D) \quad \max_{y,s} \quad b^T y \tag{8a}$$
$$s.t. \quad A^T y + s = c, \tag{8b}$$
$$s \in \mathcal{K}^*, \tag{8c}$$

and $\text{opt}(D) = \sup\left\{b^T y \;\middle|\; A^T y + s = c, s \in \mathcal{K}^*\right\}$. In particular, (D) is also a conic optimization problem.

Theorem 1 (Conic duality theorem)

1. *[Weak duality]* $\text{opt}(D) \leq \text{opt}(P)$.
2. *[Strong duality] If (P) (resp. (D)) is strongly feasible and bounded, then (D) (resp. (P)) is solvable and* $\text{opt}(P) = \text{opt}(D)$.
   If both (P) and (D) are strongly feasible, then both are solvable with same optimal value.

Proof See Theorem 1.4.4 in [11]. □

Conic duality extends the classical duality for linear programming, albeit with a number of edge cases that lead to practical difficulties. For instance, there may exist a positive duality gap even though both (P) and (D) are solvable, as illustrated in Example 1.

Example 1 (Example 8.6, [44]) Consider the primal-dual pair

$$\begin{array}{llll} (P) & \min\limits_{x_1,x_2,x_3} & x_3 & \\ & s.t. & x_2 \geq x_1, & \\ & & x_3 \geq -1, & \\ & & (x_1, x_2, x_3) \in \mathcal{L}_3, & \end{array} \qquad \begin{array}{llll} (D) & \max\limits_{y_1,y_2} & -y_2 & \\ & s.t. & (y_1, -y_1, 1 - y_2) \in \mathcal{L}_3, & \\ & & y_1, y_2 \geq 0. & \end{array}$$

Primal-feasible solutions are of the form $(x_1, x_1, 0)$, while dual-feasible solutions are of the form $(y_1, 1)$. Thus, $\text{opt}(P) = 0$ and $\text{opt}(D) = -1 < \text{opt}(P)$.

2.2 Valid inequalities

For $(\alpha, \beta) \in \mathbb{R}^n \times \mathbb{R}$, we say that $\alpha^T x \geq \beta$ is a *valid inequality* for $\mathcal{X} \subseteq \mathbb{R}^n$ if

$$\mathcal{X} \subseteq \left\{ x \in \mathbb{R}^n \;\middle|\; \alpha^T x \geq \beta \right\},$$

and a *supporting hyperplane* if, in addition, $\exists\, \tilde{x} \in \operatorname{cl}\operatorname{conv} \mathcal{X} : \alpha^T \tilde{x} = \beta$. The set of valid inequalities for $\mathcal{X}$ is denoted by $\mathcal{X}^{\#}$, i.e.,

$$\mathcal{X}^{\#} = \left\{ (\alpha, \beta) \in \mathbb{R}^n \times \mathbb{R} \;\middle|\; \forall x \in \mathcal{X},\ \alpha^T x \geq \beta \right\}. \tag{9}$$

Note that $\mathcal{X}^{\#}$ is a closed, convex set, and that

$$\left\{ x \in \mathbb{R}^n \;\middle|\; \forall (\alpha, \beta) \in \mathcal{X}^{\#},\ \alpha^T x \geq \beta \right\} = \operatorname{cl}\operatorname{conv}(\mathcal{X}). \tag{10}$$

Furthermore, for $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^n$, we have

$$\mathcal{X} \subseteq \mathcal{Y} \Rightarrow \mathcal{Y}^{\#} \subseteq \mathcal{X}^{\#},$$
$$(\mathcal{X} \cup \mathcal{Y})^{\#} = \mathcal{X}^{\#} \cap \mathcal{Y}^{\#}.$$

We now focus on the case where $\mathcal{X}$ is described by conic inequalities, and seek an algebraic description of $\mathcal{X}^{\#}$ using a finite number of conic inequalities. Note that, although $\mathcal{X}^{\#}$ is described by an infinite number of linear inequalities, as per Equation (9), this semi-infinite representation is not computationally effective.

Theorem 2 (Conic theorem on alternatives) *Consider the conic system*

$$Ax \succeq_{\mathcal{K}} b, \tag{11}$$

where $A \in \mathbb{R}^{m \times n}$ has full column rank and $\mathcal{K}$ is a proper cone.

1. *If there exists $y \in \mathbb{R}^m$ such that*

$$A^T y = 0, b^T y > 0, y \in \mathcal{K}^*, \tag{12}$$

   then (11) *has no solution.*
2. *If* (12) *has no solution, then* (11) *is almost solvable, i.e., for any $\epsilon > 0$, there exists $\tilde{b} \in \mathbb{R}^m$ such that $\|b - \tilde{b}\|_2 < \epsilon$ and the system $Ax \succeq_{\mathcal{K}} \tilde{b}$ is solvable.*
3. (12) *is solvable if and only if* (11) *is not almost solvable.*

Proof See Proposition 1.4.2 in [11]. □

Theorem 3 (Valid inequalities) *Let $\mathcal{C} = \{x \mid Ax \succeq_{\mathcal{K}} b\}$ with A of full column rank, and define*

$$\mathcal{F} = \left\{ (\alpha, \beta) \;\middle|\; \exists u \in \mathcal{K}^* : \alpha = A^T u, \beta \leq b^T u \right\}.$$

Then, $\operatorname{cl}\mathcal{F} \subseteq \mathcal{C}^{\#}$ and, in addition,

1. *if $\mathcal{C} \neq \emptyset$, then $\operatorname{cl}\mathcal{F} = \mathcal{C}^{\#}$;*

2. if $\mathcal{C}$ is well-posed, then $\mathcal{F} = \mathcal{C}^{\#}$.

Proof The inclusion $\mathcal{F} \subseteq C^{\#}$ is immediate from the definition of $\mathcal{K}^*$, and it follows that $\mathrm{cl}(\mathcal{F}) \subseteq \mathrm{cl}(C^{\#}) = C^{\#}$. Case **2** is a direct consequence of conic strong duality.

We now prove **1**. Assume $\mathcal{C} \neq \emptyset$, let $(\alpha, \beta) \in \mathcal{C}^{\#}$, and consider the systems

$$A^T u = \alpha,\ b^T u \geq \beta,\ u \in \mathcal{K}^*; \tag{13}$$

$$Ax \succeq_{\mathcal{K}} tb,\ \alpha^T x < t\beta,\ t \geq 0. \tag{14}$$

By Theorem 2, either (13) is almost solvable, or (14) is solvable. Let us prove that the latter does not hold.

Let (x, t) be a solution to (14). On the one hand, if $t > 0$, letting $\bar{x} = t^{-1}x$, we have $A\bar{x} \succeq_{\mathcal{K}} b$, i.e., $\bar{x} \in \mathcal{C}$, but $\alpha^T \bar{x} < \beta$, which contradicts $(\alpha, \beta) \in \mathcal{C}^{\#}$. On the other hand, if $t = 0$, then we have $Ax \succeq_{\mathcal{K}} 0$ and $\alpha^T x < 0$. Thus, for $x_0 \in \mathcal{C}$ and $\tau \geq 0$, we have

$$A(x_0 + \tau x) \succeq_{\mathcal{K}} b,$$

i.e., $(x_0 + \tau x) \in \mathcal{C}$. Furthermore, we have $\alpha^T(x_0 + \tau x) < \beta$ for large enough τ, which also contradicts $(\alpha, \beta) \in \mathcal{C}^{\#}$. Therefore, (14) is not solvable and (13) is almost solvable.

Thus, for any $\epsilon > 0$, there exists α_ϵ, β_ϵ and $u_\epsilon \in \mathcal{K}^*$ such that and

$$\|\alpha - \alpha_\epsilon\|_2 \leq \epsilon,\ \|\beta - \beta_\epsilon\|_2 \leq \epsilon,$$

and

$$\alpha_\epsilon = A^T u_\epsilon,\ \beta_\epsilon \leq b^T u_\epsilon,$$

i.e., $(\alpha_\epsilon, \beta_\epsilon) \in \mathcal{F}$. Taking $\epsilon \to 0$, we obtain that $(\alpha, \beta) \in \mathrm{cl}\,\mathcal{F}$. □

We will refer to the multiplier $u \in \mathcal{K}^*$ in Theorem 3 as a (conic) *Farkas multiplier*, and we say that $\alpha^T x \geq \beta$ is obtained by Farkas aggregation if $\alpha = A^T u$ and $\beta \leq b^T u$ for some $u \in \mathcal{K}^*$.

Theorem 3 highlights a fundamental difference between the linear and non-linear settings. In the linear case, $\mathcal{F}$ is polyhedral, thus, it is always closed and, if $\mathcal{C}$ is non-empty, then all valid inequalities for $\mathcal{C}$ can be obtained by Farkas aggregation. In the conic setting, however, this property may no longer hold, i.e., there may exist valid inequalities that cannot be represented through Farkas aggregation.

Example 2 Let $\mathcal{C} = \{x \mid Ax \succeq_{\mathcal{K}} 0\}$ where

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix},\ \mathcal{K} = \mathcal{L}_3,$$

i.e., $\mathcal{C} = \left\{(x,y) \in \mathbb{R}^2 \;\middle|\; (x,y,x) \in \mathcal{L}_3\right\}$. While A has full column rank and $\mathcal{K}$ is proper, the system $Ax \succeq_{\mathcal{K}} 0$ is not well-posed. It then is easy to verify that $\mathcal{C} = \{(x,0) \mid x \geq 0\}$, and that $y \geq 0$ is a valid inequality for $\mathcal{C}$.

However, for any $u \in \mathcal{K}^* = \mathcal{L}_3$, we have

$$A^T u = \begin{pmatrix} u_1 + u_3 \\ u_2 \end{pmatrix},$$

and $u_1 + u_3 > 0$ unless $u_2 = 0$. Therefore, the system $A^T u = (0,1), u \in \mathcal{K}^*$ has no solution, i.e., the valid inequality $y \geq 0$ cannot be obtained by Farkas aggregation.

Nevertheless, for $t \geq 0$, let $u_t = (\sqrt{t^2+1}, -t, 1)$, which yields the valid inequality $(\sqrt{t^2+1} - t)x + y \geq 0$. Then, as $t \to +\infty$, the term $(\sqrt{t^2+1} - t)$ becomes negligible and the inequality becomes, in the limit, $y \geq 0$.

2.3 Disjunctive inequalities

We now consider disjunctive conic sets, i.e., sets of the form

$$\mathcal{D} = \left\{ x \in \mathbb{R}^n \;\middle|\; \bigvee_{h=1}^{H} D_h x \succeq_{Q_h} d_h \right\} \tag{15}$$

$$= \bigcup_{h=1}^{H} \left\{ x \in \mathbb{R}^n \mid D_h x \succeq_{Q_h} d_h \right\}, \tag{16}$$

where $H \in \mathbb{Z}_+$ and, $\forall h$, $D_h \in \mathbb{R}^{m_h \times n}$ and $\mathcal{Q}_h$ is a proper cone. We refer to $\operatorname{conv} \mathcal{D}$ as the *disjunctive hull* and, for $(\alpha, \beta) \in \mathcal{D}^{\#}$, we say that $\alpha^T x \geq \beta$ is a *disjunctive inequality.*

We focus on disjunctive conic sets and, again, we seek a tractable algebraic characterization of valid inequalities for such sets. We begin by stating an extension to the conic setting of Balas' representation of the convex hull of a union of polyhedra [4].

Theorem 4 (Characterization of the convex hull) *Let*

$$\mathcal{D} = \bigcup_{h=1}^{H} \underbrace{\left\{ x \in \mathbb{R}^n \mid D_h x \succeq_{\mathcal{Q}_h} d_h \right\}}_{\mathcal{D}_h},$$

where, $\forall h$, $D_h \in \mathbb{R}^{m_h \times n}$ *and* $\mathcal{Q}_h$ *is a proper cone, and let*

$$\mathcal{S} = \left\{ x \in \mathbb{R}^n \;\middle|\; \exists (y_1, ..., y_H, z_1, ..., z_H) : \begin{array}{ll} \sum_h y_h = x, & \\ D_h y_h \succeq_{\mathcal{Q}_h} z_h d_h, & \forall h, \\ z_h \geq 0, & \forall h, \\ \sum_h z_h = 1, & \forall h, \\ y_h \in \mathbb{R}^n, & \forall h \end{array} \right\}.$$

Then $\operatorname{conv}(\mathcal{D}) \subseteq \mathcal{S}$ *and, in addition,*

1. *if* $\forall h, \mathcal{D}_h \neq \emptyset$, *then* $\mathcal{S} \subseteq \operatorname{cl}\operatorname{conv}(\mathcal{D})$;
2. *if* $\forall h, \mathcal{D}_h = \mathcal{X}_h + W$, *where* $\mathcal{X}_1, ..., \mathcal{X}_H$ *are non-empty, closed, bounded, convex sets and* W *is a closed convex set, then*

$$\operatorname{conv}(\mathcal{D}) = \mathcal{S} = \operatorname{cl}\operatorname{conv}(\mathcal{D}).$$

Proof See Proposition 3.3.5 in [11]. □

Next, building on Farkas multipliers and the result of Theorem 3, we can extend Balas' characterization of valid disjunctive inequalities (Theorem 3.1 in [4]) to the conic setting.

Theorem 5 (Disjunctive inequalities) *Let*

$$\mathcal{D} = \bigcup_{h=1}^{H} \underbrace{\{x \in \mathbb{R}^n \mid D_h x \succeq_{\mathcal{Q}_h} d_h\}}_{\mathcal{D}_h},$$

where, $\forall h$, $D_h \in \mathbb{R}^{m_h \times n}$ *and* $\mathcal{Q}_h$ *is a proper cone, and*

$$\mathcal{F} = \bigcap_{h=1}^{H} \underbrace{\left\{(\alpha, \beta) \in \mathbb{R}^n \times \mathbb{R} \;\middle|\; \exists u_h \in \mathcal{Q}_h^* : (\alpha = A^T u_h, \beta \leq b^T u_h)\right\}}_{\mathcal{F}_h}.$$

Then, $\mathcal{F} \subset \mathcal{D}^{\#}$ *and, in addition,*

1. *if* $\forall h, \mathcal{D}_h \neq \emptyset$, *then* $\mathcal{D}^{\#} = \bigcap_h \operatorname{cl} \mathcal{F}_h$;
2. *if* $\forall h, \mathcal{D}_h$ *is well-posed and* D_h *has full column rank, then* $\mathcal{F} = \mathcal{D}^{\#}$.

Proof Immediate from Theorem 3 and the fact that $\mathcal{D}^{\#} = \bigcap_h D_h^{\#}$. □

Example 3 Let

$$\mathcal{D} = \left\{(x, y) \in \mathbb{R}^2 \;\middle|\; (x, y, x) \in \mathcal{L}_3\right\} \cup \left\{(x, y) \in \mathbb{R}^2 \;\middle|\; (-x, y, x) \in \mathcal{L}_3\right\}.$$

Thus, $\mathcal{D} = \{(x, 0) | x \in \mathbb{R}\}$, and valid inequalities for $\mathcal{D}$ are of the form $\pm y \geq \pm\beta$ for $\beta \leq 0$. Building on Example 2, it follows that $\mathcal{F} = \{(0, 0)\} \times \mathbb{R}_-$. Therefore, only the trivial inequality $0 \geq -1$ can be represented using finite Farkas multipliers for each disjunctive term.

3 Cut separation

We consider an MI-CONIC problem of the form

$$(MICP) \quad \min_x \quad c^T x \tag{17a}$$
$$s.t. \quad Ax = b \tag{17b}$$
$$x \in \mathcal{K}, \tag{17c}$$
$$x \in \mathbb{Z}^p \times \mathbb{R}^{n-p}, \tag{17d}$$

where $A \in \mathbb{R}^{m\times n}$, $p \leq n$, and $\mathcal{K}$ is a proper cone. The continuous relaxation of $(MICP)$, denoted by (CP), is given by (17a)-(17c). The feasible sets of $(MICP)$ and (CP) are denoted by $\mathcal{X}$ and by $\mathcal{C}$, respectively.

Let $\bar{x} \in \mathbb{R}^n$ be a point to separate. Since $\bar{x}$ is typically obtained from solving a relaxation of $(MICP)$, we will assume that $A\bar{x} = b$, i.e., all linear equality constraints are satisfied. In particular, we will not assume that $\bar{x}$ is conic-feasible, i.e., we may have $\bar{x} \notin \mathcal{K}$, for instance when an outer-approximation algorithm is used.

Consider the disjunctive set

$$\mathcal{D} = \bigcup_{h=1}^{H} \mathcal{D}_h = \bigcup_{h=1}^{H} \left\{ x \;\middle|\; \begin{array}{l} Ax = b, x \in \mathcal{K} \\ D_h x \succeq_{\mathcal{Q}_h} d_h \end{array} \right\}, \tag{18}$$

where each $\mathcal{Q}_h$ is a proper cone, and $\mathcal{D} \supseteq \mathcal{X}$. Valid inequalities for $\mathcal{D}$ are referred to as *disjunctive inequalities* or, equivalently, as *disjunctive cuts*. For $(\alpha, \beta) \in \mathcal{D}^{\#}$, the inequality $\alpha^T x \geq \beta$ is *trivial* if $(\alpha, \beta) \in \mathcal{C}^{\#}$, and *non-trivial* otherwise. Following the terminology of [19], $\mathcal{K}^*$ *cuts* are trivial inequalities of the form $u^T x \geq 0$ for $u \in \mathcal{K}^*$. Finally, a cut is *violated* if $\alpha^T \bar{x} < \beta$.

It is always possible, e.g., through the use of facial reduction techniques [25], to describe each $\mathcal{D}_h$ by a well-posed system of conic inequalities. In addition, one may assume, after manual inspection, that $\forall h, \mathcal{D}_h \neq \emptyset$. However, in a cutting-plane context, systematically performing such reductions and verifications can quickly become intractable. Therefore, unless stated otherwise, we make no assumption regarding the feasibility nor well-posedness of individual disjunctive terms; we will show in Section 4 how to address such shortcomings in a systematic way.

3.1 Separation problem

Consider the *cut-generating conic problem* (CGCP)

$$\begin{aligned}
(CGCP) \quad \min_{\alpha,\beta,u,\lambda,v} \quad & \alpha^T \bar{x} - \beta & & & (19a)\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, & \forall h, & & (19b)\\
& \beta \leq b^T u_h + d_h^T v_h, & \forall h, & & (19c)\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, & \forall h, & & (19d)
\end{aligned}$$

which naturally extends Balas' CGLP to the conic setting, see also [18]. In particular, it is a conic programming problem, which can be solved by, e.g., an interior-point algorithm.

First, it follows from Theorem 5 that, if $(\alpha, \beta, u, \lambda, v)$ is feasible for (19), then $\alpha^T x \geq \beta$ is a disjunctive inequality. Under the stronger assumption of Case 2. in Theorem 5, every disjunctive inequality corresponds to a feasible solution of the CGCP. In the absence of such assumptions, however, the exact characterization of $\mathcal{D}^{\#}$ stated in Theorem 5 may not hold. For instance,

there may exist disjunctive inequalities that do not correspond to any feasible solution of the CGCP; as illustrated by Example 3, it may even be that all feasible solutions of the CGCP correspond to trivial inequalities of the form $0 \geq \beta$ for some $\beta \leq 0$.

Second, the feasible set of the CGCP is an unbounded cone, which contains the origin. Thus, the CGCP is either unbounded or bounded with objective value zero. In the former case, any unbounded ray yields a violated cut, while in the latter, no violated cut is obtained. Note that unbounded problems can lead to numerical issues for some interior-point algorithms. Therefore, it is common practice to add a normalization condition to the CGCP, whose role is further investigated in Section 4.

Third, the CGCP is strongly feasible. Indeed, let $\bar{u}_h = 0, \bar{v}_h \in \operatorname{int} \mathcal{Q}_h^*, \forall h$. Then, since $\mathcal{K}^*$ has non-empty interior, there exists $\bar{\alpha}$ such that

$$\bar{\alpha} \succ_{\mathcal{K}^*} D_h^T \bar{v}_h, \forall h.$$

Finally, letting $\bar{\lambda}_h = \bar{\alpha} - D_h^T \bar{v}_h \in \operatorname{int} \mathcal{K}^*$ and $\bar{\beta} \leq d_h^T \bar{v}_h, \forall h$, it follows that $(\bar{\alpha}, \bar{\beta}, \bar{u}, \bar{\lambda}, \bar{x})$ is strongly feasible for the CGCP.

Fourth, let $(\alpha, \beta, u, \lambda, v)$ be a feasible solution of the CGCP, and let

$$\tilde{\alpha} = \alpha - A^T u_0; \tilde{\beta} = \beta - b^T u_0; \tilde{u}_h = u_h - u_0, \forall h = 1, ..., H; \tilde{\lambda} = \lambda; \tilde{v} = v, \quad (20)$$

where $u_0 \in \mathbb{R}^m$. It is immediate to see that $(\tilde{\alpha}, \tilde{\beta}, \tilde{u}, \tilde{\lambda}, \tilde{v})$ is also feasible for the CGCP, with identical objective value since $A\bar{x} = b$. Therefore, without loss of generality, one of the u_h can be arbitrarily set to zero in the formulation of the CGCP, thereby reducing its size. Furthermore, since $\mathcal{C} \subseteq \{x \mid Ax = b\}$, it follows that

$$\mathcal{C} \cap \left\{x \;\middle|\; \alpha^T x \geq \beta\right\} = \mathcal{C} \cap \left\{x \;\middle|\; \tilde{\alpha}^T x \geq \tilde{\beta}\right\}, \quad (21)$$

i.e., the two inequalities are equivalent in the sense that both cut off the same portion of the continuous relaxation.

3.2 Membership problem

The dual problem of the CGCP is the *membership conic problem* (MCP)

$$\begin{aligned}
(MCP) \quad \max_{y,z} \quad & 0 && && (22a)\\
s.t. \quad & \sum_h y_h = \bar{x}, && && (22b)\\
& \sum_h z_h = 1, && && (22c)\\
& Ay_h = z_h b, && \forall h, && (22d)\\
& D_h y_h \succeq_{\mathcal{Q}_h} z_h d_h, && \forall h, && (22e)\\
& (y_h, z_h) \in \mathcal{K} \times \mathbb{R}_+, && \forall h, && (22f)
\end{aligned}$$

which extends Bonami's membership LP [15] to the conic setting.

A geometrical interpretation of the MCP is provided by Theorem 4. If all disjunctive terms are non-empty and have identical recession cones, then (22) is feasible if and only if $\bar{x} \in \operatorname{conv}(\mathcal{D})$. In the general case, however, the exact characterization of $\operatorname{conv}\mathcal{D}$ given by Theorem 4 may no longer hold, and we can only state that, if $\bar{x} \in \operatorname{conv}\mathcal{D}$, then the MCP is feasible.

By weak duality, if the MCP is feasible, then the objective value of the CGCP is bounded below. If, in addition, the MCP is strongly feasible, then both the MCP and the CGCP are solvable with identical objective values.

4 The roles of normalization

This section focuses on the roles of the normalization condition in the CGCP. On the one hand, through the lens of conic duality for the CGCP-MCP pair, we investigate the impact of the normalization on the solvability of CGCP. On the other hand, by characterizing optimal solutions of the normalized CGCP, we assess the theoretical properties of the corresponding cuts.

The following normalization conditions are considered:

(i) the α normalization: $\|\alpha\|_* \leq 1$,
(ii) the *polar* normalization: $\gamma^T\alpha \leq 1$,
(iii) the *standard* normalization: $\sum_h |\lambda_h|_\rho + |v_h|_{\sigma_h} \leq 1$,
(iv) the *trivial* normalization: $\sum_h |v_h|_{\sigma_h} \leq 1$,
(v) the *uniform* normalization: $\sum_h |\lambda_h|_\rho \leq 1$,

where $\gamma \in \mathbb{R}^n$, $\rho \in \operatorname{int}\mathcal{K}$ and $\sigma_h \in \operatorname{int}\mathcal{Q}_h$. While the first four normalizations can be found in the MILP literature in an appropriate form, see, e.g., [23,30], to the best of our knowledge, the proposed uniform normalization has never been studied, even in the linear setting.

Throughout this section, the strengths and shortcomings of each normalization are illustrated in a simple setting, described in Example 4 below.

Example 4 For $R > 0$, consider the MICP

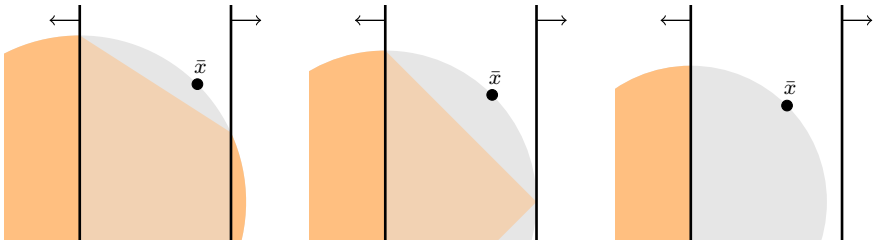
$$\min_x \quad -x_1 - x_2 \tag{23}$$
$$s.t. \quad x_0 = R, \tag{24}$$
$$x \in \mathcal{L}_3, \tag{25}$$
$$x_1, x_2 \in \mathbb{Z}, \tag{26}$$

and the split disjunction $(x_1 \leq 0) \vee (x_1 \geq 1)$. Thus, we have

$$\mathcal{D}_1 = \left\{x \in \mathbb{R}^3 \;\middle|\; x \in \mathcal{L}_3, x_0 = R, x_1 \leq 0\right\},$$
$$\mathcal{D}_2 = \left\{x \in \mathbb{R}^3 \;\middle|\; x \in \mathcal{L}_3, x_0 = R, x_1 \geq 1\right\}.$$

We consider the following three cases:

(a) $R > 1$: $\mathcal{D}_1$ and $\mathcal{D}_2$ are both strongly feasible;



(a) $R = 1.1$ (b) $R = 1.0$ (c) $R = 0.9$

Fig. 1: The three settings from Example 4, projected onto the $x_0 = 1$ space. The domain of the continuous relaxation is in gray, the split hull in orange, and the split disjunction is indicated in black.

(b) $R = 1$: $\mathcal{D}_1$ is strongly feasible and $\mathcal{D}_2$ is weakly feasible;
(c) $R < 1$: $\mathcal{D}_1$ is strongly feasible and $\mathcal{D}_2 = \emptyset$.

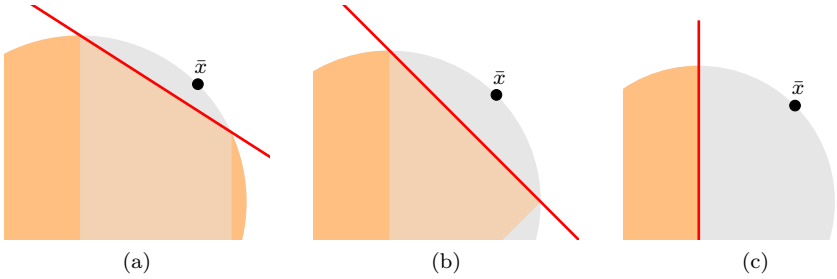
Each of these settings is illustrated in Figure 1. Finally, unless specified otherwise, $\bar{x}$ is the solution of the continuous relaxation, i.e., $\bar{x} = (R, \frac{R}{\sqrt{2}}, \frac{R}{\sqrt{2}})$. All CGCPs are solved as conic problems using Mosek 9.2 with default parameters.

4.1 α normalization

A straightforward way of bounding the CGCP is to restrict the magnitude of α. This approach was considered in previous work on MI-CONV [17,48,31], wherein authors considered restricting the ℓ_1, ℓ_∞ or ℓ_2 norm of α.

For a given norm $\|\cdot\|$, the CGCP then writes

$$\begin{aligned}
\min_{\alpha,\beta,\lambda,u,v} \quad & \alpha^T \bar{x} - \beta && && (27a)\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, && \forall h, && (27b)\\
& \beta \leq b^T u_h + d_h^T v_h, && \forall h, && (27c)\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, && \forall h, && (27d)\\
& \|\alpha\|_* \leq 1, && && (27e)
\end{aligned}$$



(a) (b) (c)

Fig. 2: Split cuts (in red) obtained with the α normalization.

Table 1: CGCP statistics for Example 4 and α normalization

	Iter	$\|\alpha\|$	$\|u_1\|$	$\|u_2\|$	$\|\lambda_1\|$	$\|\lambda_2\|$	$\|v_1\|$	$\|v_2\|$
(a)	8	1.0	0.8	2.0	1.2	2.9	0.5	1.3
(b)	67*	1.0	0.7	10958.8	1.0	15498.0	0.7	10958.1
(c)	8	1.0	0.0	13.7	0.0	19.3	1.0	12.6

*: slow progress

and the corresponding MCP, up to a change of sign in the objective value, is

$$\min_{x,y,z} \quad \|\bar{x} - x\| \tag{28a}$$
$$s.t. \quad \sum_h y_h = x, \tag{28b}$$
$$\sum_h z_h = 1, \tag{28c}$$
$$A y_h = z_h b, \qquad \forall h, \tag{28d}$$
$$D_h y_h \succeq_{\mathcal{Q}_h} z_h d_h, \qquad \forall h, \tag{28e}$$
$$(y_h, z_h) \in \mathcal{K} \times \mathbb{R}_+, \qquad \forall h. \tag{28f}$$

Geometrically, the CGCP (27) looks for a deepest cut, i.e., one that maximizes the distance from $\bar{x}$ to the hyperplane $\alpha^T x = \beta$, as measured by $\|\cdot\|$. Correspondingly, the MCP (28) computes a projection of $\bar{x}$ onto the set defined by (28b)-(28f), with respect to $\|\cdot\|$. It is trivially feasible if at least one of the disjunctive terms is non-empty, which is always the case if $\mathcal{X} \neq \emptyset$. This ensures that the CGCP (27) is never unbounded. If, in addition, each disjunctive term is strongly feasible, then the MCP (28) is strongly feasible and both the MCP and the CGCP are solvable.

Split cuts obtained with the α normalization in the context of Example 4 are illustrated in Figure 2; these results are obtained with the (self-dual) ℓ_2 norm in (27e). Furthermore, some statistics regarding the resolution of the CGCP are reported in Table 1, namely: the number of interior-point iterations (Iter), and the magnitude of α, u, λ, v in the obtained CGCP solution.

In Example 4(a), the MCP is strongly feasible, no numerical trouble is encountered, and the obtained cut is a supporting hyperplane of the split hull. On the other hand, numerical issues are encountered in Example 4(b). Indeed, as reported in Table 1, Mosek terminates due to slow progress after 67 iterations, albeit with a feasible solution, for which the magnitude of u_2, λ_2, v_2 is very large. The corresponding cut is displayed in Figure 2b. Finally, in Example 4(c), although $\mathcal{D}_2 = \emptyset$, the CGCP is solved without issue, thereby showing that it is not necessary for the MCP to be strongly feasible for the CGCP to be solvable.

Example 4(b) illustrates the numerical challenges that may arise when the MCP is not strongly feasible. Indeed, in that case, the deepest cut writes $x_1 + x_2 \leq 1$, up to a positive scaling factor. However, although $x_1 + x_2 \leq 1$ is a valid inequality for $\mathcal{D}_2$, it is straightforward to see that it cannot be represented using finite Farkas multipliers: this corresponds to case **1** in Theorem 3. Thus, the CGCP has no optimal solution, and there exists a (diverging) sequence of feasible solutions whose objective value becomes arbitrary close to opt$(CGCP)$, corresponding to a sequence of valid inequalities that, in the limit, become equivalent to $x_1 + x_2 \leq 1$. Hence, the solver eventually runs into slow progress while the magnitude of u_2, λ_2, v_2 becomes large, as observed in Table 1.

Interestingly, all cuts displayed in Figure 2 are supporting hyperplanes of the split hull. A slightly more general result is stated in Theorem 6.

Theorem 6 *Assume that the CGCP* (27) *and the MCP* (28) *are solvable, and let* $(\alpha, \beta, u, \lambda, v)$ *and* (x, y, z) *be corresponding optimal solutions. Then,*

$$\alpha^T x = \beta.$$

Proof Let δ be the optimal value of the CGCP, i.e., $\delta = \alpha^T \bar{x} - \beta$. Since the CGCP is strongly feasible and bounded, by Theorem 1, strong duality holds. Thus, we have $\delta = -\|\bar{x} - x\|$. The case $\delta = 0$ is trivial, so we assume $\delta < 0$ and, thus, $x \neq \bar{x}$.

Let $w = |\delta|^{-1}(x - \bar{x})$, i.e., $x = \bar{x} + |\delta| w$ and $\|w\| = 1$; in particular, we have $1 \geq \|\alpha\|_* \geq \alpha^T w$. It follows that

$$\begin{aligned} \alpha^T x - \beta &= \alpha^T \bar{x} + |\delta| \alpha^T w - \beta \\ &= \delta + |\delta| \alpha^T w \\ &= \delta(1 - \alpha^T w) \\ &\leq 0. \end{aligned}$$

Thus, $\alpha^T x \leq \beta$.

Next, we have

$$\begin{aligned}\alpha^T x &= \sum_h \alpha^T y_h \\ &= \sum_h u_h^T A y_h + \lambda_h^T y_h + v_h^T D_h y_h \\ &\geq \sum_h u_h^T (z_h b) + v_h^T (z_h d_h) \\ &\geq \sum_h z_h \beta \\ &= \beta,\end{aligned}$$

which concludes the proof. □

Therefore, if all disjunctive terms are non-empty, then, by Theorem 4, $x \in \operatorname{cl}\operatorname{conv}\mathcal{D}$, and the obtained inequality $\alpha^T x \geq \beta$ is indeed a supporting hyperplane of the disjunctive hull.

4.2 Polar normalization

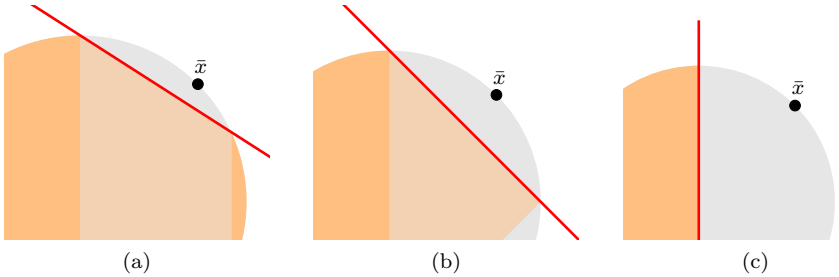
In the MILP setting, Balas and Perregard [45,8] first suggest normalizing the CGLP with a single hyperplane of the form $\alpha^T \gamma = 1$. Doing so ensures that, if the CGLP is feasible and bounded, then there exists an optimal solution for which (α, β) is an extreme ray of $(\operatorname{cl}\operatorname{conv}\mathcal{D})^{\#}$. This approach was then followed in [16,47], and more recently in [20].

For $\gamma \in \mathbb{R}^n$, the CGCP writes

$$\begin{aligned}
\min_{\alpha,\beta,u,v,\lambda} \quad & \alpha^T \bar{x} - \beta && && (29a)\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, && \forall h, && (29b)\\
& \beta \leq b^T u_h + d_h^T v_h, && \forall h, && (29c)\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, && \forall h, && (29d)\\
& \alpha^T \gamma \leq 1, && && (29e)
\end{aligned}$$

and the MCP is given by

$$\begin{aligned}
\min_{y,z,\eta} \quad & \eta && && (30a)\\
s.t. \quad & \sum_h y_h = \bar{x} + \eta\gamma, && && (30b)\\
& \sum_h z_h = 1, && && (30c)\\
& A y_h = z_h b, && \forall h, && (30d)\\
& D_h y_h \succeq_{\mathcal{Q}_h} z_h d_h, && \forall h, && (30e)\\
& (y_h, z_h) \in \mathcal{K} \times \mathbb{R}_+, && \forall h, && (30f)\\
& \eta \geq 0. && && (30g)
\end{aligned}$$



(a) (b) (c)

Fig. 3: Split cuts (in red) obtained with the polar normalization.

Table 2: CGCP statistics for Example 4 and polar normalization

	Iter	$\|\alpha\|$	$\|u_1\|$	$\|u_2\|$	$\|\lambda_1\|$	$\|\lambda_2\|$	$\|v_1\|$	$\|v_2\|$
(a)	8	1.2	0.0	1.1	1.1	2.7	0.5	1.2
(b)	66*	1.2	0.0	9912.6	1.0	14019.5	0.7	9912.6
(c)	8	1.6	0.0	20.5	0.0	28.9	1.6	18.8

*: slow progress

It follows from Theorem 4 that, if there exists $\eta \geq 0$ such that $(\bar{x} + \eta\gamma) \in \operatorname{conv} \mathcal{D}$, then the MCP (30) is feasible. This is always the case if $\gamma = x^* - \bar{x}$, for some $x^* \in \operatorname{conv} \mathcal{D}$. If, in addition, each individual disjunction is strongly feasible and x^* is obtained as a convex combination of strongly feasible points, then the MCP (30) is strongly feasible.

Split cuts obtained for Example 4 with the polar normalization are illustrated in Figure 3, and the corresponding CGCP statistics are reported in Table 2. In each case, we set $\gamma = x^* - \bar{x}$, where $x^* = (R, 0, 0)$.

In all three cases, the obtained cut is identical to the one obtained with the α normalization, although this is not the case in general. Furthermore, as reported in Table 2, numerical issues are also encountered for Example 4(b), for the same reasons as for the α normalization: the CGCP is not solvable, and Mosek terminates with slow progress while the magnitude of u_2, λ_2, v_2 diverges.

Similar to the α-normalization, if all disjunctive terms are non-empty, then cuts obtained with the polar normalization are also supporting hyperplanes of the disjunctive hull, as expressed by Theorem 7.

Theorem 7 *Assume that the CGCP* (29) *and the MCP* (30) *are solvable, and let* $(\alpha, \beta, u, \lambda, v)$ *and* (η, y, z) *be corresponding optimal solutions. Then,*

$$\alpha^T(\bar{x} + \eta\gamma) = \beta.$$

Proof By conic strong duality, we have $\alpha^T \bar{x} - \beta = -\eta$. If the optimal value of the CGCP is 0, then the result is trivial. Similarly, if $\alpha^T \gamma \leq 0$, then the optimal value of the CGCP must be 0 and the result is trivial.

We now assume that $\alpha^T\gamma > 0$ and $\text{opt}(CGCP) < 0$. Since $(\alpha, \beta, u, \lambda, v)$ is optimal for the CGCP, we must have $\alpha^T\gamma = 1$. Then,

$$\begin{aligned}\alpha^T\bar{x} - \beta &= -\eta \\ &= -(\alpha^T\gamma)\eta,\end{aligned}$$

i.e., $\alpha^T(\bar{x} + \eta\gamma) = \beta$. □

4.3 Standard normalization

One of the most common normalizations in the MILP setting is the so-called *standard normalization* [4,9,23]. Here, we introduce its conic generalization

$$\sum_h |\lambda_h|_\rho + |v_h|_{\sigma_h} \leq 1, \tag{31}$$

where $\rho \in \text{int}\,\mathcal{K}$ and $\sigma_h \in \text{int}\,\mathcal{Q}_h$. When all cones $\mathcal{K}, \mathcal{Q}_1, ..., \mathcal{Q}_H$ are non-negative orthants, (31) indeed generalizes both the standard normalization for MILP and its *Euclidean normalization* variant [23], by setting ρ and σ appropriately.

The CGCP then writes

$$\begin{aligned}
\min_{\alpha,\beta,u,v,\lambda} \quad & \alpha^T\bar{x} - \beta && && \text{(32a)}\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, && \forall h, && \text{(32b)}\\
& \beta \leq b^T u_h + d_h^T v_h, && \forall h, && \text{(32c)}\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, && \forall h, && \text{(32d)}\\
& \sum_h |\lambda_h|_\rho + |v_h|_{\sigma_h} \leq 1. && && \text{(32e)}
\end{aligned}$$

Note that (32e) consists of a single linear inequality in the λ, v space, and that it explicitly bounds their magnitude.

Up to a change of sign in the objective value, the MCP is

$$\begin{aligned}
\min_{y,z,\eta} \quad & \eta && && \text{(33a)}\\
s.t. \quad & \sum_h y_h = \bar{x}, && && \text{(33b)}\\
& \sum_h z_h = 1, && && \text{(33c)}\\
& Ay_h = z_h b, && \forall h, && \text{(33d)}\\
& D_h y_h + \eta\sigma_h \succeq_{\mathcal{Q}_h} z_h d_h, && \forall h, && \text{(33e)}\\
& (y_h + \eta\rho, z_h) \in \mathcal{K} \times \mathbb{R}_+, && \forall h, && \text{(33f)}\\
& \eta \geq 0. && && \text{(33g)}
\end{aligned}$$

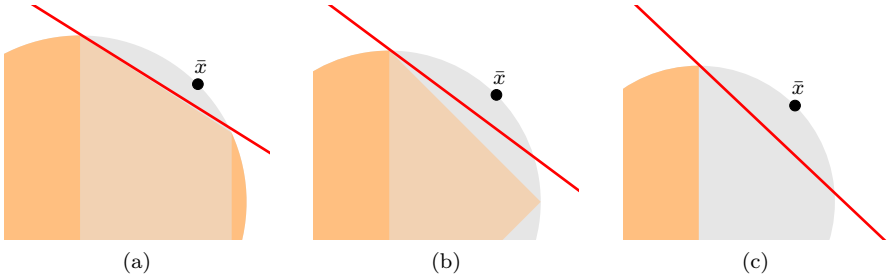


Fig. 4: Split cuts (in red) obtained with the standard normalization.

The dual counterpart of the standard normalization corresponds to penalizing the violation of the original conic constraints (22f) and (22e), through the artificial slack variable η in (33f) and (33e). As shown in Theorem 8, this ensures that the MCP is strongly feasible.

Theorem 8 *The CGCP* (32) *and the MCP* (33) *are strongly feasible.*

Proof It was shown in Section 3.1 that the un-normalized CGCP (19) admits a strongly feasible solution. Scaling this point appropriately yields a strongly feasible solution for the CGCP (32).

We now show that the MCP (33) is strongly feasible. Let $y_h = \frac{1}{H}\bar{x}$ and $z_h = \frac{1}{H}$, $\forall h$. Then, since $\mathcal{K}, \mathcal{Q}_1, ..., \mathcal{Q}_H$ are proper cones and $\rho \in \operatorname{int}\mathcal{K}, \sigma_h \in \operatorname{int}\mathcal{Q}_h$, there exists $\eta_h \geq 0$ and $\tau_h \geq 0$ such that

$$\begin{aligned} &\forall h,\ y_h + \eta_h \rho \succ_{\mathcal{K}} 0, \\ &\forall h,\ D_h y_h + \tau_h \sigma_h \succ_{\mathcal{Q}_h} d_h. \end{aligned}$$

Letting $\eta = \max(\eta_1, ..., \eta_H, \tau_1, ..., \tau_H)$ then yields a strongly feasible point (y, z, η), which concludes the proof. □

A direct consequence of Theorem 8 is that both the CGCP (32) and the MCP (33) are solvable, and that conic strong duality holds. Importantly, aside from the assumption $A\bar{x} = b$, this result does not depend on $\bar{x}$, nor on the well-posedness of individual disjunctions.

Split cuts obtained with the standard normalization are illustrated in Figure 4, and the corresponding CGCP statistics are reported in Table 3. On the one hand, Table 3 illustrate the good numerical behavior of the CGCP (32), a direct consequence of having enforced strong feasibility of the MCP. On the other hand, the obtained cuts are not as strong as the ones obtained with the α or polar normalizations. Indeed, in general, the cut obtained with the standard normalization may not be a supported hyperplane of the disjunctive hull.

Table 3: CGCP statistics for Example 4 and standard normalization

	Iter	$\|\alpha\|$	$\|u_1\|$	$\|u_2\|$	$\|\lambda_1\|$	$\|\lambda_2\|$	$\|v_1\|$	$\|v_2\|$
(a)	8	0.4	0.0	0.2	0.3	0.6	0.1	0.2
(b)	8	0.3	0.0	0.2	0.3	0.6	0.1	0.2
(c)	7	0.3	0.0	0.3	0.3	0.6	0.1	0.2

4.4 Trivial normalization

The trivial normalization is obtained by setting $\rho = 0$ in the standard normalization. The separation problem then writes

$$\begin{aligned}
\min_{\alpha,\beta,u,v,\lambda} \quad & \alpha^T \bar{x} - \beta && && (34a)\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, && \forall h, && (34b)\\
& \beta \leq b^T u_h + d_h^T v_h, && \forall h, && (34c)\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, && \forall h, && (34d)\\
& \sum_h |v_h|_{\sigma_h} \leq 1. && && (34e)
\end{aligned}$$

Note that, for a split disjunction $(-\pi^T x \geq -\pi_0) \vee (\pi^T x \geq \pi_0 + 1)$ and $\sigma_h = 1$, (34e) reduces to

$$v_1 + v_2 \leq 1, \tag{35}$$

which is the so-called trivial normalization [23] in the MILP setting. In particular, Gomory Mixed-Integer cuts correspond to optimal solutions of the CGLP with trivial normalization.

Up to a change of sign in the objective value, the MCP writes

$$\begin{aligned}
\min_{y,z,\eta} \quad & \eta && && (36a)\\
s.t. \quad & \sum_h y_h = \bar{x}, && && (36b)\\
& \sum_h z_h = 1, && && (36c)\\
& Ay_h = z_h b, && \forall h, && (36d)\\
& (y_h, z_h) \in \mathcal{K} \times \mathbb{R}_+, && \forall h, && (36e)\\
& D_h y_h + \eta \sigma_h \succeq_{\mathcal{Q}_h} z_h d_h, && \forall h, && (36f)\\
& \eta \geq 0. && && (36g)
\end{aligned}$$

Theorem 9 *The MCP* (36) *is*

1. *strongly infeasible if and only if $\bar{x}$ is infeasible for (CP);*
2. *strongly feasible if and only if $\bar{x}$ is strongly feasible for (CP);*
3. *weakly feasible if and only if $\bar{x}$ is weakly feasible for (CP).*

Proof Since $\bar{x}$ is either infeasible, strongly feasible or weakly feasible for (CP), and that these are mutually exclusive alternatives, it suffices to prove that the above conditions are sufficient; necessity follows immediately by contraposition. For simplicity, we assume that $\mathcal{K}$ is an irreducible, non-polyhedral cone. The general case is treated similarly.

1. Assume $\bar{x} \notin \mathcal{C}$, i.e., $\bar{x} \notin \mathcal{K}$ since we assumed that $A\bar{x} = b$. Consequently, there exists $s \in \mathcal{K}^*$ such that $s^T\bar{x} < 0$ and, since $\mathcal{K}$ is a proper cone, we can assume without loss of generality that $s \in \operatorname{int}\mathcal{K}^*$. Thus, setting

$$\begin{aligned} (\alpha, \beta) &= (s, 0) \\ (u_h, \lambda_h, v_h) &= (0, s, 0), \qquad\qquad \forall h, \end{aligned}$$

yields a strongly feasible solution of the CGCP (34), which is also an unbounded ray. Therefore, the MCP is strongly infeasible.

2, 3. We first show that, if $\bar{x}$ is feasible for (CP), then the MCP is feasible. Let

$$\begin{aligned} \tilde{y}_h &= H^{-1}\bar{x}, \qquad\qquad \forall h, \\ \tilde{z}_h &= H^{-1}, \qquad\qquad \forall h. \end{aligned}$$

It is then immediate that $A\tilde{y}_h = \tilde{z}_h b$, $\tilde{y}_h \in \mathcal{K}$, and $\sum_h \tilde{z}_h = 1$. Then, since $\sigma_h \in \operatorname{int}\mathcal{Q}_h$, there exists $\eta_h \geq 0$ such that

$$D_h y_h + \eta_h \sigma_h \succ_{\mathcal{Q}_h} z_h d_h.$$

Letting $\tilde{\eta} = \max_h\{\eta_h\}$, it follows that $(\tilde{y}, \tilde{z}, \tilde{\eta})$ is feasible for the MCP.

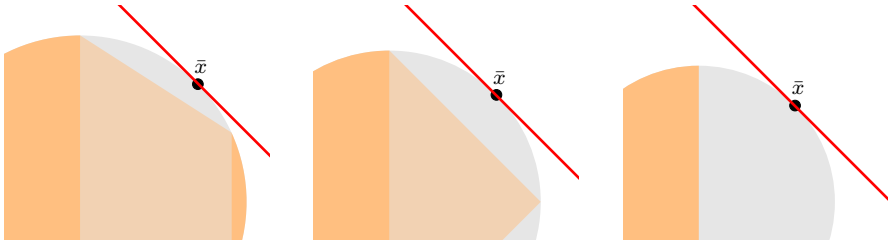
If $\bar{x} \in \operatorname{int}\mathcal{K}$, then we also have $\tilde{y}_h \in \operatorname{int}\mathcal{K}$, and thus $(\tilde{y}, \tilde{z}, \tilde{\eta})$ is strongly feasible for the MCP, which proves **2.** Reciprocally, let (y, z, η) be a strongly feasible solution of MCP. In particular, we have $y_h \in \operatorname{int}\mathcal{K}$. Then,

$$\bar{x} = \sum_h y_h \in \operatorname{int}\mathcal{K},$$

i.e., $\bar{x}$ is strongly feasible for (CP), thereby proving **3.** by contraposition. □

Theorem 9 motivates the following remarks. First, case **1.** typically arises in the context of outer-approximation algorithms, wherein fractional points generally violate non-linear conic constraints. Then, the CGCP (34) is unbounded, and the normalization (34e) imposes $|v_h|_{\sigma_h} = 0, \forall h$ in any unbounded ray, i.e., $v_h = 0$ and the obtained cut is always a trivial inequality. Thus, the trivial normalization is not suited for use within outer approximation-based algorithms.

Second, conic-infeasible points are not encountered in non-linear branch-and-bound algorithms. However, solving (CP) yields a fractional point $\bar{x} \in \partial\mathcal{C}$ that is weakly feasible, unless all non-linear conic constraints are inactive at the optimum. Although the current fractional point $\bar{x}$ may become strongly feasible after several rounds of cuts, or deeper in the branch-and-bound tree, our experience is that case **2.** rarely occurs in practice.



(a) (b) (c)

Fig. 5: Split cuts (in red) obtained with the trivial normalization.

Table 4: CGCP statistics for Example 4 and trivial normalization

	Iter	$\|\alpha\|$	$\|u_1\|$	$\|u_2\|$	$\|\lambda_1\|$	$\|\lambda_2\|$	$\|v_1\|$	$\|v_2\|$
(a)	37*	6189.7	0.0	0.7	6189.6	6190.6	0.2	0.8
(b)	49*	6814.2	0.0	0.7	6814.0	6815.0	0.3	0.7
(c)	72*	6364.4	0.0	0.7	6364.2	6365.2	0.4	0.6

*: slow progress

Third, case **3.** corresponds to the setting of Example 4. Split cuts obtained with the trivial normalization are displayed in Figure 5, and the corresponding CGCP statistics are reported in Table 4. Remarkably, all three cuts appear to be $\mathcal{K}^*$ cuts, while numerical issues, namely, slow convergence, are systematically encountered. Here, the MCP (36) is weakly feasible and the CGCP (34) is bounded but not solvable. Thus, there exists a diverging sequence of close-to-optimal solution, thereby causing slow convergence. In addition, since (34e) bounds the value of v_1, v_2, the iterates become equivalent to a $\mathcal{K}^*$ cut as the magnitude of λ increases, which explains the cuts obtained in Figure 5.

Note that the CGCP (34) may be solvable even though $\bar{x}$ is weakly feasible. However, our experience suggests that this rarely happens, and that most cases are similar to Example 4, leading to numerical issues and weak cuts.

4.5 Uniform normalization

The uniform normalization is obtained by setting $\sigma_h = 0$ in the standard normalization.

The CGCP then writes

$$\begin{aligned}
\min_{\alpha,\beta,u,v,\lambda} \quad & \alpha^T \bar{x} - \beta && && (37a)\\
s.t. \quad & \alpha = A^T u_h + \lambda_h + D_h^T v_h, && \forall h, && (37b)\\
& \beta \leq b^T u_h + d_h^T v_h, && \forall h, && (37c)\\
& (u_h, \lambda_h, v_h) \in \mathbb{R}^m \times \mathcal{K}^* \times \mathcal{Q}_h^*, && \forall h, && (37d)\\
& \sum_h |\lambda_h|_\rho \leq 1, && && (37e)
\end{aligned}$$

and, up to a change of sign in the objective, the MCP is

$$\begin{aligned}
\min_{y,z,\eta} \quad & \eta && && (38a)\\
s.t. \quad & \sum_h y_h = \bar{x}, && && (38b)\\
& \sum_h z_h = 1, && && (38c)\\
& Ay_h = z_h b, && \forall h, && (38d)\\
& (y_h + \eta\rho, z_h) \in \mathcal{K} \times \mathbb{R}_+, && \forall h, && (38e)\\
& D_h y_h \succeq_{\mathcal{Q}_h} z_h d_h, && \forall h, && (38f)\\
& \eta \geq 0. && && (38g)
\end{aligned}$$

As illustrated by Example 5, in general, the MCP (38) may not be feasible.

Example 5 Let

$$\mathcal{C} = \left\{(x_1, x_2) \in \mathbb{R}^2_+ \mid x_1 + x_2 = 1\right\},$$

and consider the disjunction

$$\{x_1 \geq 0, -x_1 \geq 0\} \vee \{x_1 \geq 1, -x_1 \geq -1\}.$$

Constraints (38d) and (38f) first yield

$$y_1 = \begin{pmatrix} 0 \\ z_1 \end{pmatrix}, \quad y_2 = \begin{pmatrix} z_2 \\ 1 - z_2 \end{pmatrix},$$

which, combined with (38c) and (38b), yields $\bar{x} = (1 - z_1, z_1)$ for $0 \leq z_1 \leq 1$. Therefore, if $\bar{x} = (-1, 2)$, then the MCP (38) is infeasible.

Nevertheless, the following results demonstrate that, for certain classes of disjunctions, namely split disjunctions, strong feasibility in the MCP is guaranteed.

Lemma 1 *If*

$$\bar{x} \in \operatorname{conv}\left(\bigcup_h \{x \mid Ax = b, D_h x \succeq_{\mathcal{Q}_h} d_h\}\right),$$

then the MCP (38) *is feasible.*

Proof Immediate from Theorem 4. □

Lemma 2 *Assume that* $\forall h,\ \mathcal{Q}_h$ *is polyhedral. Then, the MCP* (38) *is strongly feasible if and only if it is feasible.*

Proof Strong feasibility implies feasibility. Reciprocally, if (x, y, z, η) is feasible for the MCP (38), then $(x, y, z, \eta+\epsilon)$ is strongly feasible for any $\epsilon > 0$, because $\rho \in \operatorname{int} \mathcal{K}$. □

Theorem 10 *If* $\mathcal{X} \neq \emptyset$ *and* $\mathcal{D}$ *is a split disjunction, i.e.,*

$$\mathcal{D} = \left\{ x \;\middle|\; \begin{array}{l} Ax = b, x \in \mathcal{K} \\ \pi^T x \leq \pi_0 \end{array} \right\} \cup \left\{ x \;\middle|\; \begin{array}{l} Ax = b, x \in \mathcal{K} \\ \pi^T x \geq \pi_0 + 1 \end{array} \right\},$$

then the MCP (38) *is strongly feasible.*

Proof Let

$$\mathcal{S} = \operatorname{conv}\left(\left\{ x \mid Ax = b, \pi^T x \leq \pi_0 \right\} \cup \left\{ x \mid Ax = b, \pi^T x \geq \pi_0 + 1 \right\} \right).$$

First, assume there exists $\xi \in \mathbb{R}^n$ such that $A\xi = 0$ and $\pi^T \xi \neq 0$; without loss of generality, we can assume that $\pi^T \xi = 1$. Then, let $t \geq 0$ such that

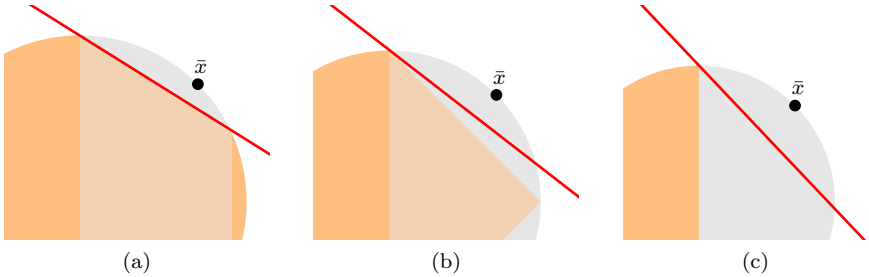
$$\begin{aligned} \pi^T(\bar{x} - t\xi) &\leq \pi_0, \\ \pi^T(\bar{x} + t\xi) &\geq \pi_0 + 1. \end{aligned}$$

In particular, $A(\bar{x} \pm t\xi) = b$ and $\bar{x} = \frac{1}{2}(\bar{x} + t\xi) + \frac{1}{2}(\bar{x} - t\xi)$. Thus, $\bar{x} \in \mathcal{S}$, and it follows from Lemma 1 that the MCP (38) is feasible.

Now assume that $\forall \xi \in \ker(A), \pi^T \xi = 0$. On the one hand, if $\pi_0 < \pi^T \bar{x} < \pi_0 + 1$, then $\mathcal{S} = \emptyset$ and thus $\mathcal{X}$ is empty, which would contradict the $\mathcal{X} \neq \emptyset$ assumption. Thus, either $\pi^T \bar{x} \leq \pi_0$ or $\pi^T \bar{x} \geq \pi_0 + 1$, i.e., $\bar{x} \in \mathcal{S}$ and the MCP is feasible by Lemma 1. Note that this latter case would never occur in practice, since one would always consider a split such that $\pi_0 < \pi^T \bar{x} < \pi_0 + 1$.

The result then follows from Lemma 2. □

Similar to the standard normalization, a consequence of Theorem 10 is that, when considering split disjunctions, the CGCP (37) and the MCP (38) are solvable with identical objective value. This is confirmed by the results of Table 2, which reports statistics for the CGCP in Example 4: no numerical issue is encountered. The corresponding split cuts are displayed in Figure 6, and are similar to the cuts obtained with the standard normalization. Likewise, cuts obtained with the uniform normalization are not, in general, supporting hyperplanes of the disjunctive hull.



(a) (b) (c)

Fig. 6: Split cuts obtained with the uniform normalization. The continuous relaxation is in gray, the split hull in orange, and the obtained cut is in red.

Table 5: CGCP statistics for Example 4 and uniform normalization

	Iter	$\|\alpha\|$	$\|u_1\|$	$\|u_2\|$	$\|\lambda_1\|$	$\|\lambda_2\|$	$\|v_1\|$	$\|v_2\|$
(a)	8	0.5	0.0	0.3	0.5	0.9	0.2	0.3
(b)	8	0.5	0.0	0.4	0.4	1.0	0.2	0.4
(c)	8	0.4	0.0	0.5	0.3	1.1	0.3	0.5

5 Separating conic-infeasible points

When $(MICP)$ (17) is solved by outer-approximation, the fractional point $\bar{x}$ may not satisfy all conic constraints. In preliminary experiments, wherein lift-and-project cuts were separated by rounds in a callback, a large proportion –often higher than 90%– of the cuts yielded by the CGCP turned out to be $\mathcal{K}^*$ cuts, which is obviously detrimental to performance. To the best of our knowledge, despite the popularity and performance of outer-approximation algorithms, this behavior has not been studied in the literature. Therefore, in this section, we will assume that $A\bar{x} = b$, but $\bar{x} \notin \mathcal{K}$.

Let $(\alpha, \beta, u, \lambda, v)$ be a solution of the CGCP, and assume that $v_h = 0$ for some $h \in \{1, ..., H\}$. Thus, we have

$$\alpha = A^T u_h + \lambda_h, \tag{39}$$

$$\beta \leq b^T u_h, \tag{40}$$

and $\alpha^T x \geq \beta$ is a trivial inequality. In addition, as noted in Section 3.1, the inequality $\lambda_h^T x \geq 0$ has the same violation, and cuts off the same portion of the continuous relaxation as $\alpha^T x \geq \beta$. This observation, which does not depend on the normalization condition, allows the *a posteriori* detection of $\mathcal{K}^*$ cuts, by checking the value of the v multipliers.

Once a $\mathcal{K}^*$ cut is identified, it can be disaggregated. Assume that $\mathcal{K} = \mathcal{K}_1 \times ... \times \mathcal{K}_N$; correspondingly, for $\lambda \in \mathcal{K}^*$, we write $\lambda = (\lambda_1, ..., \lambda_N)$ where each $\lambda_i \in \mathcal{K}_i^*$. Then, the $\mathcal{K}^*$ cut $\lambda^T x \geq 0$ is disaggregated as

$$\lambda_i^T x_i \geq 0, \quad i = 1, ..., N. \tag{41}$$

This yields more numerous, but sparser, $\mathcal{K}^*$ cuts, and results in tighter polyhedral approximations which, in turn, improves the performance of outer-approximation algorithms [19].

We now derive sufficient conditions that provide an *a priori* indication that a $\mathcal{K}^*$ cut will be generated. Unless stated otherwise, we only consider the CGCP with standard normalization. Define

$$\bar{\eta} = \min_{\eta \geq 0} \{\eta \mid \bar{x} + \eta\rho \in \mathcal{K}\}, \tag{42}$$

$$\bar{\tau}_h = \min_{\tau \geq 0} \{\tau \mid D_h\bar{x} + \tau\sigma_h \succeq_{\mathcal{Q}_h} d_h\}, \qquad \forall h, \tag{43}$$

and let $\xi = \bar{x} + \bar{\eta}\rho$ and $\bar{\lambda} \in \mathcal{K}^*$ such that $\bar{\lambda}^T\xi = 0, \left|\bar{\lambda}\right|_\rho = 1$. Such a $\bar{\lambda}$ always exists because $\xi \in \partial\mathcal{K}$, and can be obtained from, e.g., an optimal solution of the dual of (42). Also note that $\bar{\lambda}^T\bar{x} = -\bar{\eta}$.

Theorem 11 *Define $\tilde{\alpha}$ and $\tilde{\lambda}$ by $\tilde{\alpha} = \tilde{\lambda}_h = H^{-1}\bar{\lambda},\ \forall h$.*
If $\forall h, \bar{\eta} \geq H^{-1}\bar{\tau}_h$, then $(\tilde{\alpha}, 0, 0, \tilde{\lambda}, 0)$ is optimal for CGCP.

Proof Let $\tilde{\alpha}, \tilde{\lambda}$ be defined as above. It immediately follows that $(\tilde{\alpha}, 0, 0, \tilde{\lambda}, 0)$ is feasible for CGCP with objective value $-H^{-1}\bar{\eta}$. Next, we show it is optimal by exhibiting a feasible solution of MCP with the same objective value.

Let $y_h = \frac{1}{H}\bar{x}$, $z_h = \frac{1}{H}$, and $\eta = \frac{\bar{\eta}}{H}$. We have $Ay_h = z_h b, \forall h$, and $y_h + \eta\rho = \frac{1}{H}(\bar{x} + \bar{\eta}\rho) \in \mathcal{K}$. Finally,

$$\begin{aligned} D_h y_h + \eta\sigma_h &= H^{-1}\left(D_h\bar{x} + \bar{\eta}\sigma_h\right) \\ &\succeq_{\mathcal{Q}_h} H^{-1}d_h \\ &= z_h d_h. \end{aligned}$$

Thus, (y, z, η) is feasible for the MCP (33) and its objective value is $H^{-1}\bar{\eta}$, which concludes the proof. □

Theorem 11 shows that, if $\bar{x}$ is "sufficiently" conic-infeasible, as measured by the magnitude of $\bar{\eta}$, then there exists a $\mathcal{K}^*$ cut that is an optimal solution of the CGCP. Note that there is no guarantee that this optimal solution is unique –in general, it is not– nor that all CGCP-optimal solutions are $\mathcal{K}^*$ cuts. Nevertheless, we have observed that, whenever the condition of Theorem 11 was met, the obtained solution was indeed a $\mathcal{K}^*$ cut.

This suggests several strategies to avoid generating $\mathcal{K}^*$ cuts when solving the CGCP. First, one can check the value of $\bar{\eta}$ and, if large enough as per Theorem 11, avoid solving the CGCP and add an optimal $\mathcal{K}^*$ cut directly. Nevertheless, our initial experiments suggest that only a small number of cases are captured by Theorem 11. Second, one can increase the magnitude of ρ, thus reducing the value of $\bar{\eta}$, to the point where the assumptions of Theorem 11 no longer hold. Note that, as the magnitude of ρ becomes arbitrarily large, the standard normalization becomes equivalent to the uniform normalization of Section 4.5. Third, instead of imposing a normalization condition, one could

fix the cut violation to a positive value, e.g., 1, and optimize a different objective; the feasibility of the CGCP is then guaranteed by the fact that $\bar{x}$ is conic-infeasible. This approach directly relates to the *reverse-polar* CGLP introduced in [47]. Finally, one can simply try to avoid conic-infeasible points in the first place, for instance by refining the outer-approximation before cuts are separated.

6 Cut lifting and strengthening

In this section, we present conic extensions of the classical lifting and strengthening procedures in MILP. For simplicity, we will assume that $\mathcal{K} = \mathcal{K}_1 \times \mathcal{K}_2$ where $\mathcal{K}_i \subseteq \mathbb{R}^{n_i}, i = 1, 2$. Correspondingly, we write

$$A = \begin{bmatrix} A_1 & A_2 \end{bmatrix}, \; D_h = \begin{bmatrix} D_{1,h} & D_{2,h} \end{bmatrix}, \; \lambda_h = \begin{bmatrix} \lambda_{1,h} \\ \lambda_{2,h} \end{bmatrix}, \; \alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}, \; \bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix},$$

and, for ease of reading, we use the convention that $\lambda_i = (\lambda_{i,1}, ..., \lambda_{i,H})$ for $i = 1, 2$. Finally, we assume that $\bar{x}_2 = 0$.

6.1 Cut lifting

In MILP, one can formulate the CGLP in a reduced space, by projecting out the null components of $\bar{x}$, then recover a valid cut in the original space by a lifting procedure, see, e.g., [6]. The same technique is applied to mixed-binary conic problems in [18], although it is then limited to linear constraints. In contrast, here, the approach is generalized to arbitrary MICP problems and disjunctions, and to projecting out non-linear conic constraints.

Recall that $\bar{x}_2 = 0$, and consider the reduced CGCP

$$\begin{aligned}
\min \quad & \alpha_1^T \bar{x}_1 - \beta && && (44a)\\
s.t. \quad & \alpha_1 = A_1^T u_h + \lambda_{1,h} + D_{1,h}^T v_h, && \forall h, && (44b)\\
& \beta \leq b^T u_h - v_h^T d_h, && \forall h, && (44c)\\
& (u_h, \lambda_{1,h}, v_h) \in \mathbb{R}^m \times \mathcal{K}_1^* \times \mathcal{Q}_h^*, && \forall h. && (44d)
\end{aligned}$$

All the normalization conditions considered in Section 4 can be adapted to the reduced CGCP, namely by normalizing only the α_1, λ_1, v components as appropriate. For instance, the α normalization would write $\|\alpha_1\|_* \leq 1$, and the uniform normalization $\sum_h |\lambda_{1,h}|_{\rho_1} \leq 1$, for $\rho_1 \in \operatorname{int} \mathcal{K}_1$.

Any solution $(\alpha, \beta, u, \lambda, v)$ that is feasible for the CGCP yields a feasible solution $(\alpha_1, \beta, u, \lambda_1, v)$ for the reduced CGCP. In addition, since $\bar{x}_2 = 0$, the corresponding objective values are the same. Reciprocally, a feasible solution for the CGCP can be obtained from a feasible solution of the reduced CGCP as shown in Lemma 3.

Lemma 3 *Let* $(\alpha_1, \beta, u, \lambda_1, v)$ *be feasible for the reduced CGCP* (44)*. Let*

$$\alpha_2 \succeq_{\mathcal{K}_2^*} A_2^T u_h + D_{h,2}^T v_h, \qquad \forall h,$$

and, $\forall h$*, let* $\lambda_{2,h} = \alpha_2 - A_2^T u_h - D_{2,h}^T v_2$*. Then,* $(\alpha, \beta, u, \lambda, v)$ *is feasible for the CGCP* (19) *and* $\alpha^T \bar{x} - \beta = \alpha_1^T \bar{x}_1 - \beta$*.*

The proof of Lemma 3 is immediate. Note that the objective value is unaffected only because $\bar{x} = 0$: if $\bar{x}_2 \neq 0$, then the lifted inequality may not be violated.

The choice of α_2 in Lemma 3 is not unique, especially if $\mathcal{K}_2^*$ possesses high-dimensional faces. Nevertheless, a reasonable requirement is to impose that α_2 be minimal with respect to $\succeq_{\mathcal{K}_2^*}$, i.e., that there does not exist a valid $\tilde{\alpha}_2 \preceq_{\mathcal{K}_2^*} \alpha_2$. Indeed, if α_2 is not $\succeq_{\mathcal{K}_2^*}$-minimal, then $\alpha^T x \geq \beta$ is trivially dominated by $\tilde{\alpha}^T x \geq \beta$, see also [34].

A $\succeq_{\mathcal{K}_2^*}$-minimal α_2 is obtained by solving the lifting conic problem (LCP)

$$(LCP) \quad \min_{\alpha_2} \quad \rho_2^T \alpha_2 \tag{45a}$$

$$s.t. \quad \alpha_2 \succeq_{\mathcal{K}_2^*} A_2^T u_h + D_{h,2}^T v_h, \qquad \forall h, \tag{45b}$$

where $\rho_2 \in \operatorname{int} \mathcal{K}_2$, and denote by $\lambda_{2,h} \in \mathcal{K}_2^*$ the (conic) slack associated to (45b). First, since $\mathcal{K}_2$ is a proper cone, the LCP is strongly feasible. Second, we have

$$\rho_2^T \alpha_2 = \rho_2^T \lambda_{2,h} + \rho_2^T \left(A_2^T u_h + D_{h,2}^T v_h \right), \qquad \forall h, \tag{46}$$

thereby showing that the objective value of the LCP is bounded below. In addition, let (α_2, λ_2) be a feasible solution for the LCP with objective value Z. It then follows that

$$\forall h, \quad \left| \tilde{\lambda}_{2,h} \right|_{\rho_2} \leq Z - \rho_2^T \left(A_2^T u_h + D_{h,2}^T v_h \right) \tag{47}$$

in any feasible solution $(\tilde{\alpha}_2, \tilde{\lambda}_2)$ with objective value $\tilde{Z} \leq Z$. Equation (47) implicitly bounds the magnitude of λ_2, thereby ensuring that the LCP is solvable. Finally, if α_2 and $\tilde{\alpha}_2$ are feasible and $\tilde{\alpha}_2 \preceq_{\mathcal{K}_2^*} \alpha_2$, then $\rho_2^T \tilde{\alpha}_2 < \rho_2^T \alpha_2$ and α_2 cannot be an optimal solution. Thus, any optimal solution of the LCP is $\succeq_{\mathcal{K}_2^*}$-minimal.

Whenever $\mathcal{K}_2$ is not irreducible, the LCP can be decomposed per conic component, yielding smaller problems. In the linear case, taking $\mathcal{K}_2 = \mathbb{R}_+$, the LCP reduces to

$$\alpha_2 = \max_h \left\{ A_2^T u_h + D_{2,h}^T v_h \right\},$$

which is the classical lifting procedure for disjunctive cuts in MILP [23].

Lemma 3 does not account for the normalization constraint in the CGCP. Although the lifting procedure does not modify the objective value, in general, the lifted solution $(\alpha, \beta, u, \lambda, v)$ is not an optimal solution of the normalized original CGCP. Thus, the reduction in the size of the CGCP, and the associated computational gains, come at the expense of potentially weaker cuts.

6.2 Cut strengthening

Balas and Jeroslow's original derivation of monoidal strengthening for disjunctive cuts [7] exploited the non-negativity and integrality of *individual* variables to strengthen the corresponding cut coefficients. Since, in general, a conic constraint may involve several variables, it is not clear whether and how one can extend this approach to the conic setting. Thus, we restrict our attention to split cuts, and propose a conic extension of monoidal strengthening that builds on the geometric idea of Wolsey's proof of Theorem 2.2 in [6].

For simplicity, we consider the pure integer case. Given a split disjunction $(\pi^T x \le \pi_0) \vee (\pi^T x \ge \pi_0 + 1)$, the CGCP writes

$$\begin{aligned} \min \quad & \alpha^T \bar{x} - \beta \\ s.t. \quad & \alpha_1 = A^T u_1 + \lambda_1 - v_1 \pi \\ & \alpha_2 = A^T u_2 + \lambda_2 + v_2 \pi \\ & \beta \le b^T u_1 - v_1 \pi_0 \\ & \beta \le b^T u_2 + v_2(\pi_0 + 1) \\ & \lambda_1, \lambda_2 \in \mathcal{K}^*, v_1, v_2 \ge 0. \end{aligned}$$

Lemma 4 *Let $(\alpha, \beta, u, \lambda, v)$ be feasible for the CGCP, and let $\tilde{\alpha}_2$ and $\delta_\pi \in \mathbb{Z}^{n_2}$ such that*

$$\begin{aligned} \tilde{\alpha}_2 &\succeq_{\mathcal{K}_2^*} A_2^T u_1 - v_1(\pi_2 + \delta_\pi), \\ \tilde{\alpha}_2 &\succeq_{\mathcal{K}_2^*} A_2^T u_2 + v_2(\pi_2 + \delta_\pi). \end{aligned}$$

Then, $\tilde{\alpha}^T x \ge \beta$ is a valid inequality for the disjunctive set

$$\tilde{\mathcal{D}} = \left\{ x \;\middle|\; \begin{array}{l} Ax = b, x \in \mathcal{K} \\ \tilde{\pi}^T x \le \pi_0 \end{array} \right\} \cup \left\{ x \;\middle|\; \begin{array}{l} Ax = b, x \in \mathcal{K} \\ \tilde{\pi}^T x \ge \pi_0 + 1 \end{array} \right\},$$

where $\tilde{\alpha} = (\alpha_1, \tilde{\alpha}_2)$ and $\tilde{\pi} = (\pi_1, \pi_2 + \delta_\pi)$.

In the mixed-integer case, one simply needs to set to zero, in Lemma 4, the components of δ_π that correspond to continuous variables. Since $\delta_\pi \in \mathbb{Z}^{n_2}$, we have $\tilde{\pi} \in \mathbb{Z}^n$, and $(\tilde{\pi}^T x \le \pi_0) \vee (\tilde{\pi}^T x \ge \pi_0 + 1)$ is a valid disjunction for $\mathcal{X}$. Thus, the strengthened inequality is valid for $\mathcal{X}$. Furthermore, because $\bar{x}_2 = 0$, strengthening does not affect the cut violation: if the original cut is violated, so is the strengthened one. This may not hold if $\bar{x}_2 \ne 0$, i.e., the strengthened inequality might not be violated.

Similar to the lifting case, a reasonable requirement is for $\tilde{\alpha}_2$ to be minimal with respect to $\succeq_{\mathcal{K}_2^*}$. Following the same approach as in Section 6.1, we obtain the *Cut-Strengthening Problem*

$$\begin{aligned} (CSP) \quad \min_{\tilde{\alpha}_2, \delta_\pi} \quad & \rho_2^T \tilde{\alpha}_2 && \text{(48a)} \\ s.t. \quad & \tilde{\alpha}_2 \succeq_{\mathcal{K}_2^*} A_2^T u_1 - v_1(\pi_2 + \delta_\pi), && \text{(48b)} \\ & \tilde{\alpha}_2 \succeq_{\mathcal{K}_2^*} A_2^T u_2 + v_2(\pi_2 + \delta_\pi), && \text{(48c)} \\ & \delta_\pi \in \mathbb{Z}^{n_2}. && \text{(48d)} \end{aligned}$$

Similar to the LCP, the CSP can be decomposed per conic component, so that its resolution remains tractable. Furthermore, in the linear case with $\mathcal{K}_2 = \mathbb{R}_+$, the CSP reduces to

$$\tilde{\alpha}_2 = \max_{\delta \in \mathbb{Z}} \left\{ \min \left(A_2^T u_1 - v_1 \delta, A_2^T u_2 + v_2 \delta \right) \right\},$$

which is the classical monoidal strengthening of Balas and Jeroslow [7].

7 Computational results

In this section, we investigate the practical behavior of the normalization conditions of Section 4 along two lines: the progression of the gap closed, and the characteristics of the obtained cuts. Indeed, the choice of normalization impacts the numerical stability and computational efficiency of solving the CGCP, thereby affecting the rate at which gap is closed, in terms of both time and number of rounds.

Several solvers, e.g., CPLEX, Gurobi and Mosek, support classes of MI-CONIC problems. Nevertheless, to the best of our knowledge, CPLEX is the only one that exploits non-linear information when generating lift-and-project cuts. Thus, we use CPLEX as a baseline, and restrict our comparison to MISOCP instances, which is the only class of non-linear MI-CONIC problems supported by CPLEX.

7.1 Instances

We select an initial testset of 114 MISOCP instances from the CBLIB [24] collection. Each instance is first reformulated in the standard form (17) and, to address a limitation in the CPLEX Julia wrapper, all rotated second-order cone constraints are reformulated as second-order cone constraints.

Each instance is solved using the CPLEX outer-approximation algorithm on a single thread, and all other parameters are left to their default value. Instances with a root gap smaller than 1% are removed from the testset, as well as those for which no integer-feasible solution was found by CPLEX after one hour of computing time. This yields a testset of 106 instances, divided into 7 groups. Table 6 reports, for each group, the number of instances (#Inst) in that group, and the average number of: variables, integer variables, constraints, non-polyhedral cones, and non-zero coefficients.

7.2 Implementation details

Our implementation[3] is coded in Julia. All solvers, namely, CPLEX 20.1 [29], Gurobi 9.1.1 [28] and Mosek 9.2 [43], are accessed through the solver-agnostic

[3] Code available at `https://github.com/mtanneau/CLaP`

Table 6: Instance statistics

Group	#Inst	Variables	Integers	Constraints	Cones	Nz coeff.
`clay`	12	829	35	659	80	1711
`flay`	10	535	28	408	4	1121
`slay`	14	1217	92	936	14	2653
`fmo`	41	619	48	479	15	1841
`sssd`	16	694	153	546	18	1414
`tls`	5	708	147	541	26	2066
`uflquad`	8	37568	22	30203	3750	71341
All	106	3518	69	2809	304	7057

interface `MathOptInterface` [37]. Experiments are carried out on an Intel Xeon Gold 6142@2.60GHz CPU, 512GB RAM machine running Linux. With the exception of Table 6, all reported averages are shifted geometric means with a shift of 1.

Baseline. We use CPLEX internal lift-and-project cut generation at the root node as a baseline.

For each instance, we run CPLEX outer-approximation algorithm on a single thread, no presolve, no heuristics, and all cuts de-activated with the exception of lift-and-project cuts which are set to the most aggressive setting. The `CutsFactor` parameter is set to 10^{30}, thereby removing any limit on the number of cuts that can be added to the formulation, and the maximum number of cutting plane passes is set to 200. Finally, the node limit is set to zero, i.e., only the root node is explored, and we set a time limit of 1 hour.

Each CPLEX "cut pass" consists[4] of one round of cuts, plus additional components such as heuristics and reduced cost fixing. Cut generation stops as soon as no violated cuts are found, or all violated cuts are rejected. Since we deactivate heuristics and presolve, it is fair to assume that, in the present setting, one CPLEX pass corresponds to one round of cuts. Nevertheless, due to CPLEX internal work limits, e.g., numerical tolerances on a cut's coefficients, early termination may occur even though violated cuts were identified. Whether or not this is the case cannot be inferred from the CPLEX log.

Cut separation. Cuts are separated by rounds in a callback, and submitted to CPLEX as user cuts. Before this happens, CPLEX automatically adds a number of $\mathcal{K}^*$ cuts (denoted by "cone linearizations" in the CPLEX log), thereby making the initial dual bound equivalent to that of the continuous conic relaxation. Then, each round proceeds as follows.

First, $\bar{x}$ is cleaned, i.e, we set $\bar{x}_j = 0$ for all j such that $|\bar{x}_j| \leq 10^{-7}$, and $\bar{\eta}$ is computed as per Equations (42). If $\bar{\eta}$ is greater than a conic-feasibility tolerance $\epsilon_{\mathcal{K}} > 0$, the current outer-approximation is refined by adding violated $\mathcal{K}^*$ cuts. This refinement step is cheap, and it is repeated until $\bar{\eta} \leq \epsilon_{\mathcal{K}}$, up to a

[4] Personal communication with CPLEX developers.

maximum of 50 times. Large values for $\epsilon_{\mathcal{K}}$ increase the likelihood of generating $\mathcal{K}^*$ cuts from the CGCP, while small values can lead to an adverse tailing-off effect within the refinement process; following initial tests, we set $\epsilon_{\mathcal{K}} = 0.05$.

Second, for each fractional coordinate of $\bar{x}$, the CGCP is formed and solved to generate a lift-and-project cut. We do not project the CGCP onto the support of $\bar{x}$, i.e., no lifting is performed. For the standard normalization, the sufficient conditions of Theorem 11 are checked first and, if met, $\mathcal{K}^*$ cuts are added and no CGCP is solved. If no CGCP solution is available, or if the objective value of the CGCP is greater than -10^{-4}, no cut is generated and we proceed to the next fractional coordinate.

Third, the values of v_1 and v_2 are checked to identify $\mathcal{K}^*$ cuts, which are disaggregated and added to the formulation. Otherwise, the cut is strengthened following the procedure of Section 6.2. For stability, strengthening is performed only if $|v_1| + |v_2| \geq 10^{-4}$. Then, also for numerical stability, we set α_j to zero for every j such that $|\alpha_j| \leq 10^{-7}$. Similarly, if $|\beta| \leq 10^{-8}$, the cut's right-hand side is set to zero. Finally, the cut is passed to the solver.

Compact CGCP. Lift-and-project cuts are obtained from elementary split disjunctions, i.e., disjunctions of the form

$$\left(-\pi^T x \geq \pi_0\right) \vee \left(\pi^T x \geq \pi_0 + 1\right),$$

where $\pi_0 \in \mathbb{Z}$ and $\pi = e_j$ for given $j \in \{1, ..., p\}$. When formulating the CGCP, we set u_1 to zero, and substitute out α and β. This yields the compact CGCP

$$\begin{aligned}
\min_{u,\lambda,v} \quad & \bar{x}^T\lambda_1 - (\pi^T\bar{x} - \pi_0)v_1 + t_1 \\
s.t. \quad & A^T u_2 + (\lambda_2 - \lambda_1) + (v_1 + v_2)\pi = 0 \\
& b^T u_2 + (t_1 - t_2) + (v_1 + v_2)\pi_0 + v_2 = 0 \\
& \lambda_1, \lambda_2 \in \mathcal{K}^*, \\
& v_1, v_2 \geq 0 \\
& t_1, t_2 \geq 0,
\end{aligned}$$

where t_1, t_2 are non-negative slacks associated with constraints (19c). The relation $\alpha = \lambda_1 - v_1\pi$ allows to formulate the normalization condition in the λ, v space.

In practice, the equality constraints $A\bar{x} = b$ are satisfied only up to numerical tolerances. This can cause the CGCP to be unbounded whenever the u multipliers are not bounded. We have observed that setting u_1 to zero, and writing the objective as in the compact CGCP above, greatly improve the numerical stability of the CGCP, especially when $\bar{x}$ is obtained from an interior-point method.

Other details. Cuts that are generated are never added to the CGCP formulation, i.e., we only separate rank-1 lift-and-project cuts. All CGCPs are solved with Gurobi, which we found to be more robust here. We use the barrier algorithm with a single thread, no presolve, and we disable the computation of dual variables by setting the `QCPDual` parameter to 0.

We implement a simple procedure to identify implied-integer variables. If a variable y appears in a constraint of the form $a^T x \pm y = b$, where $b \in \mathbb{Z}$, all coefficients of a are integers, and x is a vector of integer or implied-integer variables, then y is an implied-integer variable. This step is repeated until no additional implied-integer variable is detected. Then, when generating cuts, the coefficients of implied-integer variables can be strengthened using the technique of Section 6.2. Unless specified otherwise, we always report results for the strengthened cuts.

Since we use an outer-approximation algorithm, we do not include the trivial normalization in our experiments. The α normalization is formulated using the ℓ_2 norm. For the polar normalization, we take $\gamma = x^* - \bar{x}$, where x^* is the best integer solution obtained by CPLEX after one hour of computing time. As per Section 4.2, this ensures that the MCP (30) is at least (weakly) feasible. While our focus is on the behavior of the cut-generation process, note that the present setting is artificial, since x^* –or any integer-feasible solution, may not be available at the root. How to choose γ in practice remains an open research avenue. Finally, for the standard and uniform normalization, and the computation of $\bar{\eta}$ in Equation (42), we set $\rho = (\rho_1, ..., \rho_N)$, where

$$\rho_i = \begin{cases} 1 & \text{if } \mathcal{K}_i = \mathbb{R}_+ \\ -1 & \text{if } \mathcal{K}_i = \mathbb{R}_- \\ (1,0,0) & \text{if } \mathcal{K}_i = \mathcal{L}_3 \end{cases},$$

and, since we only consider split cuts, we take $\sigma_1 = \sigma_2 = 1$.

7.3 Gap closed and computing time

Recall that, in our approach, each round may include the separation of $\mathcal{K}^*$ cuts to refine the current outer approximation. To the best of our knowledge, CPLEX may also refine the outer approximation in a similar way, but we do not know when this is triggered and how often it occurs in practice. In addition, total computing times include the time of building the CGCP, which typically represent 30 to 40% of total time, and up to 90% for `uflquad` instances. These large building times are in part due to limitations of the Gurobi Julia wrapper, and could be significantly reduced by using a lower-level interface. Thus, direct comparisons of computing times between the CPLEX baseline and our CGCP-based approaches should be cautious, as there is significant room for improvement in the latter. Other reported metrics, such as gap closed and cut sparsity, are not subject to performance variability unless a time limit is reached, which only happened for some of the larger `uflquad` instances.

Table 7: Completed cut rounds and CPU time

	CPLEX			Alpha			Polar			Standard			Uniform		
Group	#rd	CPU	T/rd	#rd	CPU	T/rd	#rd	CPU	T/rd	#rd	CPU	T/rd	#rd	CPU	T/rd
`clay`	133	22.8	0.2	200	284.5	1.6	76	94.6	1.4	200	232.9	1.3	192	207.3	1.3
`flay`	16	0.8	0.1	155	68.4	0.8	62	20.1	0.5	138	32.8	0.4	144	32.9	0.4
`slay`	19	3.9	0.3	145	727.9	6.1	89	86.1	1.5	175	277.5	2.2	109	141.3	1.8
`fmo`	91	10.0	0.1	200	342.2	1.9	188	204.3	1.3	130	95.4	0.8	123	77.3	0.7
`sssd`	29	2.5	0.1	182	412.7	2.4	21	24.5	1.3	22	28.0	1.4	20	23.4	1.3
`tls`	127	29.4	0.4	140	379.9	3.8	136	178.8	1.8	141	310.8	2.8	149	269.5	2.6
`uflquad`	126	491.6	5.1	1	1209.6	1209.6	1	173.0	173.0	27	3192.1	127.4	27	3171.4	126.6
All	58	9.7	0.3	128	362.0	4.1	70	96.2	2.1	98	126.5	1.9	89	101.9	1.8

Table 7 reports, for each group of instances and normalization/baseline, the average number of completed cut rounds before termination (#rd), the average total computing time in seconds (CPU) –which includes time spent within CPLEX, and the average time per round (T/rd). For instance, across the `clay` group of instances, CPLEX performs an average of 133 rounds of cuts in 22.8s, yielding an average time of 0.2s per round. Overall, with the exception of `fmo` instances, the polar normalization performs markedly fewer rounds than the other normalizations. This is due to numerical issues in the CGCP that lead to premature termination of the cut-generation. Furthermore, the time per cut round is very similar for the standard and uniform normalizations, followed closely by the polar normalization, and all three are about 2x faster than the alpha normalization. Indeed, the alpha normalization (27e) with ℓ_2 norm is nonlinear and, as highlighted in Section 4.1, the CGCP (27) may not be solvable, thus causing slow convergence and more interior-point iterations. Finally, CPLEX performs fewer rounds of cuts than our CGCP-based approaches, most likely because of some internal stopping criterion.

Next, Tables 8 and 9 report the percent gap closed after up to 10 and 200 rounds of cuts, respectively. Each table includes results with monoidal strengthening (S) and without (NS); to the best of our knowledge, CPLEX systematically applies strengthening. The percent gap closed is measured as

$$Gap = 100 \times \frac{Z^* - Z_{CP}}{Z_{MICP} - Z_{CP}},$$

where Z_{CP}, Z_{MICP}, Z^* are the optimal value of the continuous conic relaxation, the objective value of the best known integer solution, and the current lower bound after adding cuts, respectively. The conic relaxations are solved with Mosek, after removing all integer constraints.

First, while the use of strengthening almost always improves the gap closed, its effect vary across normalization and instances. For standard and uniform normalizations, strengthening is more useful in the first 10 rounds, and its effect vanishes as more rounds are performed. Indeed, gap closed with and without strengthening are virtually identical after 200 rounds, except for `tls` and `uflquad` instances where strengthening roughly doubles the gap closed. For alpha and polar normalizations, we observe substantial improvements in gap closed for `sssd` and `tls` instances. In particular, the average gap closed on `sssd` instances with the alpha (resp. polar) normalization jumps from 0.1%

Table 8: Gap closed with and without strengthening - 10 rounds

	CPLEX		Alpha		Polar		Standard		Uniform	
Group	NS	S	NS	S	NS	S	NS	S	NS	S
clay	-	2.1	0.0	0.0	0.0	0.0	9.0	9.7	3.3	3.3
flay	-	0.3	0.9	0.5	0.0	0.0	1.2	1.2	3.0	2.2
slay	-	3.6	0.8	0.4	0.0	0.0	9.1	9.3	10.6	11.3
fmo	-	6.9	0.0	0.0	0.0	0.0	23.8	23.7	24.2	24.3
sssd	-	46.2	0.1	42.5	0.3	88.1	94.6	95.6	97.7	97.6
tls	-	8.2	1.8	3.1	1.6	2.6	4.7	5.3	5.1	6.2
uflquad	-	0.6	0.0	0.0	0.0	0.0	10.0	20.2	15.7	21.8
All	-	5.5	0.2	1.0	0.1	1.1	16.0	17.2	16.5	16.8

Table 9: Gap closed with and without strengthening - 200 rounds

	CPLEX		Alpha		Polar		Standard		Uniform	
Group	NS	S	NS	S	NS	S	NS	S	NS	S
clay	-	21.6	0.0	0.0	0.0	1.0	21.6	27.2	27.3	27.3
flay	-	7.3	2.6	2.6	0.4	0.4	6.5	9.6	39.7	39.7
slay	-	14.5	13.7	16.5	0.6	1.4	79.4	79.7	84.3	84.1
fmo	-	24.4	0.0	0.0	0.2	0.3	24.5	24.5	24.5	24.5
sssd	-	95.1	11.6	96.2	7.2	97.6	99.4	99.4	99.4	99.4
tls	-	20.0	2.9	4.2	3.4	12.3	8.6	15.4	9.0	15.8
uflquad	-	5.0	0.0	0.0	0.0	0.0	14.8	28.4	22.9	31.0
All	-	22.0	1.5	2.6	0.7	2.1	28.5	32.7	36.1	37.8

(resp. 0.3%) to 42.5% (resp. 88.1%) after 10 rounds, and from 11.6% (resp. 7.2%) to 96.2% (resp. 97.6%) after 200 rounds. Overall, given that performing 200 rounds is unlikely in practice, these results suggest that, by default, strengthening should always be applied.

Second, the standard and uniform normalizations display similar performance. This is with the exception of `flay` instances, for which the uniform normalization closes significantly more gap than all other approaches. Although the standard normalization closes slightly more gap after 10 rounds, namely, 17.2% against 16.8%, the uniform normalization performs best overall after 200 rounds, with an average 37.8% gap closed across all instances. Furthermore, no numerical issues were recorded for either normalization, thereby demonstrating the benefits of ensuring strong feasibility of both CGCP and MCP.

Third and last, the standard and uniform normalizations are competitive with CPLEX in terms of gap closed. Indeed, both normalizations close significantly more gap than CPLEX after 10 and 200 rounds, notably on `slay` and `uflquad` instances, as well as `flay` instances for the uniform normalization. Only for `tls` instances are they outperformed by CPLEX. In particular, we note that both CGCP-based approaches close over three times more gap than CPLEX in the first 10 rounds. This behavior is encouraging, since closing more gap early is a desirable feature in practice.

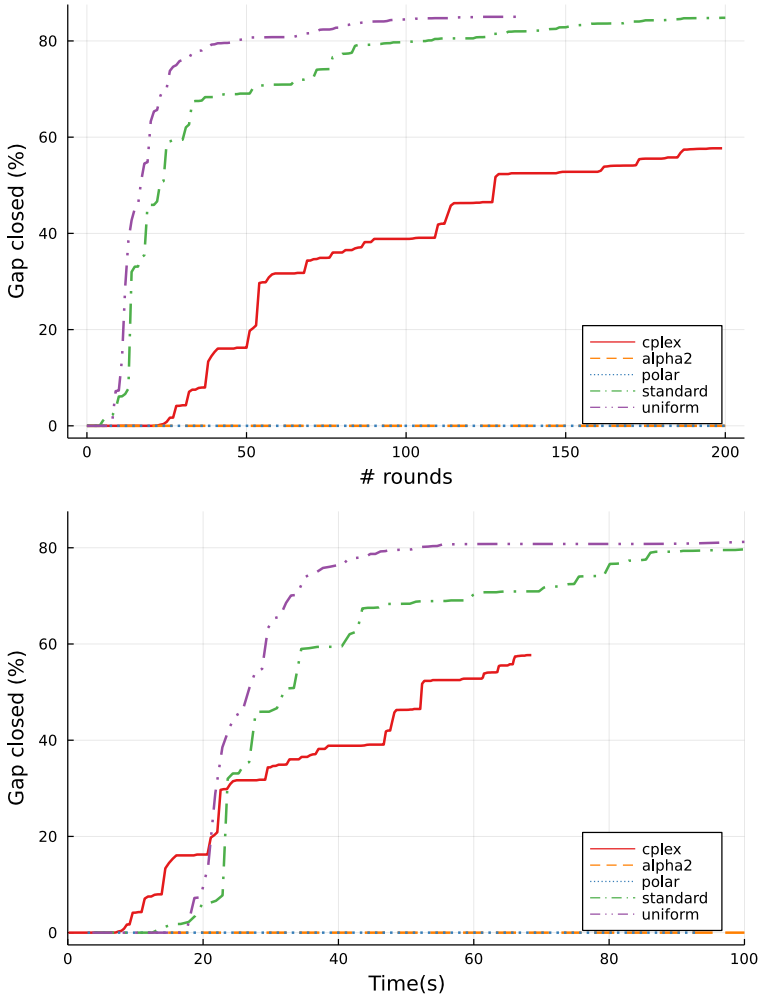


Fig. 7: Instance `slay07h`: gap closed per round (top) and time (bottom)

Finally, Figures 7, 8 and 9 illustrate the progression of the gap closed for instances `slay07h`, `sssd-weak-15-4` and `tls4`, respectively. Each figure displays, for each normalization and the baseline, the percent gap closed as a function of the number of rounds, and of computing time. Again, recall that a direct comparison of computing times between our approach and CPLEX is not meaningful, thus, we focus on general trends.

Overall, Figures 7, 8 and 9 corroborate the previous observations from Tables 7-9: the α normalization is the slowest, while the standard and uniform normalizations display similar progressions and allow to close more gap than CPLEX in the first few rounds. We also observe in Figure 8 that, when the

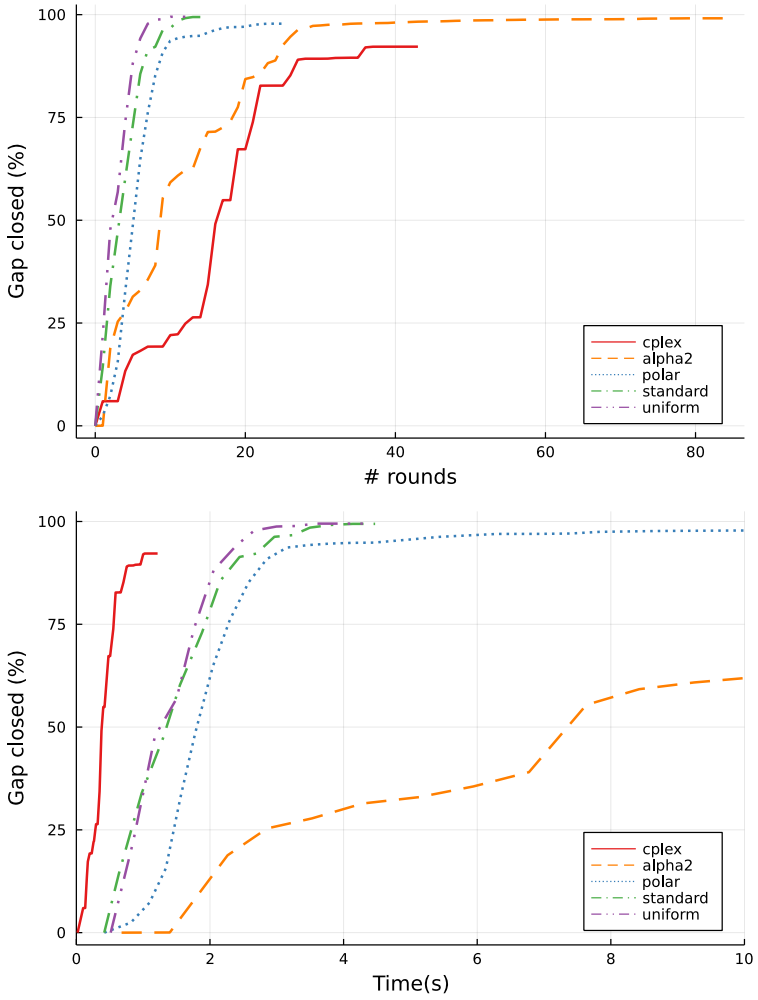


Fig. 8: Instance `sssd-weak-15-4`: gap closed per round (top) and time (bottom). The polar normalization was terminated after two rounds.

polar normalization does not run into numerical issues, it can match the performance of standard and uniform normalizations in terms of gap closed in the first few rounds.

7.4 Cut sparsity

We now study, for each normalization, the characteristics of the cuts obtained from solving the CGCP. Relevant statistics are reported after 10 and 200 rounds, in Table 10 and Table 11, respectively. For each normalization, we

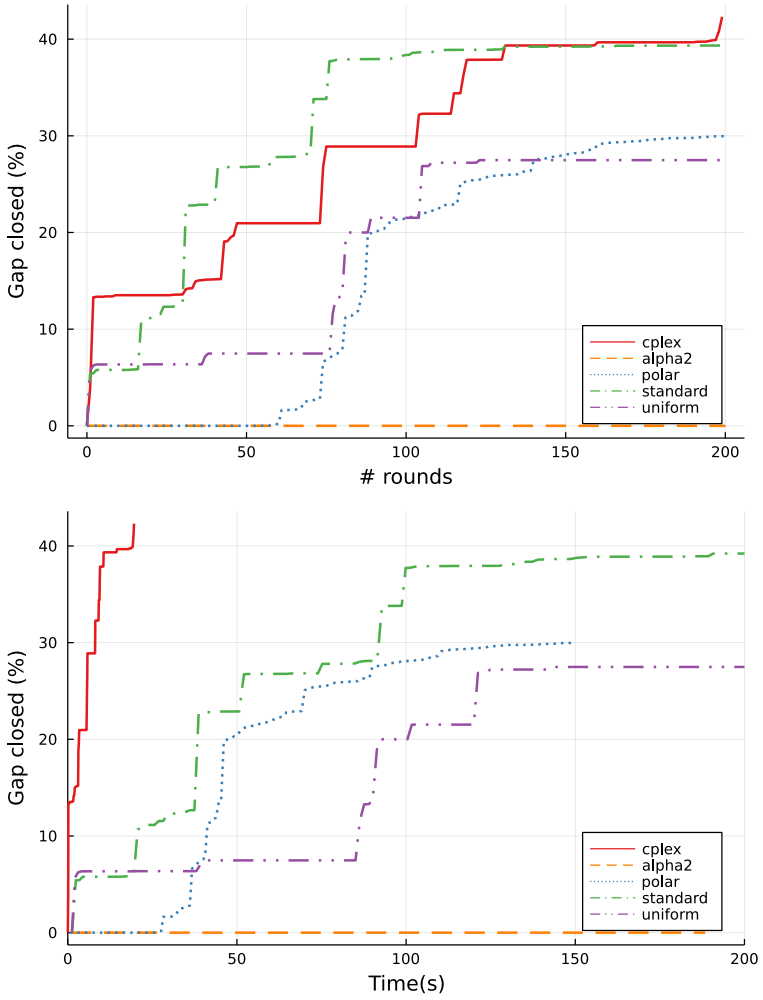


Fig. 9: Instance `tls4`: gap closed per round (top) and time (bottom)

report the geometric mean of the total number of disaggregated $\mathcal{K}^*$ cuts ($\mathcal{K}^*$) and of lift-and-project cuts (L&P). We also report the geometric mean of the density of lift-and-project cuts (%nz), measured as the percentage of non-zero coefficients per cut; the density of $\mathcal{K}^*$ cuts is not reported because they are always disaggregated. Here, $\mathcal{K}^*$ cuts refer specifically to cuts obtained from the CGCP that were identified as $\mathcal{K}^*$ cuts, i.e., we do not consider those added in the refinement steps. For example, the first row of Table 10 indicates that, for `clay` instances and after 10 rounds, the CGCP with standard normalization yielded an average of 349 disaggregated $\mathcal{K}^*$ cuts, and 88 lift-and-project cuts with an average density of 3.7%.

Table 10: Cut statistics - 10 rounds

	Alpha			Polar			Standard			Uniform		
Group	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz
`clay`	0	134	45.6	0	60	19.0	349	88	3.7	40	104	1.5
`flay`	0	119	27.6	0	62	29.1	12	112	3.5	12	111	2.3
`slay`	0	255	27.2	0	182	41.0	20	186	2.6	9	210	1.7
`fmo`	24	247	35.5	74	228	35.7	88	230	1.6	65	231	1.2
`sssd`	0	275	16.4	0	243	29.1	169	119	6.3	150	135	6.4
`tls`	871	301	13.2	2755	284	20.4	199	162	6.5	155	189	6.7
`uflquad`	0	21	33.9	0	21	55.6	226	57	0.5	158	65	0.4
All	4	184	29.2	7	144	32.5	87	150	2.6	52	162	2.0

Table 11: Cut statistics - 200 rounds

	Alpha			Polar			Standard			Uniform		
Group	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz	$\mathcal{K}^*$	L&P	%nz
`clay`	0	2560	45.6	0	645	17.2	7809	1569	5.4	3256	2006	2.5
`flay`	0	1836	24.7	0	529	28.2	411	1512	3.8	286	1894	3.2
`slay`	0	5477	21.3	0	1742	40.5	2987	2485	5.6	645	2109	4.8
`fmo`	1034	5124	33.7	3118	4365	37.1	444	2593	3.8	312	2292	3.0
`sssd`	0	5558	10.3	0	524	30.6	298	321	11.4	272	313	11.2
`tls`	15413	3966	10.0	33458	4451	16.5	3131	2006	7.2	2957	2385	7.2
`uflquad`	0	21	33.9	0	21	55.6	576	232	0.3	391	273	0.3
All	22	2872	25.3	36	1242	32.4	826	1391	4.5	493	1386	3.6

First, cuts obtained with the alpha and polar normalization are significantly denser than those obtained with the standard and uniform normalizations. Here, the former two yield cuts with an average of 25% to 32% non-zero coefficients, roughly an order of magnitude higher than the latter. The difference is most striking for `uflquad` instances, where cuts obtained with alpha and polar normalization are over 100x denser than those obtained with standard and uniform normalizations. Furthermore, cuts obtained with the uniform normalization are slightly sparser than with standard normalization, with 2.0% and 3.6% non-zero coefficients after 10 and 200 rounds for the former, compared to 2.6% and 4.5% for the latter. Interestingly, the average cut density after 200 rounds is almost twice as large as after 10 rounds, indicating that cuts become denser in the later rounds. Recall that denser cuts are less desirable, as they have an adverse effect on performance: they slow down the resolution of the current continuous relaxation, and are more prone to numerical errors.

Second, the CGCP with alpha and polar normalization almost never yields $\mathcal{K}^*$ cuts, except for `fmo` and `tls` instances. Interestingly, `tls` instances appear more prone to $\mathcal{K}^*$ cuts across all normalizations, with the effect most striking for the polar normalization. Besides, fewer $\mathcal{K}^*$ cuts are obtained with the uniform normalization than with the standard normalization. This behavior

Table 12: Gap closed by refinement $\mathcal{K}^*$ cuts and CGCP cuts (200 rounds)

	Alpha		Polar		Standard		Uniform	
Group	Ref.	CGCP	Ref.	CGCP	Ref.	CGCP	Ref.	CGCP
`clay`	0.0	0.0	0.0	1.0	0.2	27.1	0.0	27.3
`flay`	0.3	2.6	0.0	0.4	0.6	9.3	1.9	37.6
`slay`	8.2	10.5	0.4	1.3	20.0	56.6	27.1	53.2
`fmo`	0.0	0.0	0.2	0.2	2.0	22.7	2.3	22.6
`sssd`	11.1	76.8	9.6	73.7	0.0	99.4	0.2	98.9
`tls`	1.3	2.0	2.8	8.1	2.7	10.7	2.7	11.1
`uflquad`	0.0	0.0	0.0	0.0	0.7	27.4	2.3	26.7
All	1.1	2.2	0.7	1.8	1.7	29.6	2.3	33.3

is consistent with the theoretical analysis of each normalization, as well as the remarks in Section 5. Indeed, recall that cuts obtained with alpha and polar normalization are supporting hyperplanes of the disjunctive hull, while, in general, cuts obtained with standard and uniform normalization are not.

We conclude the analysis by reporting, in Table 12, the proportion of gap closed when adding $\mathcal{K}^*$ cuts obtained during the refinement steps (Ref.), and by CGCP-generated cuts (CGCP). This is done by tracking the progress of the lower bound after each refinement step and after each round of lift-and-project cuts. Note that the latter include both strengthened "true" lift-and-project cuts, and any disaggregated $\mathcal{K}^*$ obtained when solving the CGCP. Also recall that the gap closed is measured against the continuous conic relaxation. Thus, neither refinement-based nor CGCP-based $\mathcal{K}^*$ would close any gap on their own, since all are redundant with the original conic constraints. The present results only indicate that, at some point in the solving process when the current fractional solution would violate the conic constraints, tightening the outer-approximation may result in a dual bound increase. Unsurprisingly, CGCP-based cuts always close more gap than refinement-based $\mathcal{K}^*$ cuts, although, for `slay` instances, refinement-based $\mathcal{K}^*$ cuts do close a relatively large proportion of the gap. This underlines the importance of carefully managing the quality of the outer-approximation throughout the cut-generation process.

8 Conclusion

Motivated by the impact of the disjunctive framework in MILP and the recent success of conic formulations for MI-CONV, we have investigated the computational aspects of disjunctive cuts in MI-CONIC. Building on conic duality, we have extended Balas' cut-generating linear program into a cut-generating conic program, and studied the fundamental role of the normalization condition in its resolution. In doing so, we have answered several relevant questions, especially from the numerical standpoint, left open by previous developments in the area, and have raised new ones.

8.1 What we have learned...

From a theoretical standpoint, we have shown that the normalization condition in the CGCP impacts not only the theoretical properties of the obtained cuts, but also whether the CGCP is solvable in the first place. The latter has direct consequences for the numerical robustness of the CGCP resolution. In particular, we have introduced conic normalizations that guarantee conic strong duality, without any assumptions on the well-posedness of the considered disjunctions. Furthermore, we have identified the risk of generating $\mathcal{K}^*$ cuts when separating conic-infeasible points, and suggested several strategies to alleviate it. Finally, we have proposed extensions of lifting and cut strengthening to the conic setting, which provide further computational benefits.

From a computational standpoint, we have investigated the practical behavior of several normalization conditions, on a diverse set of instances. These experiments indicate that our CGCP-based cut separation is competitive with a state-of-the-art MI-CONIC solver, being able to close more gap in the early stages. In particular, the numerical robustness of the standard and uniform normalization translates in faster separation and more gap closed. Furthermore, carefully managing the outer approximation, so as to avoid large violations of conic constraints, appears critical to the cut generation performance. Finally, while the impact of cut strengthening varies between families of instances and the normalization being used, it is always beneficial, especially in the early rounds.

8.2 ... and what lies ahead

Several theoretical questions are left open. First, while we only consider linear cutting planes, whether conic cuts can be separated efficiently in the general case, and how to best integrate them within existing MI-CONIC algorithms, remains an open question. Second, dominance relations between valid inequalities were introduced in, e.g, [34,32]. Characterizing optimal solutions of the normalized CGCP in that perspective may offer further insight on the importance of normalization. Third, a unifying framework that generalizes monoidal strengthening to the MI-CONIC setting could further improve the practical effectiveness of disjunctive cuts. Recent developments in duality theory and cut-generating functions for MI-CONIC could offer the tools for doing so.

Computational experience with cuts in MI-CONIC remains scarce, and further research in that direction is needed. Decades of experience in MILP indicate that classes of disjunctive cuts with fast separation rules, e.g., Gomory Mixed-Integer cuts and Mixed-Integer Rounding cuts, yield the greatest computational benefits. Similar strategies for MI-CONIC would undoubtedly be beneficial. In a similar fashion, Fischetti et al. [23] have shown that scaling impacts the cuts yielded by the CGLP with standard normalization: there is no indication that MI-CONIC should be any different. Finally, the CGCP form allows to experiment with a number of algorithmic techniques for separating

disjunctive cuts, whose practical effectiveness will most likely be problem-specific.

Acknowledgements The second author was supported by an FRQNT excellence doctoral scholarship, and a Mitacs Globalink research award. We thank Pierre Bonami, Andrea Tramontani and Sven Wiese for several helpful discussions on the topic, as well as the anonymous referees for their comments and suggestions that helped improve the paper.

References

1. Andersen, K., Jensen, A.N.: Intersection cuts for mixed integer conic quadratic sets. In: M. Goemans, J. Correa (eds.) Integer Programming and Combinatorial Optimization, pp. 37–48. Springer Berlin Heidelberg, Berlin, Heidelberg (2013). DOI 10.1007/978-3-642-36694-9_4
2. Atamtürk, A., Narayanan, V.: Conic mixed-integer rounding cuts. Mathematical Programming **122**(1), 1–20 (2010). DOI 10.1007/s10107-008-0239-4
3. Atamtürk, A., Narayanan, V.: Lifting for conic mixed-integer programming. Mathematical Programming **126**(2), 351–363 (2011). DOI 10.1007/s10107-009-0282-9
4. Balas, E.: Disjunctive programming. In: P. Hammer, E. Johnson, B. Korte (eds.) Discrete Optimization II, *Annals of Discrete Mathematics*, vol. 5, pp. 3 – 51. Elsevier (1979). DOI 10.1016/S0167-5060(08)70342-X
5. Balas, E., Ceria, S., Cornuéjols, G.: A lift-and-project cutting plane algorithm for mixed 0–1 programs. Mathematical Programming **58**(1), 295–324 (1993). DOI 10.1007/BF01581273
6. Balas, E., Ceria, S., Cornujols, G.: Mixed 0-1 Programming by Lift-and-Project in a Branch-and-Cut Framework. Management Science (1996). DOI 10.1287/mnsc.42.9.1229
7. Balas, E., Jeroslow, R.G.: Strengthening cuts for mixed integer programs. European Journal of Operational Research **4**(4), 224–234 (1980). DOI 10.1016/0377-2217(80)90106-X
8. Balas, E., Perregaard, M.: Lift-and-project for mixed 0-1 programming: recent progress. Discrete Applied Mathematics **123**(1), 129 – 154 (2002). DOI 10.1016/S0166-218X(01)00340-7
9. Balas, E., Perregaard, M.: A precise correspondence between lift-and-project cuts, simple disjunctive cuts, and mixed integer Gomory cuts for 0-1 programming. Mathematical Programming **94**(2), 221–245 (2003). DOI 10.1007/s10107-002-0317-y
10. Belotti, P., Góez, J.C., Pólik, I., Ralphs, T.K., Terlaky, T.: A conic representation of the convex hull of disjunctive sets and conic cuts for integer second order cone optimization. In: Numerical Analysis and Optimization, pp. 1–35. Springer (2015)
11. Ben-Tal, A., Nemirovski, A.: Lectures on Modern Convex Optimization. Society for Industrial and Applied Mathematics (2001). DOI 10.1137/1.9780898718829
12. Bonami, P.: Lift-and-project cuts for mixed integer convex programs. In: O. Günlük, G.J. Woeginger (eds.) Integer Programming and Combinatoral Optimization, pp. 52–64. Springer Berlin Heidelberg, Berlin, Heidelberg (2011). URL `https://link.springer.com/chapter/10.1007/978-3-642-20807-2\_5`
13. Bonami, P.: On optimizing over lift-and-project closures. Mathematical Programming Computation **4**(2), 151–179 (2012). DOI 10.1007/s12532-012-0037-0
14. Bonami, P., Biegler, L.T., Conn, A.R., Cornuéjols, G., Grossmann, I.E., Laird, C.D., Lee, J., Lodi, A., Margot, F., Sawaya, N., et al.: An algorithmic framework for convex mixed integer nonlinear programs. Discrete Optimization **5**(2), 186–204 (2008)
15. Bonami, P., Linderoth, J., Lodi, A.: Disjunctive cuts for mixed integer nonlinear programming problems. Prog Combin Optim **18**, 521–541 (2011)
16. Cadoux, F., Lemarchal, C.: Reflections on generating (disjunctive) cuts. EURO Journal on Computational Optimization **1**(1), 51–69 (2013). DOI 10.1007/s13675-012-0006-4
17. Ceria, S., Soares, J.: Convex programming for disjunctive convex optimization. Mathematical Programming **86**(3), 595–614 (1999). DOI 10.1007/s101070050106

18. Çezik, M., Iyengar, G.: Cuts for mixed 0-1 conic programming. Mathematical Programming **104**(1), 179–202 (2005). DOI 10.1007/s10107-005-0578-3
19. Coey, C., Lubin, M., Vielma, J.P.: Outer approximation with conic certificates for mixed-integer convex problems. Mathematical Programming Computation **12**, 249–293 (2020)
20. Conforti, M., Wolsey, L.A.: Facet separation with one linear program. Mathematical Programming **178**(1), 361–380 (2019). DOI 10.1007/s10107-018-1299-8
21. Dadush, D., Dey, S., Vielma, J.: The split closure of a strictly convex body. Operations Research Letters **39**(2), 121 – 126 (2011). DOI https://doi.org/10.1016/j.orl.2011.02.002
22. Duran, M.A., Grossmann, I.E.: An outer-approximation algorithm for a class of mixed-integer nonlinear programs. Mathematical Programming **36**(3), 307–339 (1986). DOI 10.1007/BF02592064
23. Fischetti, M., Lodi, A., Tramontani, A.: On the separation of disjunctive cuts. Mathematical Programming **128**(1), 205–230 (2011). DOI 10.1007/s10107-009-0300-y
24. Friberg, H.A.: Cblib 2014: a benchmark library for conic mixed-integer and continuous optimization. Mathematical Programming Computation **8**(2), 191–214 (2016). DOI 10.1007/s12532-015-0092-4
25. Friberg, H.A.: Facial reduction heuristics and the motivational example of mixed-integer conic optimization. Tech. rep. (2016). URL `http://www.optimization-online.org/DB\_FILE/2016/02/5324.pdf`
26. Geoffrion, A.M.: Elements of large-scale mathematical programming. Management Science **16**(11), 676–691 (1970)
27. Grant, M., Boyd, S., Ye, Y.: Disciplined convex programming. In: L. Liberti, N. Maculan (eds.) Global Optimization, *Nonconvex Optimization and Its Applications*, vol. 84, pp. 155–210. Springer US (2006)
28. Gurobi Optimization, L.: Gurobi optimizer reference manual (2018). URL `https://www.gurobi.com`
29. IBM: IBM ILOG CPLEX Optimization Studio. URL `https://www.ibm.com/products/ilog-cplex-optimization-studio`
30. Kazachkov, A.: Non-Recursive Cut Generation. Ph.D. thesis, Carnegie Mellon University (2018). DOI 10.1184/R1/6720881.v1
31. Kılınç, M.R., Linderoth, J., Luedtke, J.: Lift-and-project cuts for convex mixed integer nonlinear programs. Mathematical Programming Computation **9**(4), 499–526 (2017). DOI 10.1007/s12532-017-0118-1
32. Kılınç-Karzan, F., Steffy, D.E.: On sublinear inequalities for mixed integer conic programs. Mathematical Programming **159**(1), 585–605 (2016). DOI 10.1007/s10107-015-0968-0
33. Kılınç-Karzan, F., Yıldız, S.: Two-term disjunctions on the second-order cone. Mathematical Programming **154**(1-2), 463–491 (2015). DOI https://doi.org/10.1007/s10107-015-0903-4
34. Kilin-Karzan, F.: On Minimal Valid Inequalities for Mixed Integer Conic Programs. Mathematics of Operations Research **41**(2), 477–510 (2015). DOI 10.1287/moor.2015.0737
35. Kronqvist, J., Bernal, D.E., Lundell, A., Grossmann, I.E.: A review and comparison of solvers for convex MINLP. Optimization and Engineering **20**(2), 397–455 (2019). DOI 10.1007/s11081-018-9411-8
36. Kronqvist, J., Misener, R.: A disjunctive cut strengthening technique for convex minlp. Optimization and Engineering pp. 1–31 (2020)
37. Legat, B., Dowson, O., Garcia, J.D., Lubin, M.: MathOptInterface: a data structure for mathematical optimization problems (2020)
38. Lubin, M., Yamangil, E., Bent, R., Vielma, J.P.: Extended Formulations in Mixed-Integer Convex Programming. In: Q. Louveaux, M. Skutella (eds.) Proceedings of the 18th Conference on Integer Programming and Combinatorial Optimization (IPCO 2016), *Lecture Notes in Computer Science*, vol. 9682, pp. 102–113 (2016)
39. Lubin, M., Yamangil, E., Bent, R., Vielma, J.P.: Polyhedral approximation in mixed-integer convex optimization. Mathematical Programming **172**(1), 139–168 (2018). DOI 10.1007/s10107-017-1191-y
40. Modaresi, S.: Valid inequalities and reformulation techniques for mixed integer nonlinear programming. Ph.D. thesis, University of Pittsburgh (2016)

41. Modaresi, S., Kılınç, M.R., Vielma, J.P.: Split cuts and extended formulations for mixed integer conic quadratic programming. Operations Research Letters **43**(1), 10 – 15 (2015). DOI 10.1016/j.orl.2014.10.006
42. Modaresi, S., Kılınç, M.R., Vielma, J.P.: Intersection cuts for nonlinear integer programming: convexification techniques for structured sets. Mathematical Programming **155**(1), 575–611 (2016). DOI 10.1007/s10107-015-0866-5
43. MOSEK ApS: The MOSEK Optimization Suite. URL `https://www.mosek.com/`
44. MOSEK ApS: Mosek modeling cookbook (2020). URL `https://docs.mosek.com/modeling-cookbook/index.html`
45. Perregaard, M., Balas, E.: Generating cuts from multiple-term disjunctions. In: K. Aardal, B. Gerards (eds.) Integer Programming and Combinatorial Optimization, pp. 348–360. Springer Berlin Heidelberg, Berlin, Heidelberg (2001)
46. Rockafellar, R.T.: Convex analysis, vol. 28. Princeton university press (1970)
47. Serra, T.: Reformulating the disjunctive cut generating linear program. Annals of Operations Research (2020). DOI 10.1007/s10479-020-03709-2
48. Stubbs, R.A., Mehrotra, S.: A branch-and-cut method for 0-1 mixed convex programming. Mathematical Programming **86**(3), 515–532 (1999). DOI 10.1007/s101070050103
49. Trespalacios, F., Grossmann, I.E.: Review of Mixed-Integer Nonlinear and Generalized Disjunctive Programming Methods (2014). URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/cite.201400037`
50. Trespalacios, F., Grossmann, I.E.: Cutting plane algorithm for convex generalized disjunctive programs. INFORMS Journal on Computing **28**(2), 209–222 (2016). DOI 10.1287/ijoc.2015.0669
51. Yildiz, S., Cornujols, G.: Disjunctive cuts for cross-sections of the second-order cone. Operations Research Letters **43**(4), 432–437 (2015). DOI 10.1016/j.orl.2015.06.001
52. Zhu, Y., Kuno, T.: A disjunctive cutting-plane-based branch-and-cut algorithm for 0- 1 mixed-integer convex nonlinear programs. Industrial & engineering chemistry research **45**(1), 187–196 (2006)