An echelon form of weakly infeasible semidefinite programs and bad projections of the psd cone

Gábor Pataki, Aleksandr Touzov *

October 27, 2020

Abstract

A weakly infeasible semidefinite program (SDP) has no feasible solution, but it has nearly feasible solutions that approximate the constraint set to arbitrary precision. These SDPs are ill-posed and numerically often unsolvable. They are also closely related to "bad" linear projections that map the cone of positive semidefinite matrices to a nonclosed set. We describe a simple echelon form of weakly infeasible SDPs with the following properties: it is obtained by elementary row operations and congruence transformations; it makes weak infeasibility evident; and using it we can construct any weakly infeasible SDP or bad linear projection by an elementary combinatorial algorithm.

We also prove that some SDPs in the literature are in our echelon form, for example, the SDP from the sum-of-squares relaxation of minimizing the famous Motzkin polynomial.

Key words: Semidefinite programming, weak infeasibility, facial reduction, ill-posed problems, bad projection of the semidefinite cone, sum-of-squares polynomials, Motzkin polynomial

MSC 2010 subject classification: Primary: 90C22, 49N15, 15A21 Secondary: 47A52

1 Introduction

We consider semidefinite programming (SDP) feasibility problems of the form

$$\begin{array}{rcl} \mathcal{A}X & = & b \\ X & \in & \mathcal{S}^n_+, \end{array} \tag{P}$$

where $\mathcal{A}$ is a linear map from $n \times n$ symmetric matrices to $\mathbb{R}^m$, $b \in \mathbb{R}^m$, and $\mathcal{S}^n_+$ is the set of symmetric positive semidefinite (psd) matrices.

SDPs – either in the feasibility form, or in an optimization form – appear in many fields, including combinatorial optimization, polynomial optimization, engineering, and machine learning. However, SDPs often behave pathologically and in this work we focus on their pathological kind of infeasibility, called *weak infeasibility.*

Precisely, we say that (P) is weakly infeasible, when it has no feasible solution, but there are matrices that satisfy $\mathcal{A}X = b$, and are arbitrarily close to $\mathcal{S}^n_+$.

*Department of Statistics and Operations Research, UNC Chapel Hill

Example 1. *We first look at the minimal and classical example*

$$\begin{array}{rcl} x_{11} & = & 0 \\ x_{12} = x_{21} & = & 1 \\ X & \in & \mathcal{S}^2_+, \end{array} \tag{SE}$$

where the (i,j)*th element of* X *is denoted by* x_{ij}. *If* X *satisfies the equality constraints of* (SE), *then* X *looks like*

$$X = \begin{pmatrix} 0 & 1 \\ 1 & x_{22} \end{pmatrix},$$

so it cannot be positive semidefinite. Hence (SE) *is infeasible. However, such* X *matrices can approach* $\mathcal{S}^2_+$ *arbitrarily closely since we can choose* $x_{22} > 0$ *to be large, then change the upper left corner of* X *to* $1/x_{22}$ *to make it psd.*

So we conclude that (SE) *is weakly infeasible.*

We visualize this example in Figure 1. The solid blue set bordered by a hyperbola is

$$S = \left\{ (x_{11}, x_{22}) \in \mathbb{R}^2_+ : x_{11}x_{22} \geq 1 \right\},$$

the set of diagonals of 2×2 *psd matrices with offdiagonal element equal to* 1. *We approach* S *arbitrarily closely if we fix* $x_{11} = 0$ *and make* x_{22} *large, moving towards infinity on the* x_{22} *axis.*

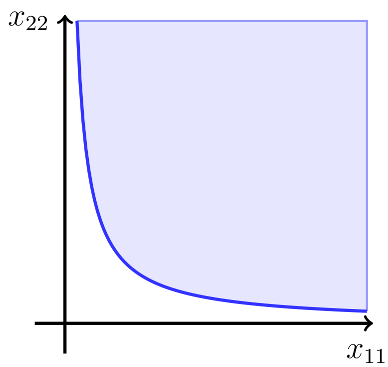


Figure 1: A visualization of (SE)

Weakly infeasible SDPs appear in the literature under many guises, some of which are modern and some classic:

- they are difficult SDPs that are often characterized as feasible ones by solvers.
- If (P) is weakly infeasible, then $\mathcal{A}\mathcal{S}^n_+$ is not closed, since b is in the closure of $\mathcal{A}\mathcal{S}^n_+$, but not in $\mathcal{A}\mathcal{S}^n_+$ itself. Linear maps that carry $\mathcal{S}^n_+$ to a nonclosed set often cause other pathologies in SDPs [23] and are also intriguing from a purely mathematical point of view. These maps were recently christened *bad projections of the psd cone*, and explored from the perspective of algebraic geometry [10].

  More broadly, such maps model linear maps that carry a closed set into a nonclosed one, since $\mathcal{S}^n_+$ is one of the simplest sets for which such maps even exist. The "(non)closedness of the linear image" question appears in several equivalent forms, for example, we may ask when the sum of closed convex cones is closed. Some of the key properties that ensure (non)closedness are Jameson's property (G) [9, 2] and existence of certain tangent directions [21, Theorem 1.1, Theorem 5.1].

- they are *ill-posed*, that is, their distance to the set of feasible instances is zero. Hence their infeasibility cannot be detected by interior point methods whose complexity depends on the distance to feasibility, the best one can do is compute solutions of nearby feasible instances [25, Theorem 13]. For a sample of the thriving literature on complexity analysis of algorithms based on condition numbers, and distance to infeasibility, we refer to [26, 27] and [24].
- Facial reduction algorithms can handle pathological SDPs, however, these algorithms must be implemented in exact arithmetic. Facial reduction algorithms originated in the eighties [4], then simpler variants were proposed in [20, 22]. We will use facial reduction as a theoretical tool to develop our echelon form. In particular, we will use [15, Part 2, Theorem 5] a weaker version of which appeared in [17].
- In contrast to the previous points, robust solutions to SDPs in the Lasserre hierarchy of polynomial optimization can be found by SDP solvers, even when exact solutions are impossible to compute [8, 13]. Some of these SDPs are weakly infeasible and one of them comes from minimizing the famous Motzkin polynomial. We closely examine this SDP in Example 4. Further, the Douglas-Rachford method presented in [16] successfully identified infeasibility of the weakly infeasible SDPs from [15]. We refer to [7] for a different algorithm that also identified infeasibility of the same instances.
- according to a classic viewpoint due to Klee [11], when (P) is weakly infeasible, the affine subspace $\{ X \, : \, \mathcal{A}X = b \}$ is an *asymptote* of $\mathcal{S}^n_+$ and the asymptotic behavior is indeed apparent on Figure 1.

In this work, we present an echelon form of weakly infeasible SDPs and bad projections of $\mathcal{S}^n_+$.

To fix basic terminology, we let $S \bullet T := \text{trace}(ST)$ be the inner product of symmetric matrices S and T, and

$$H \; = \; \{X \,|\, X \,\text{is}\, n \,\text{by}\, n \,\text{symmetric},\; \mathcal{A}X = b\,\}. \tag{1.1}$$

We represent $\mathcal{A}$ as

$$\mathcal{A}X \; = \; (A_1 \bullet X, \ldots, A_m \bullet X)^\top, \tag{1.2}$$

where the A_i are symmetric matrices.

To sketch our contributions, we revisit Example 1, where we naturally describe H in two ways. First, with equations $A_1 \bullet X = 0$ and $A_2 \bullet X = 2$, where

$$A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \text{ and } A_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{1.3}$$

This representation certifies that (SE) is infeasible.

Besides, $H = \{\lambda X_1 + X_2 \, : \, \lambda \in \mathbb{R}\}$ where

$$X_1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } X_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{1.4}$$

This generator representation proves that H is an asymptote of $\mathcal{S}^2_+$, since $\lambda X_1 + X_2$ approaches $\mathcal{S}^2_+$ as $\lambda \to +\infty$.

We see that A_1, A_2 and X_1, X_2 share a common "echelon" structure and we may wonder whether there is such a structure in every weakly infeasible SDP. The answer is naturally no, since we can easily ruin this structure even in (SE). For example, we may take linear combinations of the equations and perform congruence transformations, i.e., replace both A_i by $T^\top A_i T$ for some invertible T.

However, the same operations can untangle any weakly infeasible SDP. Precisely, in Theorem 1 we develop an echelon form of weakly infeasible SDPs with the following traits: i) it is constructed using elementary row operations and congruence transformations; ii) it makes weak infeasibility evident, since the matrices in both the equality- and in the generator representation of H have the same echelon structure iii) it enables one to construct any weakly infeasible SDP.

To elucidate the last point, we draw an analogy with basic linear algebra. We know that any infeasible linear system of equations $Ax = b$ can be converted into a form

$$\begin{array}{rcl} A'x & = & b' \\ 0^\top x & = & 1 \end{array} \tag{1.5}$$

using elementary row operations. Thus we can verify infeasibility of a linear system using the form (1.5), and construct any infeasible linear system as follows: we choose A' and b' in (1.5) arbitrarily, then perform elementary row operations. This basic algorithm always succeeds and every infeasible linear system is among its outputs.

This work shows that a similar scheme works for a more involved pathology – weak infeasibility – in a much more involved problem – an SDP. Further, in Example 4 we present an SDP that is naturally in our echelon form, without ever having to perform elementary row operations or congruence transformations. This SDP arises from a sum-of-squares (SOS) relaxation of minimizing the famous Motzkin polynomial. We thus hope that our work will be of interest to the community working on sum-of-squares optimization.

The plan of the paper is as follows. In Section 2 we review preliminaries consisting of basic linear algebra and SDP duality. In Section 3 we present and illustrate our main result, Theorem 1, and to build intution, we prove the "easy" direction. In Section 4 we describe our algorithm to construct weakly infeasible SDPs, and show that any weakly infeasible SDP is among its outputs. Our algorithm also constructs any bad projection of the psd cone. For the reader's convenience, some of the proofs are postponed to Section 5 and 6. The most difficult proof is the "hard" direction in Theorem 1, that we give in Section 6. Section 7 describes our problem library and our computational tests. In Section 8 we reinterpret Theorem 1 in two ways: as a as a "sandwich" theorem and as a "factorization" theorem. Here we also discuss open research directions.

To make the paper's results accessible to a broad audience, we prove them using only basic results in SDP duality and linear algebra, all of which we summarize in Section 2. This work has some unavoidable overlap with [15], where the lemmas of Section 5 were already proved. On the other hand, here we prove these lemmas in a more elementary fashion. Reference [15] also gave a scheme to generate weakly infeasible SDPs in a certain restricted class, however, the scheme given there does not capture even some weakly infeasible SDPs with 3×3 matrices. We comment in detail on these points in Sections 5 and Section 6.

2 Preliminaries

Matrices For a matrix or operator, say M, we denote by $\mathcal{R}(M)$ its rangespace and by $\mathcal{N}(M)$ its nullspace.

We denote by $\mathcal{S}^n$ the set of $n \times n$ symmetric matrices.

Further, N stands for the set $\{1, \dots, n\}$. Given a matrix $M \in \mathbb{R}^{n \times n}$ and $R, S \subseteq N$ we denote the submatrix of M corresponding to rows in R and columns in S by $M(R,S)$. When $R = \{r\}$ is a singleton, we simply write $M(r,S)$ for $M(\{r\},S)$. For brevity, we let $A(R) := A(R,R)$.

We denote the concatenation of matrices A and B along the diagonal by $A \oplus B$, i.e.,

$$A \oplus B := \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}.$$

Thus $M \oplus 0$ is the matrix obtained by attaching zero rows and colums to M and the dimensions of $M \oplus 0$ will be clear from the context.

Further, $X \succ 0$ means that the matrix X is symmetric and positive definite.

Basics of SDP duality The semidefinite program (P) has an optimization version that we call (P-opt) with a natural dual problem (D):

$$\text{(P-opt)} \quad \begin{array}{l} \inf\ C \bullet X \\ s.t.\ X \text{ is feasible in } (P) \end{array} \qquad\qquad \begin{array}{l} \sup\ b^\top y \\ s.t.\ \sum_{i=1}^m y_i A_i \preceq C, \end{array} \quad \text{(D)}$$

where for symmetric matrices T and S we write $T \preceq S$ to say $S - T \succeq 0$. We also say that (P-opt) is the dual of (D).

Assuming they are both feasible, the optimal value of (P-opt) is at least as large as the optimal value of (D). Besides, these optimal values agree and the optimal value of (P-opt) is attained when (D) satisfies *Slater's condition*, i.e., when there is $y \in \mathbb{R}^m$ such that $C - \sum_{i=1}^m y_i A_i \succ 0$.

We say that (P) is *strongly infeasible*, if the distance of the affine subspace H (see (1.1)) from $\mathcal{S}^n_+$ is positive. Every infeasible SDP is either strongly or weakly infeasible and strong infeasibility is the "good" kind of infeasibility: interior point methods can prove infeasibility of a strongly infeasible SDP [25, Corollary 3].

We know that (P) is strongly infeasible exactly when its *alternative system*

$$\begin{array}{rcl} \mathcal{A}^* y & \succeq & 0 \\ b^\top y & = & -1 \end{array} \qquad \text{(P-alt)}$$

is feasible.

Reformulations The following definition will be used throughout the paper. To absorb it, we need to recall how the operator $\mathcal{A}$ is represented in (1.2).

Definition 1. *We say that we* reformulate (P) *if we apply to it some of the following operations (in any order):*

(1) Exchange (A_i, b_i) and (A_j, b_j), where i and j are distinct indices in $\{1, \dots, m\}$.

(2) Replace (A_i, b_i) by $\lambda(A_i, b_i) + \mu(A_j, b_j)$, where λ and μ are reals, and $\lambda \neq 0$.

(3) Replace all A_i by $T^\top A_i T$, where T is a suitably chosen invertible matrix.

We also say that by reformulating (P) we obtain a reformulation*; and that we reformulate the map $\mathcal{A} : \mathcal{S}^n \to \mathbb{R}^m$ if we reformulate (P) with some $b \in \mathbb{R}^m$.*

Regularized facial reduction sequences To motivate our next definition, suppose that a square matrix $M \in \mathbb{R}^{n \times n}$ is in row echelon, i.e. upper triangular form

$$M = \begin{pmatrix} * & * & \dots & * \\ & * & \dots & * \\ & & \ddots & \\ & & & * \end{pmatrix}$$

where the empty cells stand for zero elements. This form serves two purposes: when the diagonal entries of M are nonzero, it makes it straightforward to see that the rangespace of M is $\mathbb{R}^n$ and that its nullspace contains only 0.

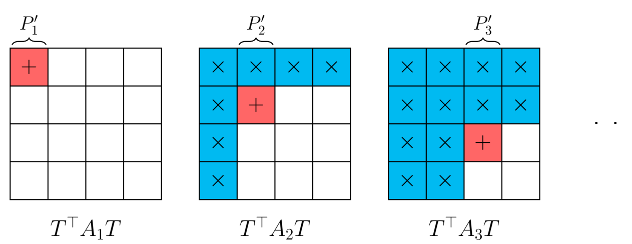
Analogously, we will use an echelon form of a sequence of symmetric matrices:

Definition 2. *We say that $(A_1, \dots, A_k)$ is a regularized facial reduction sequence with structure $\{P_1, \dots, P_k\}$ if $A_i \in \mathcal{S}^n$ for all i, the P_i are disjoint subsets of N, and for $i = 1, \dots, k$*

$$\begin{array}{rcl} A_i(P_i) & \textit{is} & \textit{diagonal with positive diagonal entries,} \\ A_i(P_1 \cup \dots \cup P_{i-1}, N) & \textit{is} & \textit{arbitrary,} \end{array} \tag{2.6}$$

and the remaining elements of all A_i are zero.

Thus, for a suitable permutation matrix T the A_i look like



where the columns of A_i with indices in P_i were permuted into columns of $T^\top A_i T$ with indices in P_i'.

Here the + red blocks are positive definite and diagonal, and the × blue blocks may have arbitrary elements.

Regularized facial reduction sequences were already defined in [15]. The definition there is more restrictive than the one here, as it assumes that the positive definite block in all A_i is the identity, and that P_1 comprises the first $|P_1|$ elements in $\{1, \dots, n\}$, P_2 comprises the next $|P_2|$ elements, and so on. Strictly speaking, the regularized facial reduction sequences in this paper should be called "relaxed regularized", but to keep the presentation simple, we will just call them regularized.

3 The main result, and the easy direction

The main result of the paper is the following.

Theorem 1. *The problem (P) is weakly infeasible if and only if it has a reformulation*

$$\begin{array}{rcl} \mathcal{A}'X & = & b' \\ X & \succeq & 0 \end{array} \tag{P_{weak}}$$

with the following properties:

1. *(1)* $(A'_1, \dots, A'_{k+1})$ *is a regularized facial reduction sequence and* $(b'_1, \dots, b'_k, b'_{k+1}) = (0, \dots, 0, -1)$ *for some* $k \geq 1$;
2. *(2) there is a regularized facial reduction sequence* $(X_1, \dots, X_{\ell+1})$ *such that* $\ell \geq 1$ *and*

$$\begin{array}{rcll} \mathcal{A}' X_i & = & 0 & \text{for } i = 1, \dots, \ell \\ \mathcal{A}' X_{\ell+1} & = & b'. & \end{array} \tag{3.7}$$

Further, the positive definite blocks in $A'_1, \dots, A'_k$ *and* $X_1, \dots, X_\ell$ *are nonempty.* □

Here we understand that $\mathcal{A}'$ is represented with symmetric matrices A'_i as

$$\mathcal{A}' X = (A'_1 \bullet X, \dots, A'_m \bullet X)^\top. \tag{3.8}$$

It is useful to visualize Theorem 1 using the matrix of inner products of the A'_i and X_j in Figure 2.

$$(A'_i \bullet X_j)_{i=1,j=1}^{m,\ell+1} = \overbrace{\left(\begin{array}{ccc|c} 0 & \dots & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & \dots & 0 & 0 \\ \hline 0 & \dots & 0 & -1 \\ \hline 0 & \dots & 0 & b'_{k+2} \\ & \ddots & & \\ 0 & \dots & 0 & b'_m \end{array} \right)}^{\ell+1} \begin{array}{l} \left.\begin{array}{l} \\ \\ \\ \\ \end{array}\right\} k+1 \\ \\ \\ \\ \end{array}$$

Figure 2: The matrix of inner products of A'_i and X_j in Theorem 1

The proof of the "only if" direction of Theorem 1 is technical and deferred to Section 6. However, the proof of the "if" direction is elementary, and we provide it below.

Proof of "if" It suffices to prove that (P_{weak}) is weakly infeasible. To that end, we first prove that it is infeasible, so to obtain a contradiction we assume that X is feasible in it. We also assume that $(A'_1, \dots, A'_{k+1})$ has structure $\{P_1, \dots, P_{k+1}\}$ (see Definition 2).

Since $A'_1 \bullet X = 0$, a positively weighted linear combination of the diagonal elements of $X(P_1)$ is zero. Since $X \succeq 0$, these elements are zero, hence the rows (and columns) of X indexed by P_1 are zero.

Continuing, $A'_2 \bullet X = 0, \dots, A'_k \bullet X = 0$ implies that the rows (and columns) of X indexed by $P_2 \cup \dots \cup P_k$ are zero. Hence the shape of A'_{k+1} and $X \succeq 0$ implies

$$A'_{k+1} \bullet X \geq 0,$$

and this contradiction proves that (P_{weak}) is indeed infeasible.

This process is illustrated on Figure 3. For convenience we assume in this figure that the columns of all matrices indexed by P_1 comes first; the columns indexed by P_2 come next; etc.

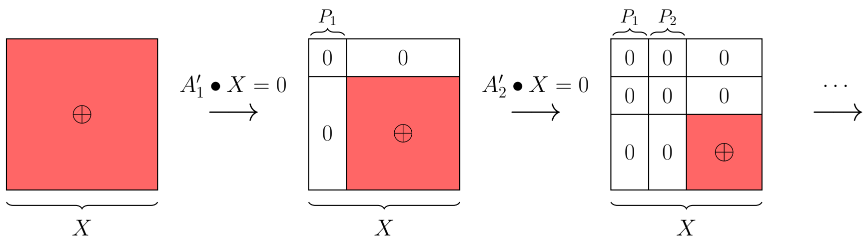


Figure 3: Proving that (P_{weak}) is infeasible

Next we prove that (P_{weak}) is not strongly infeasible. To that end, let

$$H' = \{ X \in \mathcal{S}^n : \mathcal{A}'X = b' \},$$

and fix $\epsilon > 0$. We will construct a psd matrix which is ϵ close to H'. Suppose that the structure of $(X_1, \dots, X_{\ell+1})$ is $\{Q_1, \dots, Q_{\ell+1}\}$ and for brevity, let $Q_{\ell+2} = N \setminus (Q_1 \cup \dots \cup Q_{\ell+1})$.

First we define $X_\delta \in \mathcal{S}^n$ so that $X_\delta(Q_{\ell+2}) = \delta I$ and the other elements of X_δ are zero. Here $\delta > 0$ is sufficiently small, so the norm of X_δ is at most ϵ. See the leftmost picture in Figure 4.

Second, we define $X'_{\ell+1} := X_{\ell+1} + X_\delta$. Then the $(Q_{\ell+1} \cup Q_{\ell+2})$ diagonal block of $X'_{\ell+1}$ is positive definite and $X'_{\ell+1}$ is within ϵ distance of H' (since $X_{\ell+1} \in H'$). See the middle picture in Figure 4.

Next we let $X'_\ell := \gamma_\ell X_\ell + X'_{\ell+1}$ where γ_ℓ is a positive real. From the definition of positive definiteness ($G \succ 0$ if $x^\top G x > 0$ for all nonzero x) we have that

$$X'_\ell(Q_\ell \cup Q_{\ell+1} \cup Q_{\ell+2}) \succ 0$$

if γ_ℓ is sufficiently large and X'_ℓ is still within ϵ distance of H'. We refer to the rightmost picture in Figure 4.

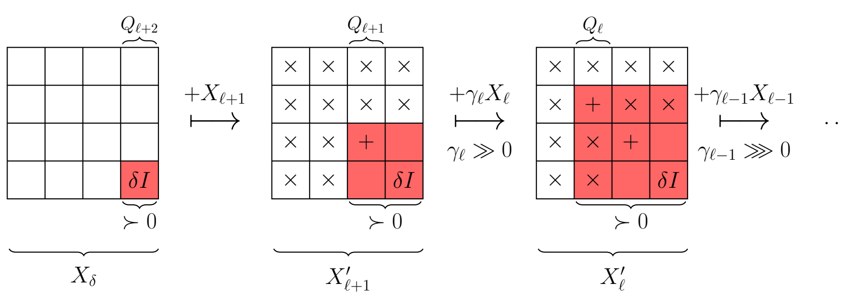


Figure 4: Proving that (P_{weak}) is not strongly infeasible

Continuing in this fashion we add $\gamma_{\ell-1} X_{\ell-1}$ to X'_ℓ for some large $\gamma_{\ell-1}$ and so on. Eventually we obtain a positive definite matrix, within ϵ distance of H', and we conclude that (P_{weak}) is not strongly infeasible. The proof is complete. $\square$

Remark 1. *We can represent the operations needed to reformulate* (P) *in a compact manner, just by two matrices. The elementary row operations can be encoded by an $m \times m$ matrix $G = (g_{ij})$ and the congruence transformations by an $n \times n$ matrix T. Then*

$$\begin{aligned} A'_i &= T^\top \left(\textstyle\sum_{j=1}^m g_{ij} A_j\right) T \text{ for } i = 1, \dots, m \\ b' &= Gb. \end{aligned} \tag{3.9}$$

Example 2. *(Example 1 continued) As a quick check, the problem (SE) needs only a minimal reformulation. To put it into the echelon form of (P_{weak}), we set $A_1' := A_1$, $A_2' := -\frac{1}{2}A_2$ and use (X_1, X_2) from equation (1.4).*

In this SDP we have $k = \ell = 1$.

A larger example follows:

Example 3. *The SDP in the form (P) with data*

$$
A_1 = \begin{pmatrix} 8 & -1 & -9 & -2 \\ -1 & -26 & 3 & 39 \\ -9 & 3 & 10 & 3 \\ -2 & 39 & 3 & -16 \end{pmatrix}, A_2 = \begin{pmatrix} 5 & -3 & -6 & -2 \\ -3 & -6 & 5 & 21 \\ -6 & 5 & 7 & 2 \\ -2 & 21 & 2 & -11 \end{pmatrix}, A_3 = \begin{pmatrix} -6 & -3 & 7 & 4 \\ -3 & 34 & 1 & -43 \\ 7 & 1 & -8 & -5 \\ 4 & -43 & -5 & 18 \end{pmatrix}
$$
$$
A_4 = \begin{pmatrix} 5 & 4 & -9 & -6 \\ 4 & -28 & 6 & 48 \\ -9 & 6 & 13 & 5 \\ -6 & 48 & 5 & -21 \end{pmatrix}, b = (-44, -22, 44, -68)^\top \tag{3.10}
$$

is weakly infeasible, but from this form this would be very difficult to tell.

However, once we reformulate it using the formulas in (3.9) and the G and T matrices

$$
G = \frac{1}{2}\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 1 & 1 & 3 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix} \quad T = \begin{pmatrix} -1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}
$$

it is brought into the form (P_{weak}) with the A_i' and X_j shown on Figure 5.

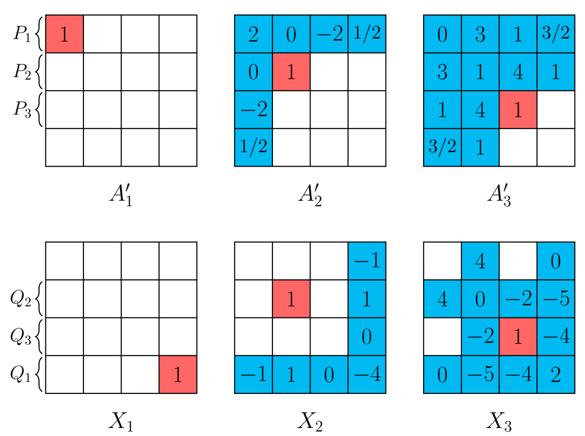


Figure 5: The A_i' and X_j obtained from reformulating the SDP (3.10)

In the reformulation the equations $A_1' \bullet X = A_2' \bullet X = 0$, $A_3' \bullet X = -1$ certify infeasibility and (X_1, X_2, X_3) certify not strong infeasibility. The matrix A_4' is omitted (and is straightforward to compute from the formulas in (3.9)). Note that now $k = \ell = 2$.

The reader may ask whether some SDPs are naturally in the echelon form of (P_{weak}) *without even having to reformulate them.* We next present such an SDP from a prominent application of semidefinite programming, polynomial optimization.

We first recall a definition. Given a multivariate polynomial $f = f(x_1, \ldots, x_n)$, we say that f is a sum of squares (SOS) if $f = \sum_{i=1}^{t} f_i^2$ for some t positive integer and f_i polynomials. An SOS polynomial is course nonnegative. On the other hand, the first example of a nonnegative polynomial that is not SOS was given by Motzkin in [18] and there are many more nonnegative polynomials than SOS polynomials [3].

Example 4. *Given the famous Motzkin polynomial*

$$f(x, y) = 1 - 3x^2y^2 + x^2y^4 + x^4y^2, \tag{3.11}$$

finding its infimum over $\mathbb{R}^2$ amounts to solving the problem

$$\begin{array}{lll} \sup & \lambda & \\ s.t. & f(x,y) - \lambda & \geq \; 0. \end{array} \tag{3.12}$$

In the SOS relaxation of (3.12) that was proposed in [12] and in [19] one solves the following problem instead:

$$\begin{array}{ll} \sup & \lambda \\ s.t. & f - \lambda \quad \textit{is SOS}. \end{array} \tag{3.13}$$

In turn, the problem (3.13) is formulated as an SDP as follows. Defining a vector of monomials [1]

$$z = (x^2, y^2, x, y, xy, xy^2, x^2y, 1).$$

we know that $f - \lambda$ is SOS if and only if $f - \lambda = X \bullet zz^\top$ for some $X \succeq 0$.

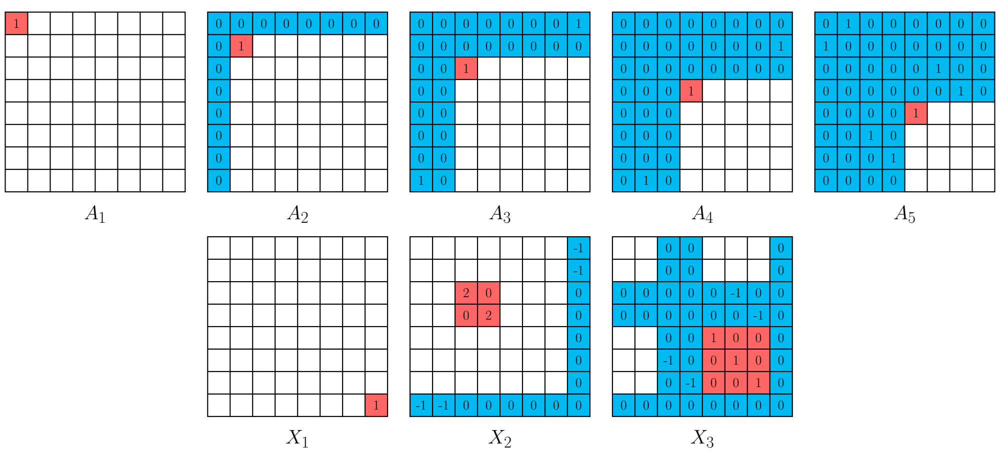


Figure 6: Certificates in the Motzkin polynomial SDP

Matching monomials in $f - \lambda$ and $X \bullet zz^\top$ we obtain the SDP

[1]To strictly follow the SOS recipe we should also include in z the monomials x^3 and y^3. We omitted these for simplicity, but it is straightforward to check that even if we do include them, the resulting SDP is still in the echelon form of (P_{weak}).

$$
\begin{array}{llcll}
\sup & -E_{88} \bullet X & & & \\
s.t. & E_{11} \bullet X & = & 0 & (x^4) \\
& E_{22} \bullet X & = & 0 & (y^4) \\
& (E_{33} + E_{18}) \bullet X & = & 0 & (x^2) \\
& (E_{44} + E_{28}) \bullet X & = & 0 & (y^2) \\
& (E_{55} + E_{12} + E_{36} + E_{47}) \bullet X & = & -3 & (x^2y^2) \\
& & \vdots & & \\
& X & \succeq & 0 &
\end{array}
\tag{3.14}
$$

In (3.14) the E_{ij} are unit matrices in $\mathcal{S}^8$ whose elements in the (i, j) and (j, i) position are 1 and the rest zero. In parentheses we show the monomial that gives the corresponding equation. For example, $E_{11} \bullet X = 0$ because $f - \lambda$ has no x^4 term. Note that in (3.14) we did not explicitly write down many equations (this is indicated by the vertical dots), for example, the equation corresponding to x^2y.

Of course, we know since the classic work of Motzkin that $f - \lambda$ is not SOS for any λ, hence (3.14) is infeasible. We next verify that it is weakly infeasible, and in the echelon form (P_{weak}) without ever having to reformulate it.

Indeed, we see that $A_1 := E_{11}, A_2 := E_{22}, \ldots, A_5 := E_{55} + E_{12} + E_{36} + E_{47}$ is a regularized facial reduction sequence, hence the equations shown in (3.14) prove it is infeasible (with -3 in place of -1).

On the other hand

$$
\begin{array}{lcl}
X_1 & = & E_{88} \\
X_2 & = & 2E_{33} + 2E_{44} - E_{18} - E_{28} \\
X_3 & = & E_{55} + E_{66} + E_{77} - E_{47} - E_{36}
\end{array}
\tag{3.15}
$$

is a regularized facial reduction sequence that proves that (3.14) is not strongly infeasible. To see why, we write the equations in (3.14) as $\mathcal{A}X = b$, then we can check that $\mathcal{A}X_1 = \mathcal{A}X_2 = 0$ and $\mathcal{A}X_3 = b$.

We can visualize the certificates on Figure 6. The certificates of infeasibility are on top and the certificates of not-strong-infeasibility are on the bottom.

We remark that [28] showed how to construct weakly infeasible SDPs from polynomial optimization problems, on the other hand this paper did not provide certificates of the kind we study in this work.

The following result characterizes linear maps that carry $\mathcal{S}^n_+$ into a nonclosed set. These maps were baptized as "bad projections of the psd cone" and explored through the lens of algebraic geometry [10].

Corollary 1. *Suppose $\mathcal{A} : \mathcal{S}^n \to \mathbb{R}^m$ is a linear map. Then $\mathcal{A}\mathcal{S}^n_+$ is not closed if and only if $\mathcal{A}$ has a reformulation $\mathcal{A}'$ with the following properties:*

(1) $(A'_1, \ldots, A'_{k+1})$ is a regularized facial reduction sequence, where $k \geq 1$;

(2) There is a regularized facial reduction sequence $(X_1, \ldots, X_{\ell+1})$, where $\ell \geq 1$ and the matrix of inner products of the the A'_i and X_j matrices looks like

$$
(A_i' \bullet X_j)_{i=1,j=1}^{m,\ell+1} \;=\; \overbrace{\left(\begin{array}{ccc|c}
0 & \dots & 0 & 0 \\
0 & \ddots & 0 & 0 \\
0 & \dots & 0 & 0 \\ \hline
0 & \dots & 0 & -1 \\ \hline
0 & \dots & 0 & \times \\
 & \ddots & & \\
0 & \dots & 0 & \times
\end{array}\right)}^{\ell+1}
\begin{array}{l} \left.\begin{array}{c} \\ \\ \\ \\ \end{array}\right\} k+1 \\ \\ \\ \\ \end{array}
$$

where the $\times$ symbols denote arbitrary elements.

□

Proof **"Only if"** Suppose $\mathcal{A}\mathcal{S}_+^n$ is not closed, and suppose $b \in \mathbb{R}^m$ is in the closure of $\mathcal{A}\mathcal{S}_+^n$ but $b \notin \mathcal{A}\mathcal{S}_+^n$. Then ($P$) is weakly infeasible, so we appeal to Theorem 1 and construct $\mathcal{A}', b'$, and $X_1, \dots, X_{\ell+1}$ therein. Then the matrix $(A_i' \bullet X_j)_{i=1,j=1}^{m,\ell+1}$ is in the form given in Figure 2, so item (2) holds.

"If" Suppose that $\mathcal{A}'$ and $(X_1, \dots, X_{\ell+1})$ are as in the statement of Corollary 1, and let $b' = \mathcal{A}' X_{\ell+1}$. Then $\mathcal{A}'$ and $(X_1, \dots, X_{\ell+1})$ satisfy (1) and (2) in Theorem 1. Hence the system (P_{weak}) therein is weakly infeasible, so b' is in the closure of $\mathcal{A}'\mathcal{S}_+^n$ but $b' \notin \mathcal{A}'\mathcal{S}_+^n$. Thus $\mathcal{A}'\mathcal{S}_+^n$ is not closed, hence neither is $\mathcal{A}\mathcal{S}_+^n$, as required. □

We next contrast Corollary 1 with an equivalent characterization of nonclosedness of $\mathcal{A}\mathcal{S}_+^n$ that we recap below in Theorem 2. Theorem 2 is obtained from [23, Theorem 1] by setting $B = 0$.

Theorem 2. *Suppose that Z is a maximum rank psd matrix in $\mathcal{R}(\mathcal{A}^*)$, the linear span of $A_1, \dots, A_m$. Assume without loss of generality that Z is of the form*

$$
Z = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix} \tag{3.16}
$$

for some $r \in \{0, \dots, n\}$. Then $\mathcal{A}\mathcal{S}_+^n$ is not closed if and only if there is a matrix $V \in \mathcal{R}(\mathcal{A}^*)$ of the form

$$
V = \begin{pmatrix} V_{11} & V_{12} \\ V_{12}^\top & V_{22} \end{pmatrix}, \tag{3.17}
$$

where $V_{22} \in \mathcal{S}_+^{n-r}$ and $\mathcal{R}(V_{12}^\top) \not\subseteq \mathcal{R}(V_{22})$. □

Next we argue that Corollary 1 is more useful than Theorem 2, although the latter is more compact. In particular, Corollary 1 can be used to construct maps under which $\mathcal{S}_+^n$ is not closed, as we show in Section 4.

On the other hand, Theorem 2 cannot be used for this purpose in a straightforward manner. Suppose indeed that we try to construct such an $\mathcal{A}$ and choose $A_1 := Z$ as in (3.16) (with $0 < r < n$) and $A_2 := V$ as in (3.17) as elements of $\mathcal{R}(\mathcal{A}^*)$. However, we cannot guarantee that Z remains a maximum rank psd matrix in $\mathcal{R}(\mathcal{A}^*)$ after we chose the other A_i : for example, Z would certainly stop being a maximum rank psd matrix in $\mathcal{R}(\mathcal{A}^*)$ if we pick $A_3 := I$.

4 How to construct any weakly infeasible SDP and bad projection of the psd cone

We now build on Theorem 1, and present a combinatorial algorithm, Algorithm 1, to construct weakly infeasible SDPs of the form (P). In fact, we will show that Algorithm 1 can construct *any* such SDP. Algorithm 1 (with Algorithm 2 as a subroutine) also constructs any bad projection of the psd cone, and provides a vector b in the closure of $\mathcal{A}\mathcal{S}_+^n$ such that $b \notin \mathcal{A}\mathcal{S}_+^n$.

If $\mathcal{A}'$, b' and $X_1, \dots, X_{\ell+1}$ are as in Theorem 1, then the matrix of their inner products looks like in Figure 2. Hence the following algorithm generates a weakly infeasible SDP, and any weakly infeasible SDP is among its outputs.

To simplify notation, in Algorithms 1 and 2 we write A_i in place of A_i'.

Algorithm 1 Construct Weak SDP

1: Choose k and ℓ positive integers and $(A_1, \dots, A_{k+1})$ and $(X_1, \dots, X_{\ell+1})$ regularized facial reduction sequences, which satisfy the following *base equations:*

$$\boxed{A_i \bullet X_j = \begin{cases} 0 & \text{if } (i,j) \neq (k+1, \ell+1) \\ -1 & \text{if } (i,j) = (k+1, \ell+1) \end{cases}} \tag{BASE}$$

2: Choose $A_{k+2}, \dots, A_m$ so they have zero $\bullet$ product with $\{X_1, \dots, X_\ell\}$.
3: Set $b_i := A_i \bullet X_{\ell+1} \quad i = 1, \dots, m.$
4: Reformulate (P).

Step 1 ensures that the first $k+1$ rows of the $(A_i \bullet X_j)_{i,j=1}^{k+1,\ell+1}$ matrix look like as required in Figure 2. Steps 2 and 3 ensure that the rest of the matrix looks like as required in the same figure.

The only nontrivial step in Algorithm 1 is Step 1, so the question is, how to carry out this step?

The main idea is that we have many degrees of freedom in choosing the nonzero blocks in the A_i and the X_j, and only a small number of equations to satisfy. Precisely, by a straightforward count the A_i alltogether have at least constant times k^3 potentially nonzero blocks of the form $A_i(P_s, P_t)$, where $\{P_1, \dots, P_{k+1}\}$ is the structure of $(A_1, \dots, A_{k+1})$. Similarly, the X_j have at least constant times ℓ^3 potentially nonzero blocks. So if we set these blocks in the right order, then the A_i and X_j will satisfy the base equations, of which there are only $(k+1)(\ell+1)$.

To carry out this plan, we need two lemmas.

Lemma 1. *Suppose that $(A_1, \dots, A_{k+1})$ is a regularized facial reduction sequence with structure $\{P_1, \dots, P_{k+1}\}$ and $(X_1, \dots, X_{\ell+1})$ is a regularized facial reduction sequence with structure $\{Q_1, \dots, Q_{\ell+1}\}$.*

Also suppose that $(A_1, \dots, A_{k+1})$ and $(X_1, \dots, X_{\ell+1})$ satisfy the base equations (BASE). Then the following hold:

$$P_1 \cap (Q_1 \cup \dots \cup Q_{\ell+1}) = \emptyset \tag{4.18}$$
$$Q_1 \cap (P_1 \cup \dots \cup P_{k+1}) = \emptyset. \tag{4.19}$$

Proof We prove (4.19), the proof of (4.18) is analogous. Since $A_1, \dots, A_{k+1}$ is a regularized facial reduction sequence, and $X_1 \succeq 0$, an argument like in the "if" direction in Theorem 1 proves $X_1(P_1 \cup \cdots \cup P_k, N) = 0$. Since the only nonzero entries of X_1 are in $X_1(Q_1)$, the statement follows. $\square$

In Lemma 2 the b_j reals are indexed from 2 to $\ell+1$ for convenience. Lemma 2 shows how to solve a linear system of equations in an unusual setup, in which only the right hand side is fixed. For Lemma 2 we need to define the inner product of possibly nonsymmetric matrices M and Y as the trace of $M^T Y$.

Lemma 2. *Given positive integers p, q and ℓ, and real numbers $b_2, \ldots, b_{\ell+1}$ there is a polynomial time algorithm to find $M, Y_2, \ldots, Y_{\ell+1}$ in $\mathbb{R}^{p\times q}$ such that*

$$\begin{aligned} M \bullet Y_2 &= b_2 \\ &\vdots \\ M \bullet Y_{\ell+1} &= b_{\ell+1}. \end{aligned} \tag{4.20}$$

Furthermore, any solution to (4.20) is a possible outcome of this algorithm.

Proof If all b_j are zero, we first choose an arbitrary M, then choose $Y_2, \ldots, Y_{\ell+1}$ to solve the system (4.20). If not all b_j are zero, we proceed in the same fashion, but we make sure to pick $M \neq 0$. $\square$

The following algorithm constructs facial reduction sequences that satisfy the base equations (BASE).

Algorithm 2 Base Equations Algorithm

1: Choose positive integers k and ℓ and regularized facial reduction sequences $(A_i)_{i=1}^{k+1}$ with structure $\{P_i\}_{i=1}^{k+1}$ and $(X_j)_{j=1}^{\ell+1}$ with structure $\{Q_j\}_{j=1}^{\ell+1}$ which satisfy

$$P_1 \neq \emptyset, \ldots, P_k \neq \emptyset,\ Q_1 \neq \emptyset,\ (4.18),\ \text{and } (4.19).$$

2: **for** $i = 2 : k+1$ **do**

3: Set $A_i(P_{i-1}, Q_1), X_2(P_{i-1}, Q_1), \ldots, X_{\ell+1}(P_{i-1}, Q_1)$ to satisfy the base equations (BASE) with left hand side $A_i \bullet X_2, \ldots, A_i \bullet X_{\ell+1}$.

4: **end for**

Lemma 3. *Step 3 of Algorithm 2 can be implemented so that the algorithm is correct.*

Proof For brevity, define $P_{k+2} := N \setminus (P_1 \cup \cdots \cup P_{k+1})$. Let us fix $i \in \{2, \ldots, k+1\}$. To implement step 3 of Algorithm 1, we first set the (P_{i-1}, Q_1) block of $A_i, X_2, \ldots, X_{\ell+1}$ to zero, then introduce a target vector $(b_2, \ldots, b_{\ell+1})^\top$:

$$b_j = \begin{cases} -\frac{1}{2} A_i \bullet X_j & \text{if } (i,j) \neq (k+1, \ell+1) \\ -\frac{1}{2}\big(A_i \bullet X_j + 1\big) & \text{if } (i,j) = (k+1, \ell+1). \end{cases}$$

Next we invoke Lemma 2 with $A_i(P_{i-1}, Q_1)$ in place of M and $X_j(P_{i-1}, Q_1)$ in place of Y_j for $j = 2, \ldots, \ell+1$. Finally we symmetrize A_i and the X_j, namely we set $A_i(Q_1, P_{i-1}) := A_i(P_{i-1}, Q_1)^\top$ and $X_j(Q_1, P_{i-1}) := X_j(P_{i-1}, Q_1)^\top$.

It remains to show that performing step 3 in Algorithm 2 keeps the previously satisfied equations satisfied. Suppose we perform step 3 with a certain $i \in \{3, \ldots, k+1\}$. After this the base equations (BASE) with left hand side $A_i \bullet X_j$ for $j = 2, \ldots, \ell+1$ are satisfied.

Let us fix $t \in \{2, \ldots, i-1\}$ and $j \in \{2, \ldots, \ell+1\}$. We will show that the equations with left hand side $A_t \bullet X_j$ remain satisfied.

For that, we note that the support of A_t is contained in the blocks

$$\begin{aligned} \mathcal{I}_1 &:= (P_1 \cup \cdots \cup P_t, P_1 \cup \cdots \cup P_t) \\ \mathcal{I}_2 &:= (P_1 \cup \cdots \cup P_{t-1}, P_{t+1} \cup \cdots \cup P_{k+2}) \\ \mathcal{I}_3 &:= (P_{t+1} \cup \cdots \cup P_{k+2}, P_1 \cup \cdots \cup P_{t-1}) \end{aligned} \tag{4.21}$$

cf. Definition 1. In iteration i we change $X_j(P_{i-1}, Q_1)$ and $X_j(Q_1, P_{i-1})$ for $j = 2, \ldots, \ell+1$. So it suffices to show

$$(P_{i-1}, Q_1) \cap \mathcal{I}_1 = \emptyset \tag{4.22}$$
$$(P_{i-1}, Q_1) \cap \mathcal{I}_2 = \emptyset \tag{4.23}$$
$$(Q_1, P_{i-1}) \cap \mathcal{I}_3 = \emptyset. \tag{4.24}$$

Indeed, (4.22) follows by (4.19). Further, (4.23) and (4.24) follow from $t-1 < i-1$ and the proof is complete. □

In summary, we have the following theorem.

Theorem 3. *Algorithm 1 always correctly constructs a weakly infeasible SDP and any weakly infeasible SDP of the form (P) is among its outputs.*

Proof Lemmas 1–3 imply that Algorithm 1 always correctly outputs a weakly infeasible SDP. On the other hand, suppose that (P) is weakly infeasible and defined by linear map $\mathcal{A}$ and right hand side b. Then (P) has a reformulation (P_{weak}) as presented in Theorem 1. For simplicity, let us denote the operator in (P_{weak}) by $\mathcal{A}$ and represent $\mathcal{A}$ with matrices $A_1, \ldots, A_m$. Also let us denote the right hand side in (P_{weak}) by b.

Assume that the first $k \geq 1$ equations in (P_{weak}) prove that it is infeasible. We know that (P_{weak}) is not strongly infeasible, and we let $(X_1, \ldots, X_{\ell+1})$ be the sequence that certifies this as presented in Theorem 1. Here $\ell \geq 1$.

Suppose that $(A_1, \ldots, A_k)$ has structure $\{P_1, \ldots, P_{k+1}\}$ and $(X_1, \ldots, X_{\ell+1})$ has structure $\{Q_1, \ldots, Q_{\ell+1}\}$. We know from Theorem 1 that $P_1, \ldots, P_k$ and $Q_1, \ldots, Q_\ell$ are nonempty, in particular, Q_1 is nonempty. Hence, by Lemma 2 we see that $(A_1, \ldots, A_{k+1})$ and $(X_1, \ldots, X_{\ell+1})$ are among the possible outputs of Algorithm 2. Besides, $A_{k+2}, \ldots, A_m$ and $b_{k+2}, \ldots, b_m$ are possible outputs of Steps 2 and 3 in Algorithm 1. Because of the reformulation step 4 we see that (P) is a possible output of Algorithm 1, and the proof is complete. □

As a quick check, Algorithm 1 can construct a variant of (SE) by starting with $k = \ell = 1$, $P_1 = \{1\}$, $Q_1 = \{2\}$, $P_2 = Q_2 = \emptyset$ and

$$A_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, A_2 = \begin{pmatrix} 0 & \alpha \\ \alpha & 0 \end{pmatrix}, X_1 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, X_2 = \begin{pmatrix} 0 & \beta \\ \beta & 0 \end{pmatrix},$$

where α and β are arbitrary. Then Algorithm 2 sets α and β to satisfy $2\alpha\beta = -1$ then sets $b = (0, -1)^\top$.

A larger example is

Example 5. *Let $k = 1$, $\ell = 1$, $P_1 = \{1\}$, $P_2 = \{2\}$, $Q_1 = \{3\}, Q_2 = \{2\}$, α and β arbitrary reals, and*

$$A_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, A_2 = \begin{pmatrix} 0 & 0 & \alpha \\ 0 & 1 & 0 \\ \alpha & 0 & 0 \end{pmatrix}, X_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, X_2 = \begin{pmatrix} 0 & 0 & \beta \\ 0 & 1 & 0 \\ \beta & 0 & 0 \end{pmatrix}.$$

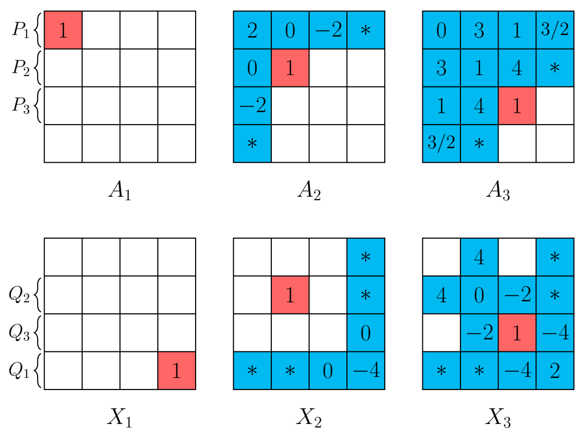
Our algorithms construct a weakly infeasible SDP from this data as follows: Algorithm 2 sets α and β so that $\alpha\beta = -1$. After this the A_i and X_j satisfy the base equations (BASE). Then step 2 of Algorithm 1 chooses

$$A_3 := \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix},$$

as it has zero inner product with X_1. Finally. step 3 sets $b = (0, -1, 0)^\top$. Note that we have $k = \ell = 1$.

We note that a scheme to construct weakly infeasible SDPs was given in [15]. However, in that scheme we assumed that the positive definite blocks of the A_i and X_j matrices do not overlap, so the variety of weakly infeasible SDPs it can generate is limited. In particular, it cannot even construct the small SDP of Example 5.

Example 6. *(Example 3 continued) We now show how Algorithm 1 generates the SDP in Example 3. We first select (A_1, A_2, A_3) and (X_1, X_2, X_3) regularized facial sequences below:*



Here at the beginning the entries marked by $*$ are arbitrary. We first use Lemma 2 to ensure $A_2 \bullet X_2 = A_2 \bullet X_3 = 0$ by setting

$$A_2(1,4) = 1/2,\ X_2(1,4) = -1,\ X_3(1,4) = 0.$$

We then ensure $A_3 \bullet X_2 = 0, A_3 \bullet X_3 = -1$ by setting

$$A_3(2,4) = 1,\ X_2(2,4) = 1,\ X_3(2,4) = -5.$$

(Of course we also set the symmetric entries, for example, we set $A_2(4,1) = 1/2$ and so on.)

We finally select A_4 to be orthogonal to X_1 and X_2

$$A_4 = \frac{1}{2}\begin{pmatrix} -1 & -2 & -1 & 3 \\ -2 & 2 & -1 & 2 \\ -1 & -1 & 0 & -1 \\ 3 & 2 & -1 & 0 \end{pmatrix}$$

and set $b_i = A_i \bullet X_3$ for $i = 1, \ldots, 4$. We thus get the weakly infeasible instance of Example 3 with the A_i and X_j shown above and $b = (0, 0, -1, -12)^\top$.

5 Proofs: certificates of infeasibility and not strong infeasibility separately

In this section we construct two distinct reformulations of (P): one to certify that it is infeasible, and the second to certify that it is not strongly infeasible. Lemma 4 already appeared in [14] and Lemma 6 already appeared in [15]. The latter result closely followed the arguments in [17].

Here we give shorter and more elementary proofs.

We first state a necessary condition for a semidefinite system to be infeasible.

Lemma 4. *Suppose $B \in \mathcal{S}^n$ and $L \subseteq \mathcal{S}^n$ is a subspace such that $(B + L) \cap \mathcal{S}^n_+ = \emptyset$.*

Then the system (5.25) is feasible.

$$\begin{array}{rl} B \bullet Y & \leq 0 \\ Y & \in L^\perp \cap \left(\mathcal{S}^n_+ \setminus \{0\}\right). \end{array} \tag{5.25}$$

Proof Let $\mathcal{B} : \mathbb{R}^m \to \mathcal{S}^n$ be a linear map whose rangespace is L. We claim that the optimal value of the SDP

$$\begin{array}{rl} \sup & y_0 \\ s.t. & -By_0 - \mathcal{B}y \preceq 0 \end{array} \tag{5.26}$$

is zero. Indeed, its optimal value is nonnegative, since $(y, y_0) = (0, 0)$ is feasible in it. On the other hand, the optimal value cannot be positive: if $y_0 > 0$ were feasible with some y, then we would get the contradiction $B + \frac{1}{y_0}\mathcal{B}y \succeq 0$.

First assume that (5.26) satisfies Slater's condition. Then its dual (which is of the form (P-opt)), is feasible. Any Y feasible in the dual of (5.26) satisfies

$$B \bullet Y = -1, \, \mathcal{B}^* Y = 0,$$

as required.

Second, assume that (5.26) does not satisfy Slater's condition. We claim that the optimal value of the SDP

$$\begin{array}{rl} \sup & t \\ s.t. & tI - By_0 - \mathcal{B}y \preceq 0 \end{array} \tag{5.27}$$

is zero. Indeed, it is nonnegative since setting all variables to zero we obtain a feasible solution. On the other hand if $t > 0$ were feasible with some y_0 and y, then the contradiction $By_0 + \mathcal{B}y \succeq tI \succ 0$ would follow.

Note that (5.27) does satisfy Slater's condition with $t = -1$, $y = 0$, and $y_0 = 0$. Thus there is a Y feasible in the dual of (5.27), which satisfies

$$B \bullet Y = 0, \, \mathcal{B}^* Y = 0, \, I \bullet Y = 1,$$

as required. $\square$

Lemma 5. *The SDP (P) is infeasible if and only if it has a reformulation*

$$\begin{array}{rcl} \mathcal{A}'X & = & b' \\ X & \succeq & 0 \end{array} \tag{P_{infeas}}$$

in which $(A'_1, \ldots, A'_{k+1})$ is a regularized facial reduction sequence and $(b'_1, \ldots, b'_k, b'_{k+1}) = (0, \ldots, 0, -1)$ for some $k \geq 0$.

Proof The "if" direction can be found verbatim in the proof of the "if" direction in Theorem 1.

To show the "only if" direction, we proceed by induction on m, the number of equations in the system $\mathcal{A}X = b$. We start the process with $\mathcal{A}' := \mathcal{A},$ and $b' := b$.

Case $m = 1$ After applying a congruence transformation, we can assume that $A_1 = \Lambda \oplus 0$ where Λ is diagonal. Since (P) is infeasible, we can assume $\Lambda \geq 0$ and $b_1 < 0$. Let r be the rank of Λ. After possibly one more congruence transformation and a rescaling, we set

$$k := 0,\ A_1' := I_r \oplus 0,\ b_1' := -1, \tag{5.28}$$

(possibly with $r = 0$) and stop.

Case $m > 1$ If the linear system $\mathcal{A}X = b$ is infeasible, then by elementary linear algebra there is y such that

$$\mathcal{A}^* y = 0,\ b^\top y = -1.$$

So we execute the operations listed in (5.28) with $r = 0$ and stop.

If the linear system system $\mathcal{A}X = b$ is feasible, we fix $X_0 \in \mathcal{S}^n$ such that $\mathcal{A}X_0 = b$. We apply Lemma 4 with $L := \mathcal{N}(\mathcal{A})$ and $B := X_0$ and find Y feasible in the system (5.25). We have that $Y \in L^\perp,$ and $L^\perp = \mathcal{R}(\mathcal{A}^*)$ so we write $Y = \mathcal{A}^* y$ for some y and deduce

$$0 \geq X_0 \bullet Y \ = \ X_0 \bullet \mathcal{A}^* y = (\mathcal{A}X_0)^\top y = b^\top y.$$

Since $Y \neq 0,$ there exists an orthogonal T matrix such that $T^\top Y T = I_r \oplus 0$ for some $r \geq 1$. We set

$$A_1' \ := \ I_r \oplus 0,\ b_1' := b^\top y,$$

and replace A_i' by $T^\top A_i' T$ for $i = 2, \ldots, m$.

If $b_1' < 0$ then we rescale y (and Y) so that $b_1' = -1$. Note that we must have $b_1' < 0$ if $r = n$. We then set $k = 0$ and stop.

If $b_1' = 0,$ then we must have $r < n$. We then delete the equation $A_1' \bullet X = 0$ and also delete the first r rows and columns from all the other A_i'. We thus obtain a smaller SDP, say $(P'),$ with $m-1$ equations and order $n-r$ matrices. We see that (P') is infeasible (if X' were feasible in it, then $X := 0 \oplus X'$ would be feasible in (P)). So we proceed by induction, as a reformulation of (P') into the form of (P_{infeas}) yields a reformulation of (P) into the same form.

∎

As a quick sanity check, we consider the SDP in (SE), with $k = 1$ and A_1 and A_2 given in (1.3). After a minimal reformulation, namely setting $A_2' = -\frac{1}{2}A_2,$ it is in the form (P_{infeas}).

The proof of Lemma 5 actually implies that the positive definite blocks in $A_1', \ldots, A_k'$ are nonempty, and can be chosen as identity matrices. However, the positive definite block in A_{k+1}' may be empty, as it is in (SE), or in its reformulated version given in the previous paragraph.

We next present Lemma 6 to construct a certificate that (P) is not strongly infeasible.

Lemma 6. *The SDP (P) is not strongly infeasible if and only if it has a reformulation*

$$\begin{array}{rcl} \mathcal{A}'' X & = & b'' \\ X & \succeq & 0, \end{array} \tag{$P_{\text{notstrong}}$}$$

such that for some regularized facial reduction sequence $(X_1, \ldots, X_{\ell+1})$ *with* $\ell \geq 0$ *the following holds:*

$$\begin{array}{rcl} \mathcal{A}'' X_i & = & 0 \textit{ for } i = 1, \ldots, \ell \textit{ and} \\ \mathcal{A}'' X_{\ell+1} & = & b''. \end{array} \tag{5.29}$$

Proof The proof of the "if" direction is already contained in the proof of the "if" direction in Theorem 1.

To show the "only if" direction, we assume that (P) is not strongly infeasible, and we choose an operator $\mathcal{B} : \mathcal{S}^n \rightarrow \mathbb{R}^m$ such that $\mathcal{R}(\mathcal{A}^*) = \mathcal{N}(\mathcal{B})$. We also choose $X_0 \in \mathcal{S}^n$ such that $\mathcal{A}X_0 = b$ (such an X_0 must exist, otherwise (P) would be strongly infeasible). Since (P) is not strongly infeasible, the alternative system (P-alt) is infeasible. We claim that (P-alt) is equivalent to

$$\begin{array}{rcl} \mathcal{B}Y & = & 0 \\ X_0 \bullet Y & = & -1 \\ Y & \succeq & 0. \end{array} \tag{5.30}$$

Indeed, by the choice of $\mathcal{B}$ we have that $\mathcal{B}Y = 0$ for some $Y \in \mathcal{S}^n$ iff $Y = \mathcal{A}^* y$ for some $y \in \mathbb{R}^m$. For any such Y and y we see that

$$b^\top y \;=\; (\mathcal{A}X_0)^\top y \;=\; X_0 \bullet Y,$$

and this proves that (P-alt) and (5.30) are equivalent.

Thus, by Lemma 5, the system (5.30) has a reformulation of the form (P_{infeas}), in which for some $\ell \geq 0$ the first $\ell + 1$ equations prove the infeasibility. These equations are of the form

$$\begin{array}{rcl} X_j \bullet Y & = & 0 \, (j = 1, \ldots, \ell) \\ X_{\ell+1} \bullet Y & = & -1, \end{array} \tag{5.31}$$

where $(X_1, \ldots, X_{\ell+1})$ is a regularized facial reduction sequence.

Note that in (5.30) the only equation with nonzero right hand side is $X_0 \bullet Y = -1$. Given that from (5.30) we derived equations (5.31) by elementary row operations and by congruence transformations, we see that

$$\begin{array}{rcll} X_j & \in & T^\top \mathcal{R}(\mathcal{B}^*) T & \text{for } j = 1, \ldots, \ell \\ X_{\ell+1} & \in & T^\top \big(\mathcal{R}(\mathcal{B}^*) + X_0 \big) T & \end{array} \tag{5.32}$$

for some invertible matrix T.

Observe that $\mathcal{R}(\mathcal{B}^*) = \mathcal{N}(\mathcal{A})$. Then from (5.32) we deduce that for $i = 1, \ldots, m$

$$A_i \bullet T^{-\top} X_j T^{-1} \;=\; \begin{cases} 0 & \text{if } j \in \{1, \ldots, \ell\} \\ b_i & \text{if } j = \ell + 1 \end{cases}$$

holds. We have $A_i \bullet T^{-\top} X_j T^{-1} \;=\; T^{-1} A_i T^{-\top} \bullet X_j$ for all i. We define the operator $\mathcal{A}''$ as

$$\mathcal{A}'' X = (A_1'' \bullet X, \ldots, A_m'' \bullet X)^\top,$$

where $A_i'' = T^{-1} A_i T^{-\top}$ for all i and let $b'' = b$. We see that $\mathcal{A}'', b''$, and the $X_1, \ldots, X_{\ell+1}$ that we already defined satisfy the requirements of our lemma. □

For another sanity check, suppose we let A_1 and A_2 as in (1.3), and we set $A_1'' = A_1$ and $A_2' = -\frac{1}{2} A_2$, and X_1 and X_2 as in (1.4). Then (A_1'', A_2'') and (X_1, X_2) with $b = (0, -1)^\top$ satisfy the conclusions of Lemma 6.

6 Proof of Theorem 1

Section 5 showed how to produce a reformulation (P_{infeas}) to prove that (P) is infeasible; and another reformulation ($P_{\text{notstrong}}$) to prove that it is not strongly infeasible. In this section we show that a single reformulation can accomplish both. This reformulation was fairly straightforward to produce when we started with a simple problem like (SE). In the general case we need a technical proof.

We will use operators that transform a certain targeted block of a matrix, bringing us to Definition 3 below. To absorb it, we need to recall the notation $M(R, S)$ and $M(R)$ for blocks of a matrix M from the start of Section 2.

Definition 3. *Suppose $R \subseteq N$ and G is matrix of order $|R|$. The matrix $I_{R,G}$ is obtained from the $n \times n$ identity by replacing $I(R)$ by G, i.e., by performing the following two steps:*

$$\begin{aligned} I_{R,G} &:= I, \\ I_{R,G}(R) &:= G. \end{aligned}$$

For example, if $n = 4$, $R = \{1, 4\}$, and $G = \begin{pmatrix} 2 & 3 \\ 4 & 5 \end{pmatrix}$, then

$$I_{R,G} = \begin{pmatrix} 2 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 4 & 0 & 0 & 5 \end{pmatrix}.$$

Suppose $M \in \mathbb{R}^{n \times n}$ and $R \subseteq N$. Then the operation

$$M := M I_{R,G}$$

right multiplies by G the columns of M indexed by R and leaves the rest of M unchanged.

Given subsets $R_1, \ldots, R_t$ of N and indices i and j such that $1 \leq i \leq j \leq t$ we will use the following shorthand:

$$R_{i:j} := R_i \cup R_{i+1} \cup \cdots \cup R_j. \tag{6.33}$$

We will often use a congruence transformation to put matrices into a convenient block diagonal form, bringing us to the following lemma:

Lemma 7. *Suppose $X \in \mathcal{S}^n$ and $R_1, \ldots, R_t$ are disjoint subsets of N such that*

$$X(R_{1:t}) \succeq 0.$$

Then there is an invertible matrix T such that

$$(T^\top XT)(R_{1:t}) \text{ is nonnegative diagonal,}$$

and T can be chosen as the product of $n \times n$ invertible matrices

$$T = I_{R_1,U_1} W_1 \ldots I_{R_t,U_t} W_t,$$

where

(1) the U_i are orthonormal matrices for all i.

(2) right multiplying an $n \times n$ matrix, say M, by W_i adds multiples of columns in $M(N, R_i)$ to columns of $M(N, R_j)$ for some indices $j \in \{i + 1, \ldots, t\}$.

□

Suppose W_i is a matrix in the statement of the Lemma 7. We can describe W_i algebraically as follows: i) it has all 1 entries on the main diagonal; ii) the block $W_i(R_i, R_j)$ may be nonzero for some $j \in \{i+1, \ldots, t\}$; iii) all the other blocks of W_i are zero.

Proof of Lemma 7 Let U_1 be a matrix of orthonormal eigenvectors of $X(R_1)$ and define $T := I_{R_1,U_1}$. Then $(T^\top XT)(R_{1:t})$ looks like on the first picture of Figure 7 below, with $\times$ representing arbitrary elements.

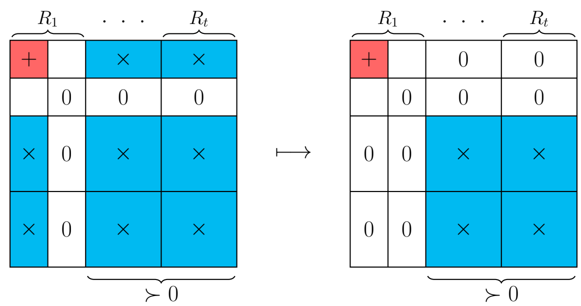


Figure 7: How to diagonalize X in Lemma 7

Next we let W_1 be a matrix such that right multiplying $T^\top XT$ by W_1 adds columns of $T^\top XT$ indexed by R_1 to columns indexed by R_j to zero out the $T^\top XT(R_1, R_j)$ block for all $1 < j$. Then the $R_{1:t}$ region of $W_1^\top T^\top XTW_1$ looks like in the right picture on Figure 7.

We then redefine $T := TW_1$, let $X' := T^\top XT$, and continue in like fashion with the $R_{2:t}$ diagonal block of X'. $\square$

The next definition is from the theory of facial reduction [4, 22].

Definition 4. *We say that a sequence of symmetric matrices $X_1, \ldots, X_t$ is a* facial reduction sequence for $\mathcal{S}^n_+$ *if*

$$X_1 \in \mathcal{S}^n_+, \text{ and } X_{i+1} \in \left(\mathcal{S}^n_+ \cap X_1^\perp \cap \cdots \cap X_i^\perp\right)^* \text{ for } i = 1, \ldots, t-1.$$

Here, for a set $C \subseteq \mathcal{S}^n$ we write $C^* = \{Y : X \bullet Y \geq 0 \text{ for all } X \in C\}$ for its dual cone.

Evidently, a regularized facial reduction sequence is a facial reduction sequence, but the converse is not true in general.

Lemma 8 below follows from Lemma 1 in [15]; however, below we give a simpler proof.

Lemma 8. *Suppose that $(X_1, \ldots, X_t)$ is a facial reduction sequence, and V is an invertible matrix. Then $(V^\top X_1 V, \ldots, V^\top X_t V)$ is also a facial reduction sequence.*

Proof Let $(X_1, \ldots, X_t)$ be as stated. For brevity, define the map $\mathcal{V} : \mathcal{S}^n \to \mathcal{S}^n$ as $\mathcal{V}X = V^\top XV$ for $X \in \mathcal{S}^n$. Then the conjugate of $\mathcal{V}$ is computed as $\mathcal{V}^*Y = VYV^\top$ for $Y \in \mathcal{S}^n$.

Let us fix $i \in \{1, \ldots, t-1\}$ and let $Y \in \mathcal{S}^n_+ \cap (\mathcal{V}X_1)^\perp \cap \cdots \cap (\mathcal{V}X_i)^\perp$. We will show

$$\mathcal{V}X_{i+1} \bullet Y \geq 0, \tag{6.34}$$

and this will prove our claim. From the definition of the conjugate we deduce

$$\mathcal{V}^*Y \in \mathcal{S}^n_+ \cap X_1^\perp \cap \cdots \cap X_i^\perp. \tag{6.35}$$

Hence $\mathcal{V}X_{i+1} \bullet Y = X_{i+1} \bullet \mathcal{V}^* Y \geq 0$, where the inequality follows from (6.35) and from $(X_1, \dots, X_t)$ being a facial reduction sequence. Hence (6.34) follows and the proof is complete. □

We can now prove the difficult direction in Theorem 1.

Proof of "only if" in Theorem 1 Suppose that (P) is weakly infeasible.

We let (P_{infeas}) be the reformulation produced by Lemma 5 with operator $\mathcal{A}'$ and right hand side b'. We claim that $k \geq 1$, so to obtain a contradiction, suppose $k = 0$. Then (P-alt) is feasible (with $\mathcal{A}^* y = A'_1$), hence (P) is strongly infeasible, which is the desired contradiction.

We also fix ($P_{\text{notstrong}}$), the reformulation produced by Lemma 6 with operator $\mathcal{A}''$ and right hand side b''. Further, we let $(X_1, \dots, X_{\ell+1})$ be the regularized facial reduction sequence produced by Lemma 6. We must have $\ell \geq 1$, since if ℓ were 0, then X_1 would be feasible in (P), which would be a contradiction.

As usual, we represent the operator $\mathcal{A}'$ with matrices A'_i and the operator $\mathcal{A}''$ with matrices A''_i for $i = 1, \dots, m$. Further, following the proof of Lemma 4 we assume without loss of generality that the positive definite blocks in the A'_i are identities.

If ($P_{\text{notstrong}}$) is the same as (P_{infeas}), then there is nothing to do. Otherwise, since both are reformulations of (P), we can transform ($P_{\text{notstrong}}$) into (P_{infeas}) if we

(1) perform a sequence of elementary row operations on the equations $A''_i \bullet X = b''_i$; then

(2) replace all A''_i by $V^\top A''_i V$ for some invertible matrix V.

Suppose we first perform only the elementary row operations in (1), and for simplicity we still call the resulting reformulation ($P_{\text{notstrong}}$) with operator $\mathcal{A}''$ (represented by matrices A''_i) and right hand side b''. Afterwards equations (5.29) still hold. At this point we have

$$\begin{array}{rcll} A'_i & = & V^\top A''_i V & \text{for } i = 1, \dots, m, \\ A''_i \bullet X_s & = & V^\top A''_i V \bullet V^{-1} X_s V^{-\top} & \text{for } i = 1, \dots, m; \text{ for } s = 1, \dots, \ell + 1. \end{array} \tag{6.36}$$

We next perform the following operations:

$$X_s := V^{-1} X_s V^{-\top} \quad \text{for } s = 1, \dots, \ell + 1. \tag{6.37}$$

Let us consider the following *invariant conditions*, where $j \in \{0, \dots, \ell + 1\}$:

(INV-1): The semidefinite system (P_{infeas}) is a reformulation of (P) with properties given in Lemma 5. In particular, $(A'_1, \dots, A'_{k+1})$ is a regularized facial reduction sequence and $b'_1 = \dots = b'_k = 0, b'_{k+1} = -1$.

(INV-2):

$$\begin{array}{rcl} \mathcal{A}' X_s & = & 0 \text{ for } s = 1, \dots, \ell \\ \mathcal{A}' X_{\ell+1} & = & b'. \end{array} \tag{6.38}$$

(INV-3): $(X_1, \dots, X_{\ell+1})$ is a facial reduction sequence.

(INV-4): $(X_1, \dots, X_j)$ is a regularized facial reduction sequence.

We claim that all these conditions hold when $j = 0$. Indeed, (**INV-1**) holds since (P_{infeas}) was constructed in Lemma 5. Condition (**INV-4**) holds vacuously. Condition (**INV-2**) holds by (6.36) and since we performed the operations in (6.37). Finally, condition (**INV-3**) holds by Lemma 8, since we started with $(X_1, \ldots, X_{\ell+1})$ being a regularized facial reduction sequence, then we performed operations (6.37).

The goal is to have the invariant conditions satisfied with $j = \ell + 1$. Once that is done, the proof is complete, since we can set (P_{weak}) equal to (P_{infeas}). So let us assume that $j \in \{0, \ldots, \ell\}$ is an integer, and all the invariant conditions hold with j.

We will perform Step j below, which transforms $\mathcal{A}'$ and $X_1, \ldots, X_{\ell+1}$ and we will prove that afterwards the invariant conditions hold with $j+1$. Recall the notation (6.33).

Step j We assume that $A'_1, \ldots, A'_{k+1}$ has structure $\{P_1, \ldots, P_{k+1}\}$ and set $P_{k+2} := N \setminus P_{1:(k+1)}$. We also assume that $X_1, \ldots, X_j$ has structure $Q_1, \ldots, Q_j$.

By condition (**INV-4**) and using an argument like in the proof of the "If" direction in Theorem 1 we see that $\mathcal{S}^n_+ \cap X_1^\perp \cap \cdots \cap X_j^\perp$ is the set of matrices whose rows and columns corresponding to $Q_{1:j}$ are zero. By condition (**INV-3**) we have

$$X_{j+1} \in \left(\mathcal{S}^n_+ \cap X_1^\perp \cap \cdots \cap X_j^\perp\right)^*,$$

hence

$$X_{j+1}(N \setminus Q_{1:j}) \succeq 0. \tag{6.39}$$

Let us define $R_i = P_i \setminus Q_{1:j}$ for $i = 1, \ldots, k+2$. Then $R_1, \ldots, R_{k+2}$ is a partitioning of $N \setminus Q_{1:j}$ and we rewrite (6.39) as

$$X_{j+1}(R_1 \cup \cdots \cup R_{k+2}) \succeq 0. \tag{6.40}$$

So X_{j+1} looks like on Figure 8, where the red submatrix is positive semidefinite.

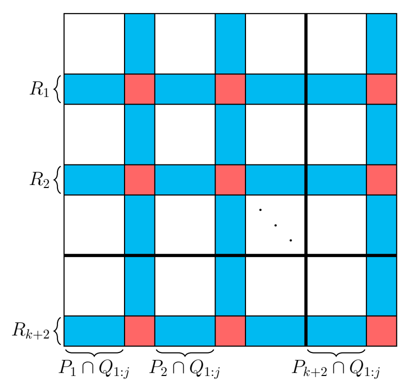


Figure 8: X_{j+1} before it is transformed

We now apply Lemma 7 with $X := X_{j+1}$, $t := k+2$ and the R_i just defined. Let T be the transformation matrix produced by Lemma 7, then

$$T = I_{R_1,U_1} W_1 \ldots I_{R_{k+2},U_{k+2}} W_{k+2}, \tag{6.41}$$

where

(a) the U_i are orthonormal matrices for all i.

(b) right multiplying an $n \times n$ matrix, say M, by W_i adds multiples of columns in $M(N, R_i)$ to columns of $M(N, R_j)$ where $i < j$.

We perform the operations

$$\begin{aligned} X_s &:= T^\top X_s T \quad \text{for } s = 1,\dots,\ell+1, \\ A_i' &:= T^{-1} A_i' T^{-\top} \quad \text{for } i = 1,\dots,m, \end{aligned} \tag{6.42}$$

and set

$$Q_{j+1} := \{ t \mid \text{the } (t,t) \text{ element of } X_{j+1}(N \setminus Q_{1:j}) \text{ is positive} \}.$$

We claim that the invariant conditions now hold for $j+1$. Indeed, (**INV-2**) holds by how we redefined the X_s and A_i' in (6.42). Condition (**INV-3**) holds by Lemma 8.

We next look at (**INV-4**). We first show on Figure 9 how X_{j+1} looks before and after step (6.42). To better see what happened, we permuted the rows and columns of X_{j+1}, so that rows (and columns) indexed by $Q_{1:j}$ come first. The $\oplus$ block stands for a positive semidefinite block.

Step (6.42) transforms $X_{j+1}(N \setminus Q_{1:j})$ to be nonnegative diagonal, and in this process $X_{j+1}(Q_{1:j}, N \setminus Q_{1:j})$ (and the symmetric counterpart) is also transformed. We show the changed block in red in Figure 9.

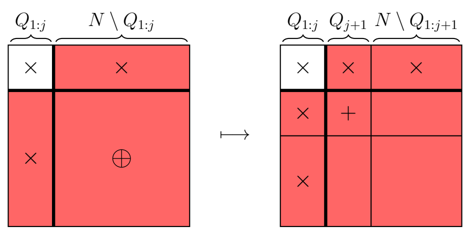


Figure 9: How step (6.42) changes X_{j+1}

Next we look at how $X_1, \dots, X_j$ change due to Step (6.42), so we fix $s \in \{1,\dots,j\}$. Given the factorization of T in (6.41), replacing X_s by $T^\top X_s T$ amounts to running Algorithm 3 below:

Algorithm 3 Transforming X_s

---

for $t = 1 : (k+2)$ **do**
  (*) $X_s := I_{R_t,U_t}^\top X_s I_{R_t,U_t}$;
  (**) $X_s := W_t^\top X_s W_t$;
end for

We claim that after Algorithm 3 is run, the matrix X_s remains in the same shape it was in before. Suppose that this is true after we performed steps (*) and (**) for $t = 1,\dots,q-1$, where $q \geq 1$.

We next perform Step (*) with $t = q$. This amounts to first multiplying $X_s(N, R_q)$ from the right by U_q, then multiplying $X_s(R_q, N)$ from the left by $U_q^\top$. Since $R_1 \cup \dots \cup R_{k+2} = N \setminus Q_{1:j}$, we see that

$$R_q \subseteq N \setminus Q_{1:j} \subseteq N \setminus Q_{1:s}.$$

We depict X_s on Figure 10, with the affected parts shaded in red and conclude that X_s remains in the shape it was in before Step (6.42). Note that on Figure 10 we permuted the rows and columns of X_s so that $X_s(Q_{1:s})$ is in the upper left corner.

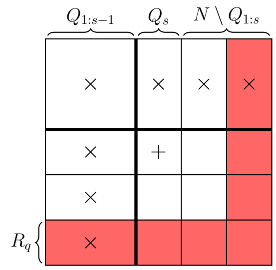


Figure 10: How steps (*) and (**) in Algorithm 3 change X_s, where $s \leq j$.

We next perform Step (**) with $t = q$. Multiplying X_s from the right by W_q adds multiples of columns in $X_s(N, R_q)$ to columns in $X_s(N, R_{q'})$ where $q' \in \{q+1, \ldots, k+2\}$. Then we perform the analogous row operations. Thus Figure 10 again tells us that X_s remains in the same shape.

We thus conclude that condition (**INV-4**) holds after Step (6.42) with $j+1$ instead of j.

We next prove that condition (**INV-1**) remains unchanged after Step (6.42) is executed, so we look at how the A_i' change. Let us fix $i \in \{1, \ldots, m\}$.

Given the decomposition (6.41), we have

$$\begin{aligned} T^{-\top} &= I^{-\top}_{R_1 U_1} W_1^{-\top} \ldots I^{-\top}_{R_{k+2}, U_{k+2}} W^{-\top}_{k+2}, \\ T^{-1} &= W^{-1}_{k+2} I^{-1}_{R_{k+2} U_{k+2}} \ldots W_1^{-1} I^{-1}_{R_1, U_1}. \end{aligned} \tag{6.43}$$

We also know that for all $t \in \{1, \ldots, k+2\}$ by the the definition of $I_{R,G}$ and by $U_t^\top = U_t^{-1}$ the following hold:

$$\begin{aligned} I^{-1}_{R_t, U_t} &= I_{R_t, U_t^\top} \\ I^{-\top}_{R_t, U_t} &= I_{R_t, U_t}. \end{aligned} \tag{6.44}$$

Thus, given (6.43), performing Step (6.42) on A_i' amounts to running Algorithm 4 below.

Algorithm 4 Transforming A_i'

---

for $t = 1 : k+2$ **do**
  (*) $A_i' := I_{R_t, U_t^\top} A_i' I_{R_t, U_t}$;
  (**) $A_i' := W_t^{-1} A_i' W_t^{-\top}$;
end for

We claim that after Algorithm 4 is run, the matrix A_i' remains in the shape it was in before.

Suppose that this is true after we performed steps (*) and (**) for $t = 1, \ldots, q-1$, where $q \geq 1$. Next we perform (*) with $t = q$. We need to keep in mind that $q \in \{1, \ldots, k+2\}$ and $R_q \subseteq P_q$.

We distinguish three cases.

$\mathbf{q = i}$ We show on Figure 11 the A_i' matrix before and after step (*). The changed portion is in red. First the submatrix $A_i'(N, R_q)$ is multiplied from the right by U_q, then the submatrix $A_i'(R_q, N)$ is multiplied from the left by $U_q^\top$.

Thus $A_i'(R_q) = I$ is replaced by $U_q^\top I U_q = I$, so it remains an identity. In summary, A_i' has the same form before and after Step (*).

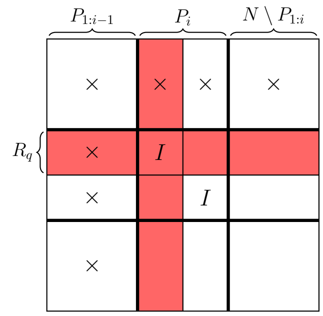


Figure 11: How step (*) in Algorithm 4 changes A'_i, when $q = i$

q < i We show on Figure 12 the A'_i matrix, with the changed portion in red.

First the submatrix $A'_i(N, R_q)$ is multiplied from the right by U_q, then the submatrix $A'_i(R_q, N)$ is multiplied from the left by $U_q^\top$.

Again, A'_i has the same form before and after Step (*).

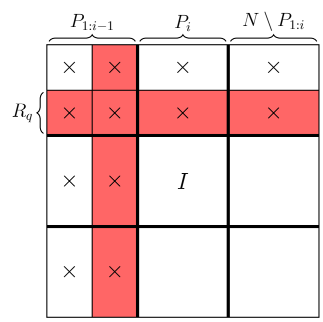


Figure 12: How step (*) in Algorithm 4 changes A'_i, when $q < i$

q > i We show on Figure 13 the A'_i matrix, with the changed portion in red.

First the submatrix $A'_i(N, R_q)$ is multiplied from the right by U_q, then the submatrix $A'_i(R_q, N)$ is multiplied from the left by $U_q^\top$.

Yet again, A'_i has the same form before and after Step (*).

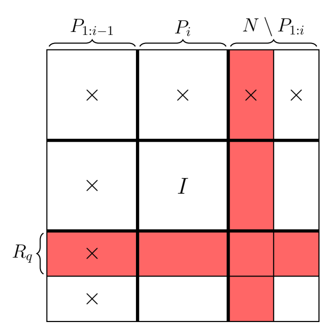


Figure 13: How step (6.42) changes A'_i, when $q > i$

	Miniature	Small	Medium	Large
k	1	3	3	4
ℓ	1	1–3	2–4	2–4
n	3	5–15	25–40	120–240
m	2–5	4–7	4–7	5–8

Table 1: Parameters of our weakly infeasible SDPs

Next we perform Step (**) with $t = q$. We recall that right multiplying A_i' by W_q adds columns of $A_i'(N, R_q)$ to columns in $A_i'(N, R_{q'})$ where $q < q'$. By the algebraic description of W_q (after the statement of Lemma 7) we see that multiplying A_i' from the right by $W_q^{-\top}$ adds columns of $A_i'(N, R_{q'})$ to columns in $A_i'(N, R_q)$ where $q < q'$. (Multiplying A_i' from the left by W_q^{-1} works analogously on the rows of A_i'.)

Recall that $R_q \subseteq P_q$ and $R_{q'} \subseteq P_{q'}$. Thus Figures 11, 12, and 13 tell us that A_i' remains in the same shape as it was before Step (**). Thus condition (**INV-1**) holds, and the proof is complete. $\square$

7 Our problem library and computational tests

Using the results of the previous sections, we now generate a library of weakly infeasible SDPs. We accompany our SDPs with an intuitive visualization and examine whether their infeasibility can be recognized by the prominent SDP solvers MOSEK [1] and SDPA-GMP [5].

Libraries of weakly infeasible SDPs are available [15], the latter library was generated using the Lasserre relaxation of polynomial optimization problems. On the other hand, any weakly infeasible SDP is a possible output of our Algorithm 1, so our current library includes instances that are unlikely to appear in any previous collection.

The instances All 80 instances were generated using Algorithm 1 and are split into two classes: clean and messy.

- Clean instances were constructed by steps 1–3 of Algorithm 1, without using the reformulation step of step 4, so our clean SDPs are in the echelon form (P_{weak}).
- From each clean instance we created a corresponding messy instance as follows. We applied elementary row operations that we represent by an $m \times m$ integral matrix $F = (f_{ij})$, then a congruence transformation that we represent by an $n \times n$ integral matrix T.

  That is, if (P) is a clean instance, then in the corresponding messy instance constraint i is

$$T^\top (\sum_{j=1}^{m} f_{ij} A_j) T \bullet X \; = \; Fb.$$

We further categorize the instances as "miniature", "small", "medium", and "large", with parameters given in Table 1. In each category the clean and messy instances are in one-to-one correspondence. For example, from the "mini, clean, 9" instance we constructed the "mini, messy, 9" instance. In all instances all data is integral, so their weak infeasibility can be verified by hand, in exact arithmetic.

Data storage For convenience we give our instances in three formats.

(1) In the ".mat" files (in Matlab format) the $A_1, \ldots, A_m$ are stored as rows of a matrix A and the $X_1, \ldots, X_{\ell+1}$ are stored as rows of a matrix X. These files also contain the right hand side b.

For each messy instance the files also include the matrices F and T that were used to create it from the corresponding clean instance.

(2) The ".cbf" and ".dat-s" files contain the same SDPs. The ".cbf" files can be directly read by MOSEK and the ".dat-s" files can be directly read by SDPA-GMP.

The ".jpg" files in the "image" subdirectories contain visualizations of the matrices A_i, X_j and F and T for each problem. Matrices $A_1, \ldots, A_{k+1}$ and $X_1, \ldots, X_{\ell+1}$ are color coded, just like in Figure 5.

Computational testing For computational testing we selected the SDP solvers MOSEK and SDPA-GMP as representative industry standards. MOSEK is currently the only commercially available SDP solver, it is fast and accurate on most industrial problems, however, it has limited numerical precision. On the other hand, SDPA-GMP can carry out calculations with precision 10^{-200}.

The results are in Table 2, where we reported a solver's output as "correct" if it marked an instance as infeasible.

	Miniature		Small		Medium/Large	
	Clean	Messy	Clean	Messy	Clean	Messy
MOSEK	0	0	0	0	0	0
SDPA-GMP	10	10	0	2	0	0
Total correct	10	10	10	10	20	20

Table 2: Number of infeasible instances correctly identified by MOSEK and SDPA-GMP

We see that while MOSEK failed to identify infeasibility of any of the SDP instances, SDPA-GMP correctly identified the infeasibility of most of the miniature and of some of the small instances. However, both solvers failed on the the larger instances.

The problem instances are available from the webpage of the first author.

8 Discussion and conclusion

We presented an echelon form of weakly infeasible SDPs that permits us to construct any weakly infeasible SDP and any bad projection of the psd cone by a combinatorial algorithm. Similar normal forms have been available for other types of SDPs and linear maps.

For example, Section 2.2.1 in [23] gave a "good reformulation" of so called well-behaved semidefinite systems of the form $\sum_{i=1}^m x_i A_i \preceq B$. We know that the system $\sum_{i=1}^m x_i A_i \preceq 0$ is well behaved if and only if the set $\mathcal{A}\mathcal{S}_+^n$ is closed, thus setting $B = 0$ we obtain a normal form of linear maps that carry the psd cone to a *closed* set. This normal form actually permits one to construct any such linear map. In contrast, deriving such a normal (echelon) form of weakly infeasible SDPs and of maps that carry $\mathcal{S}_+^n$ to a *nonclosed* set (i.e., bad projections of $\mathcal{S}_+^n$) turned out to be more technical.

To conclude the paper, we reinterpret Theorem 1 in two equivalent forms.

The first interpretation is a kind of "sandwich theorem" which we state in terms of weak infeasibility of $H \cap \mathcal{S}_+^n$ where the affine subspace H is defined in (1.1).

Theorem 4. *The semidefinite program $H \cap \mathcal{S}_+^n$ is weakly infeasible if and only if there are positive integers k and ℓ, an invertible matrix T, and regularized facial reduction sequences $(A_1, \dots, A_{k+1})$ and $(X_1, \dots, X_{\ell+1})$ such that*

$$\begin{aligned} \{ X_{\ell+1} + \sum_{j=1}^{\ell} y_j X_j \} \quad &\subseteq \quad T^\top H T \\ &\subseteq \quad \{X : A_1' \bullet X = \dots = A_k' \bullet X = 0, A_{k+1}' \bullet X = -1\}. \end{aligned}$$

□

Here $T^\top H T$ stands for the set $\{ T^\top X T : X \in H \}$.

The second is a "factorization" theorem. Suppose that (P_{weak}) in Theorem 1 was obtained by elementary row operations that we represent by an $m \times m$ matrix G and by congruence transformations that we represent by an $n \times n$ matrix T, see the discussion in Remark 1. We define the map $\mathcal{T} : \mathcal{S}^n \to \mathcal{S}^n$ as $\mathcal{T}(X) = TXT^\top$ for $X \in \mathcal{S}^n$. Then in Theorem 1 the operators $\mathcal{A}$ and $\mathcal{A}'$ and vectors b and b' are related as

$$\mathcal{A}' = G\mathcal{A}\mathcal{T},\, b' = Gb,$$

so $(\mathcal{A}, b)$ can be "factorized" into $(\mathcal{A}', b')$ (and vice versa). How would we actually compute this factorization? For that, we need to find the Y matrix in Lemma 4, so we must solve SDPs in exact arithmetic. Afterwards we must compute the U_i and W_i transformation matrices in Section 6 in exact arithmetic.

Thus the echelon form (P_{weak}) of weakly infeasible SDPs is in a sense similar to the Jordan echelon form of a matrix, which also must be computed in exact arithmetic [6]. At the same time, both our echelon form, and the Jordan echelon form yield both theoretical and practical insights.

We finally mention an intriguing research question related to sum-of-squares optimization, a thriving research area. As we saw in Example 4 the SDP from minimizing the sum-of-squares (SOS) relaxation of the Motzkin polynomial is weakly infeasible, and is in the echelon form of (P_{weak}) without having to reformulate it. It would be interesting to see whether the same is true of other sum-of-squares SDPs.

Acknowledgement This work was supported by the National Science Foundation, award DMS-1817272.

References

[1] Erling D Andersen and Knud D Andersen. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pages 197–232. Springer, 2000. 27

[2] Heinz H Bauschke, Jonathan M Borwein, and Wu Li. Strong conical hull intersection property, bounded linear regularity, Jameson's property (G), and error bounds in convex optimization. *Mathematical Programming*, 86(1):135–160, 1999. 3

[3] Grigoriy Blekherman. There are significantly more nonnegative polynomials than sums of squares. *Israel Journal of Mathematics*, 153(1):355–380, 2006. 10

[4] Jonathan M. Borwein and Henry Wolkowicz. Regularizing the abstract convex program. *J. Math. Anal. App.*, 83:495–530, 1981. 3, 21

[5] K. Fujisawa, M. Fukuda, M. Kojima, K. Nakata, M. Nakata, and M. Yamashita. SDPA (semidefinite programming algorithm) and SDPA-GMP User's Manual – Version 7.1.0. *Department of Mathematical and Computing Sciences, Tokyo Institute of Technology. Research Reports on Mathematical and Computing Sciences Series B-448*, 2008. 27

[6] Gene H Golub and James Hardy Wilkinson. Ill-conditioned eigensystems and the computation of the jordan canonical form. *SIAM review*, 18(4):578–619, 1976. 29

[7] Jonathan D Hauenstein, Alan C Liddell Jr, and Yi Zhang. Numerical algebraic geometry and semidefinite programming. 2018. 3

[8] Didier Henrion and Jean-Bernard Lasserre. Detecting global optimality and extracting solutions in GloptiPoly. In *Positive polynomials in control*, pages 293–310. Springer, 2005. 3

[9] Graham James Oscar Jameson. The duality of pairs of wedges. *Proceedings of the London Mathematical Society*, 3(3):531–547, 1972. 3

[10] Yuhan Jiang and Bernd Sturmfels. Bad projections of the psd cone. *arXiv preprint arXiv:2006.09956*, 2020. 2, 11

[11] Victor Klee. Asymptotes and projections of convex sets. *Mathematica Scandinavica*, 8(2):356–362, 1961. 3

[12] Jean B Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001. 10

[13] Jean-Bernard Lasserre and Victor Magron. In sdp relaxations, inaccurate solvers do robust optimization. *SIAM Journal on Optimization*, 29(3):2128–2145, 2019. 3

[14] Minghui Liu and Gábor Pataki. Exact duality in semidefinite programming based on elementary reformulations. *SIAM J. Opt.*, 25(3):1441–1454, 2015. 17

[15] Minghui Liu and Gábor Pataki. Exact duals and short certificates of infeasibility and weak infeasibility in conic linear programming. *Mathematical Programming*, pages 1–46, 2017. 3, 4, 6, 16, 17, 21, 27

[16] Yanli Liu, Ernest K Ryu, and Wotao Yin. A new use of Douglas–Rachford splitting for identifying infeasible, unbounded, and pathological conic programs. *Mathematical Programming*, 177(1-2):225–253, 2019. 3

[17] Bruno F. Lourenço, Masakazu Muramatsu, and Takashi Tsuchiya. A structural geometrical analysis of weakly infeasible sdps. *Journal of the Operations Research Society of Japan*, 59(3):241–257, 2016. 3, 17

[18] Theodore Samuel Motzkin. The arithmetic-geometric inequality. *Inequalities (Proc. Sympos. Wright-Patterson Air Force Base, Ohio, 1965)*, pages 205–224, 1967. 10

[19] Pablo A Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical programming*, 96(2):293–320, 2003. 10

[20] Gábor Pataki. A simple derivation of a facial reduction algorithm and extended dual systems. *preprint http://www. unc. edu/˜ pataki/papers/fr. pdf*, 19, 2000. 3

[21] Gábor Pataki. On the closedness of the linear image of a closed convex cone. *Math. Oper. Res.*, 32(2):395–412, 2007. 3

[22] Gábor Pataki. Strong duality in conic linear programming: facial reduction and extended duals. In David Bailey, Heinz H. Bauschke, Frank Garvan, Michel Théra, Jon D. Vanderwerff, and Henry Wolkowicz, editors, *Proceedings of Jonfest: a conference in honour of the 60th birthday of Jon Borwein*. Springer, also available from http://arxiv.org/abs/1301.7717, 2013. 3, 21

[23] Gábor Pataki. Characterizing bad semidefinite programs: normal forms and short proofs. *SIAM Review*, 61(4):839–859, 2019. 2, 12, 28

[24] Javier Peña. Understanding the geometry of infeasible perturbations of a conic linear system. *SIAM Journal on Optimization*, 10(2):534–550, 2000. 3

[25] Javier Peńa and James Renegar. Computing approximate solutions for convex conic systems of constraints. *Mathematical Programming*, 87(3):351–383, 2000. 3, 5

[26] James Renegar. Some perturbation theory for linear programming. *Mathematical Programming*, 65(1-3):73–91, 1994. 3

[27] James Renegar. Linear programming, complexity theory and elementary functional analysis. *Mathematical Programming*, 70(1-3):279–351, 1995. 3

[28] Hayato Waki. How to generate weakly infeasible semidefinite programs via Lasserre's relaxations for polynomial optimization. *Optim. Lett.*, 6(8):1883–1896, 2012. 11