COMPUTATIONAL GEOMETRY OF POSITIVE DEFINITENESS

MARKO HUHTANEN∗ AND OTTO SEISKARI †

Abstract. In matrix computations, such as in factoring matrices, Hermitian and, preferably, positive definite elements are occasionally required. Related problems can often be cast as that of existence of respective elements in a matrix subspace. For two dimensional matrix subspaces, first results in this regard are due to Finsler. To assess positive definiteness in larger dimensional cases, the task becomes computational geometric in a natural way. A Hermitian element of the Frobenius norm one with the maximal least eigenvalue is found. To this end, extreme eigenvalue computations are combined with ellipsoid and perceptron algorithms.

Key words. Hermitian matrix subspace, positive definiteness, least eigenvalue maximization, joint numerical range, computational geometry, convex analysis

AMS subject classifications. 15B48, 47L25

1. Introduction. Appearing in diverse applications, positive definiteness is a central notion for square matrices and operators; see, e.g., [18] and [5]. For related computational matters, see [15, Chapter 4.2]. For matrix subspaces, the concept of positive definiteness is a far more delicate issue. Matrix subspaces with Hermitian and, preferably, positive definite elements arise in factoring problems and in large scale numerical linear algebra of preconditioning [20, 9]. In both cases, the existence of these elements reflects fundamental aspects of operators. The challenge with matrix subspaces lies, not least computationally, in the fact that the subset of positive definite matrices can be a tiny, needle-like set. This paper is concerned with making this computational geometrically more quantitative. Ways to locate positive definite elements are devised. A most positive definite element is found.

Focusing on two dimensional matrix subspaces, first results regarding the existence of positive definite elements are due to Finsler [12]. (For related computations, see [10, 17].) The three dimensional case is related with the investigations of Binding [6]. Later, in semidefinite programming, a similar task defines the feasibility problem of semidefinite programs [27]. Quantitatively, for inclusion regions, we employ strictly positive maps of the simplest possible type. Denote by \( V \) a matrix subspace of \( \mathbb{C}^{n \times n} \) over \( \mathbb{R} \) whose elements are Hermitian. In terms of an orthonormal basis \( V_1, \ldots, V_k \) of \( V \), this leads us to consider the map

\[ x \mapsto (x^* V_1 x, \ldots, x^* V_k x) \quad \text{with} \quad ||x|| = 1 \tag{1.1} \]

whose image is seemingly the most tangible object to study positive definiteness of \( V \). Traditionally, its convexity has been an object of interest; see [16] and references therein. It is noteworthy that the convexity of the image in a basis of \( V \) implies convexity in any of its basis. Thereby we are primarily dealing with a property of the matrix subspace \( V \) rather than that of the map (1.1).

We devise methods to approximate the image of (1.1) with a small number of half-spaces. In this sense the problem becomes computational (not necessarily con-
vex) geometric. In particular, \( \mathcal{V} \) possessing positive definite elements is information involving just a single half-space. This interpretation leads to the notion of most positive definite element of \( \mathcal{V} \). To generate half-spaces, we use the fact that the structure of maps of the form (1.1) is invariant under orthogonal transformations. This combined with eigenvalue computations for the boundary points of the convex hull of the image yields relatively sharp information on the location of the image.

It is a natural task to find the distance of the image of (1.1) from the origin, yielding an orthogonal invariant of \( \mathcal{V} \). We solve the problem for the convex hull of the image. Equivalently, we look for a positive definite element of \( \mathcal{V} \) of the Frobenius norm one having the maximal least eigenvalue. Two algorithms proposed to solve the problem are based on the ellipsoid algorithm used in convex optimization. These methods allow us to locate a most positive definite element in the prescribed sense and compute the distance of the convex hull of the image of (1.1) from the origin. For the easier feasibility problem of locating a positive definite element in \( \mathcal{V} \), the perceptron algorithm is suggested as a simpler alternative.

The paper is organized as follows. In Section 2 fundamentals of Hermitian matrix subspaces are presented, including examples. In Section 3 geometric aspects of locating positive definite elements of a Hermitian matrix subspace are developed. Algorithms to solve the maximal least eigenvalue problem and locating positive definite elements are devised in Section 4. In Section 5 numerical experiments are presented to illustrate the performance of the algorithms. In Appendix A the classical case \( \dim \mathcal{V} = 2 \) and Finsler’s result is covered. Related problems involving positive definite matrices are discussed in Appendix B.

2. Hermitian matrix subspaces and positive definiteness. Assume \( \mathcal{V} \) is a matrix subspace of \( \mathbb{C}^{n \times n} \) over \( \mathbb{C} \) (or \( \mathbb{R} \)). Regarding our interests, so-called nonsingular matrix subspaces are of central relevance [20, 9]. A matrix subspace is said to be nonsingular if it contains invertible elements. Among matrix subspaces, nonsingularity is a generic property [22].

For additional properties, the set of Hermitian matrices \( \mathcal{H} \) is of real dimension \( n^2 \) in \( \mathbb{C}^{n \times n} \). On Hermitian matrix subspaces the standard inner product

\[
(V, W) = \text{tr} \, WV
\]

is used. Here the notion of Hermitian matrix subspace is defined in a natural way as follows.

**Definition 2.1.** A matrix subspace \( \mathcal{V} \) of \( \mathbb{C}^{n \times n} \) over \( \mathbb{R} \) is Hermitian if all its elements are Hermitian.

The Hermitian elements of any matrix subspace \( \mathcal{V} \) of \( \mathbb{C}^{n \times n} \) over \( \mathbb{C} \) (or \( \mathbb{R} \)) can be readily recovered by computing the nullspace of the real linear map

\[
V \mapsto V - V^*
\]

from \( \mathcal{V} \) to \( \mathbb{C}^{n \times n} \). We call this nullspace the Hermitian matrix subspace of \( \mathcal{V} \).

Because of the following fact, Hermitian matrix subspaces are related with a number of classical notions.

**Proposition 2.2.** A matrix subspace \( \mathcal{V} \) of \( \mathbb{C}^{n \times n} \) over \( \mathbb{C} \) is closed under the Hermitian transposition if and only if its Hermitian matrix subspace spans \( \mathcal{V} \).

**Proof.** Since the converse claim is clear, suppose \( \mathcal{V} \) is closed under the Hermitian transposition. For any basis of \( \mathcal{V} \), take the Hermitian and skew-Hermitian parts to have a spanning set for \( \mathcal{V} \). (The Hermitian and skew-Hermitian parts of a matrix \( A \in \mathbb{C}^{n \times n} \) are defined as \( \frac{1}{2}(A + A^*) \) and \( \frac{1}{2i}(A - A^*) \).) \( \square \)
For an obvious example, recall that a $C^*$-algebra is closed under the Hermitian transposition. For its relaxation, an operator system is a matrix subspace over $\mathbb{C}$ which is closed under the Hermitian transposition and contains the identity matrix; see [5] and references therein.

Equivalence is a fundamental operation on matrix subspaces which also can be regarded as a relaxation. Matrix subspaces $\mathcal{V}$ and $\mathcal{W}$ are said to be equivalent if there exist invertible matrices $X,Y \in \mathbb{C}^{n \times n}$ such that $\mathcal{W} = X\mathcal{V}Y^{-1}$. Hermitian structure is preserved in congruence, i.e., when $Y^{-1} = X^*$. For a necessary and sufficient condition on a matrix subspace $\mathcal{V}$ over $\mathbb{R}$ to be equivalent to a Hermitian matrix subspace, suppose $\mathcal{V}_1, \ldots, \mathcal{V}_k$ is its basis. Consider the problem of finding out, whether the matrices $X\mathcal{V}_1Y^{-1}, \ldots, X\mathcal{V}_kY^{-1}$ are Hermitian for some invertible matrices $X$ and $Y$. To solve this, compute the intersection of the nullspaces of the real linear maps

$$M \mapsto V_jM - M^*V_j^*$$

on $\mathbb{C}^{n \times n}$, for $j = 1, \ldots, k$. If there exists an invertible element $M$ in the intersection, then $X$ and $Y$ are determined by the condition $Y^{-1}X^{-*} = M$. With $k = 2$ this arises in the generalized eigenvalue problem.

Denote by $\mathbb{P}_n$ the convex cone of positive definite matrices in $\mathbb{C}^{n \times n}$. (See, e.g., [2, II Sec. 12–15] for the convexity of $\mathbb{P}_n$.) Of course, $\mathbb{P}_n$ and its closure are of tremendous importance in convex optimization, see, e.g., [7].

**DEFINITION 2.3.** A Hermitian matrix subspace $\mathcal{V}$ is said to possess positive definite elements if $\mathcal{V} \cap \mathbb{P}_n \neq \emptyset$.

For a classical two dimensional example, consider the generalized eigenvalue problem. Then it is of central relevance to know if the respective matrix subspace possesses positive definite elements; see [25, Chapter 15.3]. Further examples follow.

**EXAMPLE 1.** Denote by $\mathcal{H}$ the set of Hermitian matrices. An invertible matrix $A \in \mathbb{C}^{n \times n}$ is the product of a Hermitian matrix and a positive definite matrix if and only if the Hermitian subspace of $\mathcal{V} = A^{-1}\mathcal{H}$ contains positive definite elements. (Of course, $A$ is Hermitian if and only if $\mathcal{V}$ contains the identity.) This is a classical notion, such a matrix is said to be symmetrizable [4].

**EXAMPLE 2.** In view of preconditioning large linear systems, assume having an invertible sparse matrix $A \in \mathbb{C}^{n \times n}$. Consider sparse solutions $W \in \mathbb{C}^{n \times n}$ to

$$AW - W^*A^* = 0.$$  

Clearly, there are sparse solutions as $W = A^*$ illustrates. (Leading to the normal equations, this is not attractive in general.) Let $\mathcal{W}$ denote their span. Then one is interested in the positive definite elements of the matrix subspace $AW$ by the fact that with the respective products the conjugate gradient method can be executed.

Observe that Hermitian matrix subspaces possessing no positive definite matrices can be large dimensional. (Consider a matrix subspace with the $(1,1)$-entry equaling zero.)

Whenever nonempty, $\mathcal{V} \cap \mathbb{P}_n$ is an open subset of $\mathcal{V}$ by the fact that if $V \in \mathcal{V} \cap \mathbb{P}_n$, then $V + E \in \mathcal{V} \cap \mathbb{P}_n$ for $E \in \mathcal{V}$ small enough in norm.\(^3\) Hence the convex cone $\mathcal{V} \cap \mathbb{P}_n$ is a submanifold of $\mathcal{V}$ of the same dimension. This is useful, although hardly completely satisfactory information.

\(^3\)For small $n$, to test whether $V \in \mathcal{V} \cap \mathbb{P}_n$, it is advisable to attempt to compute a Cholesky factorization [15, p. 146].
Example 3. The set of diagonal Hermitian matrices in $\mathbb{C}^{3 \times 3}$ is isometrically isomorphic to $\mathbb{R}^3$ in a natural way. The positive definite elements correspond to

$$\{(d_1, d_2, d_3) \in \mathbb{R}^3 : d_j > 0, j = 1, 2, 3\}. \quad (2.5)$$

Let $\mathcal{V}$ be a two dimensional subspace of $\mathbb{R}^3$ (i.e., a plane through the origin) whose intersection with (2.5) is a sharp needle-like set.

For the volume, on the tangent spaces of $\mathcal{V} \cap \mathbb{P}_n$ we employ the standard inner product (2.1). Because the intersection can be a very small set, any purely random process to decide whether $\mathcal{V}$ possesses positive definite elements is highly unlikely to be successful. It is informative, for comparison, to bear in mind that the set of Hermitian matrices is of dimension $n^2$ in $\mathbb{C}^{n \times n}$ of which $\mathbb{P}_n$ occupies just a $\frac{1}{2n}$ portion.

The question of existence of positive semidefinite elements and estimating their volume can be turned, at least in principle, into a problem in real algebraic geometry. For the minimum dimension of the underlying space, denote by $\mathcal{L}$ the set of lower triangular matrices with real diagonal entries, regarded as a subspace of $\mathbb{C}^{n \times n}$ over $\mathbb{R}$ of dimension $n^2$.

**Theorem 2.4.** To the set of positive semidefinite elements of a Hermitian matrix subspace $\mathcal{V} \subset \mathbb{C}^{n \times n}$ corresponds a real homogeneous variety of $\mathcal{L} \subset \mathbb{C}^{n \times n}$.

**Proof.** By the Cholesky factorization, a Hermitian matrix $H$ is positive definite if and only if $H = LL^*$ for a lower triangular matrix with a positive diagonal. Moreover, if $L$ is lower triangular, it readily seen that $LL^*$ positive definite if and only if $L$ has nonzero diagonal entries. Otherwise $LL^*$ is positive semidefinite.

For the construction, with respect to the inner product (2.1), denote by $P$ the orthogonal projector on $\mathcal{H}$ onto $\mathcal{V}$. To characterize the positive semidefinite elements of $\mathcal{V}$, define

$$L \mapsto (I - P)LL^* \quad (2.6)$$

from $\mathcal{L}$ to $\mathcal{H}$. This equals zero if and only if $LL^*$, which is positive semidefinite, belongs to $\mathcal{V}$. Let $M_1, \ldots, M_l$ be an orthonormal basis of the orthogonal complement of $\mathcal{V}$ in $\mathcal{H}$. Then $L$ is mapped to zero by (2.6) if and only if

$$(LL^*, M_j) = 0 \text{ for } j = 1, \ldots, l. \quad (2.7)$$

Since these are homogeneous polynomial maps of degree two in the entries of $L$ separated into real and imaginary parts, to the positive semi-definite elements of $\mathcal{V}$ corresponds a homogeneous variety of $\mathcal{L}$. \(\square\)

There are $n^2 - \dim \mathcal{V}$ equations. (For computational aspects of real algebraic geometry, see [3].) As an extreme, the corresponding variety is the whole $\mathcal{L}$ if and only if $\mathcal{V} = \mathcal{H}$. Involving $n^2$ real variables, solving (2.7) does not appear very realistic unless $n$ is small. On the positive side, though, the degrees of the polynomial equations are just two.

The subspace $\mathcal{L}$ has the advantage that its elements mapped from the variety to the positive definite elements are immediately recovered.

**Corollary 2.5.** $\mathcal{V}$ does not possess positive definite elements if and only if the variety contains only singular elements.

---

4It is somewhat exceptional to use the inner product (2.1) with the manifold of positive definite matrices. For the usual Riemannian geometry of nonpositive curvature, see [5, Chapter 6].
3. Locating positive definite elements geometrically. There are several necessary and sufficient conditions guaranteeing positive definiteness of a Hermitian matrix [18, Chapter 7]. For a Hermitian matrix subspace \( V \), an analogous problem consists of locating positive definite elements, if any. (If \( V \) is not Hermitian, then start by computing its Hermitian matrix subspace.) As just described, with matrix subspaces the challenge lies in the fact that the subset of positive definite matrices can be needle-like.

3.1. Positive definiteness and the joint numerical range. To locate possible positive definite elements for \( k > 2 \), an approach can be based on polynomial inequalities. (For \( k = 2 \), see Appendix A.) To this end, suppose \( V_1, \ldots, V_k \) is a basis of a Hermitian matrix subspace \( V \) and set

\[
V \equiv V(t_1, \ldots, t_k) = t_1V_1 + \cdots + t_kV_k \tag{3.1}
\]

with \( t_j \in \mathbb{R} \) for \( j = 1, \ldots, k \). A Hermitian matrix is positive definite if and only if all its leading principal minors are positive; see, e.g., [18, p. 404].\(^5\) This gives rise to \( n \) polynomial inequalities in the parameters \( t_1, \ldots, t_k \) for determining \( V \cap \mathbb{P}_n \).

Clearly, even for moderate \( n \), dealing with the determinants of large leading principal submatrices is computationally very unappealing. In particular, it certainly may not be the simplest way to inspect the structure of \( V \cap \mathbb{P}_n \).

Regions including \( V \cap \mathbb{P}_n \) can be determined more economically with the help of strictly positive maps. For matrix analysis of positive maps, see [5, Chapter 2] and references therein.

**Definition 3.1.** A linear map \( \Phi : \mathbb{C}^{n \times n} \to \mathbb{C}^{l \times l} \) is strictly positive if \( \Phi(A) \) is positive definite whenever \( A \) is.

For a straightforward example, the linear map on \( \mathbb{C}^{n \times n} \) to any leading principal submatrix is strictly positive.

With (3.1), in terms of a strictly positive linear map \( \Phi \), define

\[
(t_1, \ldots, t_k) \longmapsto -\det \Phi(V(t_1, \ldots, t_k)) \tag{3.2}
\]

This is a homogeneous polynomial of degree \( l \). Here it is natural to choose the basis \( V_1, \ldots, V_k \) to be orthonormal, so that volumes in the parameter space \( (t_1, \ldots, t_k) \in \mathbb{R}^k \) are comparable with volumes in \( V \). This is always assumed in what follows. To have regions including \( V \cap \mathbb{P}_n \), we are interested in those parameter values for which the function (3.2) is positive.

For linear inequalities, inexpensive to generate, consider the strictly positive linear map \( \Phi_x(A) = x^*Ax \) for any fixed \( x \in \mathbb{C}^n \). This inserted into (3.2) gives rise to the open half-space in \( \mathbb{R}^k \) (through the origin) defined as

\[
\sum_{j=1}^k (x^*V_jx)t_j > 0. \tag{3.3}
\]

A finite number of them yields an unbounded convex polytope in the parameter space, as long as there are no contradicting inequalities.

What is the best we can do with a small number of half-spaces? With an orthonormal basis \( V_1, \ldots, V_k \) of \( V \), define

\[
V(x) = (x^*V_1x, \ldots, x^*V_kx). \tag{3.4}
\]

\(^5\) Also called Sylvester’s criterion.
for $x \in S^{n-1} = \{ x \in \mathbb{C}^n : ||x|| = 1 \}$. Clearly, $V$ is smooth. Without any loss of generality, we allow only orthonormal bases of $V$. Then we have orthogonal invariance in the sense that $UV(x)$, where $U \in \mathbb{R}^{k \times k}$ is an orthogonal matrix, is of the same form as (3.4) but in another orthonormal basis of $\mathcal{V}$. And conversely, to any orthonormal basis corresponds such a transformation $U$.

**Example 4.** Let $V_j$ be simultaneously unitarily diagonalizable. (That is, $UV_jU^*$ are diagonal for a unitary matrix $U$, for $j = 1, \ldots, k$.) Then the image of (3.4) is a convex polytope.

The image of $V$ is called the joint numerical range of matrices $V_1, \ldots, V_k$. With respect to the image, a single half-space determines the existence of positive definite elements, if any.

**Proposition 3.2.** A Hermitian matrix subspace $\mathcal{V}$ possesses positive definite elements if and only if the image of (3.4) is contained in an open half-space.

**Proof.** Suppose there is a positive definite linear combination $\sum_{j=1}^k t_j V_j$. Assume $\sum_{j=1}^k t_j^2 = 1$. Then consider $UV(x)$ where $U \in \mathbb{R}^{k \times k}$ is an orthogonal matrix having $(t_1, \ldots, t_k)$ as its first row. This is just (3.4) in another orthonormal basis. By construction, its first component is strictly positive, so that the image is contained in an open half-space.

For the converse, if the image is contained in an open half-space, then (3.3) holds for some $(t_1, \ldots, t_k)$ and for every nonzero $x$. Thereby the linear combination $\sum_{j=1}^k t_j V_j$ is positive definite. $\square$

This yields a way to define the most positive definite element as follows.

**Definition 3.3.** Assume a Hermitian matrix subspace $\mathcal{V}$ possesses positive definite elements. A most positive definite element corresponds to a hyperplane through the origin which is the farthest away from the image of $V$.

By the fact that the image of $V$ is connected, any hyperplane of the definition is outside the convex hull of the image. Hence, the notion is well-defined.

In the case $\dim \mathcal{V} = 2$ we are dealing with the numerical range of a matrix. Admitting many extensions, (3.4) is among them [19, pp. 85–87], being perhaps the most natural one (except that no assumptions on orthonormality are made). Traditionally, its convexity has been an object of interest, leading to the respective notion for matrix subspaces.

**Definition 3.4.** A Hermitian matrix subspace $\mathcal{V}$ is said to have convex numerical range if the image of the map (3.4) is convex.

This is well-defined by the fact that the image of (3.4) being convex in a basis $V_1, \ldots, V_k$ of $\mathcal{V}$ is necessary and sufficient for being convex in any basis of $\mathcal{V}$. This follows from composing $MV(x)$ with any invertible matrix $M \in \mathbb{R}^{k \times k}$ and recovering the corresponding map (3.4).

For convexity results, the case $\dim \mathcal{V} \leq 3$ can be regarded as well-understood [19, p. 86]. An interesting open problem (not considered here) is to identify cases in which $\mathcal{V}$ having convex numerical range is a generic property among Hermitian subspaces of the same dimension in $\mathbb{C}^{n \times n}$.

**Proposition 3.5.** Suppose a Hermitian matrix subspace $\mathcal{V}$ has convex numerical range. Then vanishing of (3.4) at a point is a necessary and sufficient condition on $\mathcal{V}$ not to possess positive definite elements.

To deal with any Hermitian matrix subspace $\mathcal{V}$, set

$$v(\mathcal{V}) = \min_{||x||=1} ||\mathcal{V}(x)||,$$

(3.5)
i.e., the distance of the image of $V$ from the origin. Whether or not $V$ has convex numerical range, this is certainly a quantity of interest. (The Crawford number$^6$ for two, not necessarily orthonormal, Hermitian matrices is defined analogously.) In a way, $v(V)$ yields an opposite of the numerical radius which would correspond to taking the maximum instead. Recall that the numerical radius of a matrix $A \in \mathbb{C}^{n \times n}$ is

$$w(A) = \max_{\lambda \in F(A)} |\lambda|,$$

where $F(A)$ denotes the numerical range of $A$.$^7$ (For its computation, see [29].)

A minimizer yields a good candidate for constructing a positive definite element, yielding an optimal solution in the following case.

**Theorem 3.6.** Suppose a Hermitian matrix subspace $V$ has convex numerical range. If a unit vector $x \in \mathbb{C}^n$ satisfies $v(V) = \|V(x)\| > 0$, then

$$\sum_{j=1}^{k} (x^*V_j x)V_j$$

is the most positive definite element of $V$.

**Proof.** Clearly, $V(x)$ is a boundary point of the image of $V$. Take the half-space $T = \{(t_1, \ldots, t_k) \in \mathbb{R}^k : \sum_{j=1}^{k} (x^*V_j x)t_j < 0\}$. Then, by the convexity assumption, $V(x) + T$ does not intersect the image of $V$. This proves the claim. \[ \square \]

In the next section an algorithm for computing the distance of the convex hull of the image of $V$ from the origin is devised. Thus, $v(V)$ is computable in case $V$ has convex numerical range. Otherwise we obtain a lower bound which still suffices for locating a most positive definite element.

Generating uniformly random points of the image of $V$ seems quite hopeless by the fact that computing values of $V$ at random points of $S^{n-1}$ is not a good idea. The boundary of the image is more accessible. This is due to the fact that for the convex hull of the image we can find support planes by computing extreme eigenvalues and corresponding eigenvectors of Hermitian matrices. Recall that a support plane of a closed set has at least one common point with the set such that the entire set lies in one of the two half-spaces determined by the plane.

**Algorithm 1** Computing a boundary point of the image of $V$.

1: Choose a unit vector $u = (u_1, \ldots, u_k)$ and set $V = \sum_{j=1}^{k} u_j V_j$.
2: Compute an extreme eigenvalue and respective unit eigenvector $x$ of $V$.
3: Set $p = (x^*V_1 x, \ldots, x^*V_k x)$.

Observe that the vector $p$ is on that part of the boundary of the image of $V$ which intersects the boundary of the convex hull of the image of $V$.

In the algorithm, there are two alternatives for the extreme; either the smallest or the largest eigenvalue of $V$. We denote by $\lambda(u)$ the smallest. (Clearly, $\lambda(u) > 0$ if and only if $V$ is positive definite.) In both cases,

$$\{t \in \mathbb{R}^k : \sum_{j=1}^{k} u_j (t_j - p_j) = 0\}$$

$^6$Called the Crawford number of a Hermitian pair.

$^7$As a curiosity, a result by T. Ando states $w(A) < 1$ if and only if the $V \cap \mathbb{P}_n \neq \emptyset$ for a certain Hermitian matrix subspace. See [5, Theorem 3.5.1].
(a) The support plane (solid line) and a boundary point \( p \) (cross) corresponding to direction \( u \) (an unit vector such that \( \lambda(u)u = p^Tuu = \tilde{u} \) is the dashed line).

(b) The boundary lines of the dual cone \( F^* \) (solid lines) and the smallest convex cone containing \( F \) (dashed lines).

**Fig. 3.1.** A \( k = 2 \) dimensional \( F \) (thick outline). The origin is marked with a circle.

yields a support plane of the image of \( V \). It is noteworthy that with the Hermitian Lanczos method, numerical computation of the extreme eigenvalues and corresponding eigenvectors is inexpensive for sparse matrix subspaces.\(^8\) For the Hermitian Lanczos method, see [25]. These are readily programmed, e.g., in MATLAB.

When the unit vector \( u \) is randomly chosen in Algorithm 1, we expect \( V \) to be indefinite, i.e., the hyperplane

\[
\{(t_1, \ldots, t_k) : \sum_{j=1}^{k} u_j t_j = 0\}
\]

intersects the image of \( V \).

### 3.2. Computational geometry for the convex hull of the joint numerical range

Denote by \( F \subset \mathbb{R}^k \) the convex hull of the image of \( V \) and by \( S^{k-1}_\mathbb{R} = \{ u \in \mathbb{R}^k : \|u\| = 1 \} \) the set of unit vectors in \( \mathbb{R}^k \). By executing Algorithm 1, for any \( u \in S^{k-1}_\mathbb{R} \) we can compute \( \lambda(u) \in \mathbb{R} \) and a boundary point \( p \in F \) such that

\[
\lambda(u) = u^T p = \min_{t \in F} u^T t.
\]

A graphical illustration of this is given in Figure 3.1(a). For the compact set \( F \), its dual cone is defined as

\[
F^* = \{ t \in \mathbb{R}^k : p^T t \geq 0 \text{ for all } p \in F \}.
\]

A graphical illustration of a dual cone in two dimensions is given in Figure 3.1(b). Notice that the boundary lines of \( F^* \) are perpendicular to the opposite boundary lines of the smallest convex cone containing \( F \).

For a Hermitian matrix subspace \( \mathcal{V} \), we are interested in solving the minimization problem

\[
\begin{align*}
\max_{\mathcal{V}} & \quad \lambda_{\min}(V) \\
\text{s.t.} & \quad V \in \mathcal{V} \\
& \quad V \succeq 0 \quad \text{(equivalent to } \lambda_{\min}(V) \geq 0) \\
& \quad \|V\|_F = 1,
\end{align*}
\]

\(^8\)A matrix subspace is sparse if its members are sparse with a common sparsity pattern.
where $\lambda_{\min}(V)$ denotes the smallest eigenvalue of a Hermitian matrix $V$. If $V_1, \ldots, V_k$ is an orthonormal basis of $V$, then (3.10) is equivalent to

$$\max \quad \lambda(u)$$

s.t. $u \in F^*$ (equivalent to $\lambda(u) \geq 0$)

$$\|u\| = 1$$

with $V = \sum_{i=1}^{k} u_i V_i$. The strict feasibility problem (3.10) means locating a positive definite matrix in $V$. The strict feasibility problem (3.11) means locating an element $u \in S_{\mathbb{R}}^{k-1}$ satisfying $v^T u > 0$ for all $v \in F$, as given in Definition 3.3 in terms of the corresponding hyperplane. Observe that the latter problem (3.11) can also be seen as a “dual” of the convex optimization problem

$$\min \quad \|p\|$$

s.t. $p \in F$ (3.12)

under the assumption $0 \notin F$.

**Theorem 3.7.** If $p'$ solves (3.12), then $u' = p'/\|p'\|$ solves (3.11) with $\lambda(u') = \|p'\|$.

**Proof.** If $\lambda(u') = p^T u' < \|p'\|$ for some $p \in F$, then there exists a point on the line segment between $p$ and $p'$ closer to the origin than $p'$, which is a contradiction. Therefore $\lambda(u') = \|p'\|$. In addition, for any $u \in S_{\mathbb{R}}^{k-1}$,

$$\lambda(u) = p^T u \leq (p')^T u \leq \|p'\| = \lambda(u'),$$

for some $p \in F$, which proves the claim. \(\square\)

From Theorem 3.7 we can conclude that for any feasible $u \in S_{\mathbb{R}}^{k-1}$ and $p \in F$ holds

$$\lambda(u) \leq \lambda(u') = \|p'\| \leq \|p\|$$

(3.13)

and therefore any pair of primal and dual feasible points $(p, u)$ can be used to bound the optimal values in (3.12) and (3.11).

In the next section we devise methods to so solve the computational geometric problems (3.11) and (3.12) relying, in essence, only on a “least eigenvalue solver”, i.e., using Algorithm 1 we assume that for a given $u \in S_{\mathbb{R}}^{k-1}$ we can generate $\lambda(u)$ and $p \in F$ such that $\lambda(u) = p^T u$.

4. Algorithms. Next algorithms for solving the positive definiteness problems are devised.

4.1. Perceptron algorithm for feasibility. The strict feasibility problem related with (3.11) consists of finding a unit vector $u$ such that $p^T u > 0$ for all $p \in F$, i.e., an unit vector in the interior of the dual cone $F^*$ (assuming $0 \notin F$). It can be solved fairly efficiently using a simple method known as the perceptron algorithm shown in Algorithm 2. The polynomial convergence of this algorithm is stated as follows. The proof is adapted to our setting from [14].

**Theorem 4.1 (Perceptron Convergence Theorem).** Algorithm 2 will converge in at most $\max_{p \in F} \|p\|^2 / \lambda^2(u')$ steps.

**Proof.** Let $u'$ be the solution to (3.11) and $\lambda(u_i) \leq 0$ for $i = 1, \ldots, j$. Then, for any $i = 1, \ldots, j$, holds

$$\|\tilde{u}_{i+1}\|^2 = \|\tilde{u}_i\|^2 + 2\tilde{u}_i^T p_i + \|p_i\|^2 \leq \|\tilde{u}_i\|^2 + \max_{p \in F} \|p\|^2$$

where $\lambda_{\min}(V)$ denotes the smallest eigenvalue of a Hermitian matrix $V$. If $V_1, \ldots, V_k$ is an orthonormal basis of $V$, then (3.10) is equivalent to

$$\max \quad \lambda(u)$$

s.t. $u \in F^*$ (equivalent to $\lambda(u) \geq 0$)

$$\|u\| = 1$$

with $V = \sum_{i=1}^{k} u_i V_i$. The strict feasibility problem (3.10) means locating a positive definite matrix in $V$. The strict feasibility problem (3.11) means locating an element $u \in S_{\mathbb{R}}^{k-1}$ satisfying $v^T u > 0$ for all $v \in F$, as given in Definition 3.3 in terms of the corresponding hyperplane. Observe that the latter problem (3.11) can also be seen as a “dual” of the convex optimization problem

$$\min \quad \|p\|$$

s.t. $p \in F$ (3.12)

under the assumption $0 \notin F$.

**Theorem 3.7.** If $p'$ solves (3.12), then $u' = p'/\|p'\|$ solves (3.11) with $\lambda(u') = \|p'\|$.

**Proof.** If $\lambda(u') = p^T u' < \|p'\|$ for some $p \in F$, then there exists a point on the line segment between $p$ and $p'$ closer to the origin than $p'$, which is a contradiction. Therefore $\lambda(u') = \|p'\|$. In addition, for any $u \in S_{\mathbb{R}}^{k-1}$,

$$\lambda(u) = p^T u \leq (p')^T u \leq \|p'\| = \lambda(u'),$$

for some $p \in F$, which proves the claim. \(\square\)

From Theorem 3.7 we can conclude that for any feasible $u \in S_{\mathbb{R}}^{k-1}$ and $p \in F$ holds

$$\lambda(u) \leq \lambda(u') = \|p'\| \leq \|p\|$$

(3.13)

and therefore any pair of primal and dual feasible points $(p, u)$ can be used to bound the optimal values in (3.12) and (3.11).

In the next section we devise methods to so solve the computational geometric problems (3.11) and (3.12) relying, in essence, only on a “least eigenvalue solver”, i.e., using Algorithm 1 we assume that for a given $u \in S_{\mathbb{R}}^{k-1}$ we can generate $\lambda(u)$ and $p \in F$ such that $\lambda(u) = p^T u$.

4. Algorithms. Next algorithms for solving the positive definiteness problems are devised.

4.1. Perceptron algorithm for feasibility. The strict feasibility problem related with (3.11) consists of finding a unit vector $u$ such that $p^T u > 0$ for all $p \in F$, i.e., an unit vector in the interior of the dual cone $F^*$ (assuming $0 \notin F$). It can be solved fairly efficiently using a simple method known as the perceptron algorithm shown in Algorithm 2. The polynomial convergence of this algorithm is stated as follows. The proof is adapted to our setting from [14].

**Theorem 4.1 (Perceptron Convergence Theorem).** Algorithm 2 will converge in at most $\max_{p \in F} \|p\|^2 / \lambda^2(u')$ steps.

**Proof.** Let $u'$ be the solution to (3.11) and $\lambda(u_i) \leq 0$ for $i = 1, \ldots, j$. Then, for any $i = 1, \ldots, j$, holds

$$\|\tilde{u}_{i+1}\|^2 = \|\tilde{u}_i\|^2 + 2\tilde{u}_i^T p_i + \|p_i\|^2 \leq \|\tilde{u}_i\|^2 + \max_{p \in F} \|p\|^2$$
where possible to construct an updated ellipsoid \(j\) contains the solution, where

\[
\mathcal{E}_j = \{ v \in \mathbb{R}^k : (v - t_j)^T A_j^{-1} (v - t_j) \leq 1 \},
\]

and hence \(\|\tilde{u}_{j+1}\|^2 \leq j \max_{p \in F} \|p\|^2\). On the other hand for any \(i = 1, \ldots, j\) we have

\[
\tilde{u}_{i+1}^T u' = \tilde{u}_i^T u' + p_i^T u' \geq \tilde{u}_i^T u' + \lambda(u')
\]

which yields \(\tilde{u}_{j+1}^T u' \geq j \lambda(u')\). Thereby

\[
\sqrt{j} \max_{p \in F} \|p\| \geq \|\tilde{u}_{j+1}\| \geq \tilde{u}_{j+1}^T u' \geq j \lambda(u'),
\]

that is, \(j \leq \max_{p \in F} \|p\|^2 / \lambda^2(u')\).

4.2. Ellipsoid algorithm. The problem (3.11) is an optimization problem on the \((k-1)\)-sphere \(S_{k-1}^{k-1}\) and thereby not as such a convex optimization problem in \(\mathbb{R}^k\). However, some standard convex optimization techniques may still be applied to the problem. In what follows, an ellipsoid algorithm is devised to solve the task.

To this end, consider an ellipsoid

\[E_j = \mathcal{E}(A_j, t_j) = \{ v \in \mathbb{R}^k : (v - t_j)^T A_j^{-1} (v - t_j) \leq 1 \},\]

where \(A_j \in \mathbb{R}^{k \times k}\) is positive definite and \(t_j \in \mathbb{R}^k\) is the center of \(E_j\). Assume that \(E_j\) contains an optimal point \(u' \in \mathbb{R}^k\) of an optimization problem. If \(c_j \in \mathbb{R}^k\) and \(\beta_j \in \mathbb{R}\) are such that \(c_j^T u' \geq \beta_j\), then also

\[E_j \cap H_{c_j, \beta_j}\]

contains the solution, where \(H_{c_j, \beta_j} = \{ v \in \mathbb{R}^k : c_j^T v \geq \beta_j \}\). Define

\[
\alpha_j = \frac{\beta_j - c_j^T t_j}{\sqrt{c_j^T A_j c_j}}.
\]

Then the half-space \(H_{c_j, \beta_j}\) defines a valid cut of \(E_j\) if \(\alpha_j = 0\) and a valid deep cut if \(0 < \alpha_j \leq 1\). If \(\alpha_j > 1\), the intersection is empty. For any \(1 > \alpha_j > -1/k\), it is possible to construct an updated ellipsoid

\[E_{j+1} = \{ v \in \mathbb{R}^k : (v - t_{j+1})^T A_{j+1}^{-1} (v - t_{j+1}) \leq 1 \},\]

where

\[
t_{j+1} = t_j - \frac{1 + k \alpha_j}{k + 1} b_j, \quad b_j = \frac{A_j c_j}{\sqrt{c_j^T A_j c_j}}, \quad A_{j+1} = \frac{k^2 (1 - \alpha_j^2)}{k^2 - 1} \left( A_j - \frac{2(1 + k \alpha_j)}{(k+1)(1+\alpha_j)} b_j b_j^T \right),
\]

and

\[
\alpha_{j+1} = \frac{\beta_j - c_j^T t_{j+1}}{\sqrt{c_j^T A_{j+1} c_j}}.
\]
such that \( E_{j+1} \supset E_j \cap \Pi_{\alpha_j, \beta_j} \) and the volume of \( E_{j+1} \) is strictly less than that of \( E_j \) [13]. If \( \alpha_j \geq 0 \), then

\[
\text{volume}(E_{j+1}) \leq e^{-\frac{1}{2}} \text{volume}(E_j)
\]

[8]. In particular, if one can find an initial ellipsoid \( E_0 \) such that \( u' \in E_0 \) and for each half-space \( \Pi_{\alpha_j, \beta_j} \) holds \( \alpha_j \geq 0 \), then the solution \( u' \) is contained in a sequence of ellipsoids \( E_j \) whose volume tends geometrically to zero. This procedure is known as the (deep cut) ellipsoid algorithm [13].

For the problem (3.11) an ellipsoid algorithm can be devised as follows. Initially, set \( t_0 = 0 \) and \( A_0 = I \). For any \( j > 0 \), set \( u_j = t_j/\|t_j\| \) if \( \|t_j\| > 0 \). Otherwise, pick an arbitrary \( u_j \in S^{k-1}_R \). As shown later, it can be ensured that \( \|t_j\| \leq 1 \). Let \( \lambda_j^{\text{best}} = \max(\{\lambda(u_i), i = 0, 1, \ldots, j\}) \) (4.3) and \( p_j \in F \) be such that \( \lambda(u_j) = p_j^T u_j \). By setting \( c_j = p_j \), \( \beta_j = \lambda_j^{\text{best}} \) we get

\[
\beta_j - c_j^T t_j = \lambda_j^{\text{best}} - \|t_j\|p_j^T u_j = \lambda_j^{\text{best}} - \|t_j\|\lambda(u_j) \geq 0,
\]

which means that \( \alpha_j \geq 0 \). This corresponds to a valid (possibly deep) cut because

\[
c_j^T u' = p_j^T u' \geq \lambda(u') \geq \lambda_j^{\text{best}} = \beta_j.
\]

One may then construct an ellipsoid \( E(\hat{A}_{j+1}, \hat{t}_{j+1}) \) according to (4.2). If \( \|\hat{t}_{j+1}\| \leq 1 \), set \( t_{j+1} = \hat{t}_{j+1} \) and \( A_{j+1} = \hat{A}_{j+1} \) and continue. Otherwise, the choices \( \hat{c}_j = -\hat{t}_{j+1}/\|\hat{t}_{j+1}\| \) and \( \beta_j = -1 \) correspond to a valid deep cut to \( E(\hat{A}_{j+1}, \hat{t}_{j+1}) \) whose update according to (4.2) yields an ellipsoid \( E(A_{j+1}, t_{j+1}) \) such that \( \|t_{j+1}\| \leq 1 \) (or if this does not hold, the procedure may be repeated as long as it does).

Regarding the relative error of the solution, the error bounds given in [13] cannot be used since the value \( \lambda(u_j) \) is not an evaluation of the objective function at the center \( t_j \) of the ellipsoid. For the problem (3.11), strict error bounds are given by

\[
\lambda_j^{\text{best}} = \lambda(u_j^{\text{best}}) \leq \lambda(u') \leq \lambda_j^{\text{best}} := \min_{0 \leq i \leq j} \|p_i\|. \tag{4.4}
\]

Another upper bound is\(^9\)

\[
\lambda(u') \leq p^T u' = p^T t + p^T (u' - t) \leq p^T t + \sqrt{p^T A p}
\]

for any \( p \in F \) and \( E(A, t) \ni u' \). Therefore set

\[
\lambda_j^{\text{max}} = \min(p_j^T t_j + \sqrt{p_j^T A_j p_j}, \|p_j\|, \lambda_{j-1}^{\text{max}}), \tag{4.5}
\]

where \( \lambda_0^{\text{max}} = \infty \). With these, the resulting ellipsoid algorithm is summarized as Algorithm 3.

\(^9\)If \( v = \arg \max_{v \in E(A, 0)} p^T v \), then \( \nabla_v(p^T v - \mu v^T A^{-1}v) = 0 \iff v = \frac{1}{2\mu} A p \), and from \( vA^{-1}v \leq 1 \) we get \( p^T v = \sqrt{p^T A p} \).
Algorithm 3 Deep-cut ellipsoid algorithm to solve (3.11)

1: Set $j = 0$, $A_0 = I$, $t_0 = 0$, pick $u_1 \in S_{\mathbb{R}}^{k-1}$
2: repeat
3: Increase $j$
4: Compute $p_j \in F$ such that $\lambda(u_j) = p_j^T u_j$.
5: Compute $t_j, A_j$ from (4.1) and (4.2), applying norm cuts if necessary.
6: Update $\lambda_j^{\text{best}}, \lambda_j^{\text{max}}$ and $u_j^{\text{best}}$ according to (4.3), (4.4) and (4.5).
7: Set $u_{j+1} = t_j / \|t_j\|$
8: until $\lambda_j^{\text{max}} - \lambda_j^{\text{best}} < \epsilon$ (or some other stopping criterion is satisfied)
9: return $u_j^{\text{best}}$.

4.3. Accelerated ellipsoid algorithm. The principles described in Section 4.2 can be used to construct other cutting-plane-based methods to solve the problem (3.11). For instance, the ellipsoid method may be sped up by storing multiple points $p_j$ computed so far. Namely, for any $p \in F$ and $j \geq 0$ holds

$$p^T u' \geq \lambda_j^{\text{best}}. \tag{4.6}$$

If $\frac{\lambda_j^{\text{best}} - p^T t_j}{\sqrt{p^T A_j p}} > -1/k$, then (4.6) defines a cut which can be used to decrease the volume of the ellipsoid when updated according to (4.2). In theory, an ellipsoid $E(A, t)$ may be cut during the same iteration until it satisfies

$$\frac{\lambda_j^{\text{best}} - p_i^T t}{\sqrt{p_i^T A p_i}} \leq -1/k, \quad \forall 0 \leq i \leq j \quad \text{and} \quad \frac{||t||^2 - ||t'||^2}{\sqrt{t^T A t}} \leq -1/k.$$ 

It may not be feasible to find such an ellipsoid exactly, but an approximation may be computed by iterating and cutting over all $p_i, i = 0, 1, \ldots, j$ multiple times.

Algorithm 5 describes a relatively straightforward multiple cutting-plane scheme that can be used to speed up the ellipsoid algorithm if the execution time is dominated by the eigenvalue computations. It works as Algorithm 3, except that the ellipsoid is also cut with (at most $M_j$) constraints from the previous eigenvalue computation rounds. As in the initialization phase, the cutting is repeated over multiple $M_j$ rounds. On lines 18–19 the stored constraints are pruned so that only the ones that contributed with the deepest cuts (greatest $\alpha$) remain. The upper bound $\lambda_{\text{max}}$ is updated whenever possible.

A way to further speed up the method is to use additional inequalities (3.3) to construct an initial ellipsoid. For example, a necessary condition for the positive semidefiniteness of a matrix $V \in \mathcal{V}$ is that all its diagonal elements are nonnegative. This yields $n$ initial linear constraints in $\mathbb{R}^k$. Algorithm 4 describes a method to compute the initial ellipsoid $E(A, t)$ for ellipsoid methods (Algorithms 3 and 5) when used to solve problem (3.10). This method performs $M_0$ rounds of cutting the ellipsoid with the “diagonal equations” emerging from the diagonal positivity requirements.

5. Numerical experiments. The difficulty of the problem (3.11) is closely related to $\lambda(u')$, the distance between the origin and $F$, and the size of $F$. With respect to these parameters, we designed easy and challenging experiments. This makes the construction of matrix subspaces $\mathcal{V} \subset \mathbb{C}^{n \times n}$ somewhat involved.

Start with a Hermitian matrix subspace $\mathcal{V}$ spanned by the matrices $\hat{V}_j = \frac{1}{2}(A_j + A_j^*)$, where each $A_j \in \mathbb{C}^{n \times n}$, $j = 1, \ldots, k$ is a random band matrix with normally
Algorithm 4 Initialization scheme for ellipsoid algorithms for the problem (3.10)

1: Set $A \leftarrow I$, $t \leftarrow 0$.
2: for $i = 1, \ldots, n$ do
3:   Define $d_i = [(V_1)_{ii} \ldots (V_k)_{ii}]^T$
4: end for
5: for $j = 1, \ldots, M_0$ do
6:   for $i = 1, \ldots, n$ do
7:     Calculate $\alpha$ for the diagonal equation $d_i^T t \geq 0$ according to (4.1)
8:     Update $A$ and $t$ according to (4.2) if $\alpha > -1/k$
9:   end for
10: Apply a norm cut to $A$ and $t$ if $\|t\| > 1$.
11: end for
12: return $A, t$

Algorithm 5 Accelerated ellipsoid algorithm to solve (3.11)

1: Set $A \leftarrow I$, $t \leftarrow 1$ or calculate them using Algorithm 4
2: Initialize $P \leftarrow []$ with an empty matrix
3: repeat
4:   Set $u \leftarrow t/\|t\|$ if $t \neq 0$ or an arbitrary $u \in S^{k-1}_+$ otherwise
5:   Compute $p \in F$ such that $\lambda(u) = p^T u$, store $P \leftarrow [p P]$
6:   Update $\lambda_{\text{best}}$, $u_{\text{best}}$ and $\lambda_{\text{max}}$ according to (4.3), (4.4), and (4.5).
7:   Let $m$ be the number of columns in $P$.
8:   Initialize $\alpha_i = -1$ for $i = 1, \ldots, m$.
9:   for $j = 1, \ldots, M_1$ do
10:      for $i = 1, \ldots, m$ do
11:         Let $p$ be the $i$th column of $P$
12:         Calculate $\alpha$ for the cutting-plane $p^T t \geq \lambda_{\text{best}}$
13:         If $\alpha > -1/k$, update $A, t$ and $\lambda_{\text{max}}$
14:      end for
15:      Set $\alpha_i \leftarrow \max(\alpha_i, \alpha)$
16: end for
17: Apply norm cut to $A$ and $t$ if $\|t\| > 1$.
18: end for
19: Sort the columns of $P$ to descending order of $\alpha_i$’s
20: Drop all $i$ columns with $\alpha_i < -1/k$, keeping at most $M_2$ columns
21: until $\lambda_{\text{max}} - \lambda_{\text{best}} < \epsilon$ (or some other stopping criterion is satisfied)
22: return $u_{\text{best}}$

distributed complex elements having bandwidth $2j + 1$. Band matrices are used because full random Hermitian matrices were observed to produce $V$ whose range seemed to resemble the $k$-ball. (This we regard as an unfounded bias.) Obviously, matrix subspaces constructed in this way are sparse if $k \ll n$. These matrix subspaces typically cannot be expected to contain positive definite elements (based on numerical experiments). Therefore we translate the basis matrices to have feasible problems.

For any given $b \geq 0$, we construct a Hermitian matrix subspace $V$ such that

$$\min_{p \in F} \|p\| = b. \quad (5.1)$$

First take any Hermitian matrix subspace $\tilde{V}$ with an orthonormal basis $\tilde{V}_1, \ldots, \tilde{V}_k$. 
Choose an arbitrary unit vector $u \in \mathbb{R}^k$ and calculate a boundary point $p$ corresponding to $\lambda(u) = p^T u$ for the convex hull $\tilde{F}$ of the image of $\tilde{V}$. Then form

$$V'_j = \tilde{V}_j + (bu_j - p_j)I$$

and orthonormalize to have $V = \text{span}\{V'_1, \ldots, V'_k\} = \text{span}\{V_1, \ldots, V_k\}$, where the matrices $V_1, \ldots, V_k$ are orthonormal and (5.1) holds.

Based on this construction, for various $n$ and $k$, two types of random problems are generated. The size of the image of $V$ is approximated by $d = \|p' - p\|$, where $p'$ is a boundary point corresponding to $\lambda(-u)$. Using this number $d$, easy problems with $b = \frac{d}{10}$ and challenging problems with $b = \frac{d}{1000}$ are generated.

The performance of Algorithm 2, Algorithm 3 and our accelerated Algorithm 5, initialized with Algorithm 4, are compared. The performances are measured in terms of the number of iterations (or equivalently, the number of eigenvalue computations) required to solve the problems. Each cell is an average over at least ten runs with different random matrix subspaces $V$. As the parameters of Algorithms 4 and 5 we used (quite arbitrarily) $M_0 = 10$, $M_1 = 3$, $M_2 = 50$.

Table 5.1 compares the performance of all three algorithms on the strict feasibility problem, i.e., the problem of locating a positive definite element in a Hermitian subspace. In Algorithms 3 and 5 this is achieved by setting the stopping criterion to $\lambda^\text{best} > 0$. In case $b = 10^{-1}d$, an initialization with Algorithm 4 was, in most cases, enough to have a valid solution and the iteration counts in the “Acc. Ellipsoid” column are therefore close to 1. With Algorithm 2, the iteration counts varied considerably within a class of problems with same parameters (e.g. from 21 to occasionally hundreds with $k = 15, n = 1000, b = 10^{-3}d$). The iteration counts for Algorithms 3 and 5 were more stable in all problems, primarily depending on $b$ and secondarily on $k$.

Table 5.2 compares the performance of Algorithms 3 and 5 on the least eigenvalue maximization problem (3.10). A relative stopping criterion $\frac{\lambda^\text{max} - \lambda^\text{best}}{\lambda^\text{max}} < \epsilon = 10^{-6}$ was used. The problems marked with a dash took too long to solve. The results indicate that the problem (3.10) can be solved reasonably efficiently in matrix subspaces with low dimension $k$, if solving the extremal eigenvalue problems is feasible. Otherwise the iteration count does not seem to have much dependence on $n$. The difficulty of solving the maximization problem seems to primarily depend on the dimension $k$ (and not so much on $b$). The average CPU time needed to solve the problem using Algorithm 5 with $b = 10^{-3}d, k = 15, n = 1000$ on the test workstation\(^\text{10}\) was 24 seconds such that 85% of it was spent inside the least eigenvalue solver routine (MATLAB’s eigs function).

Appendix A: The case $\dim V = 2$. The two dimensional case is instructive, classical and can be solved satisfactorily. For the two dimensional case, discussed in terms of matrix pairs, see [23].

Denote by $F(M)$ the numerical range of a matrix $M \in \mathbb{C}^{n \times n}$.

**Theorem 5.1.** [12, 1] Assume the matrices $V_1$ and $V_2$ span a Hermitian matrix subspace $V$. Then $0 \in F(V_1 + iV_2)$ if and only if $V$ does not possess positive definite elements.\(^\text{11}\)

**Proof.** By a characterization of Finsler [12, Satz 1], there exists a positive definite

\(^{10}\)2.66GHz Intel Core 2 Quad Q8400, 3.2GB RAM, MATLAB R2010b on Debian GNU/Linux

\(^{11}\)In [4, p.76] a related result is called Finsler’s theorem.
Table 5.1
Average number of iterations needed to solve the strict feasibility problem of (3.10)

<table>
<thead>
<tr>
<th>$b$</th>
<th>$k$</th>
<th>$n$</th>
<th>Perceptron</th>
<th>Ellipsoid</th>
<th>Acc. Ellipsoid</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>100</td>
<td>7.4</td>
<td>5.3</td>
<td>1.02</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>1000</td>
<td>16</td>
<td>6.0</td>
<td>1.01</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>5000</td>
<td>26</td>
<td>6.4</td>
<td>1.00</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>100</td>
<td>7.4</td>
<td>6.5</td>
<td>1.12</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>1000</td>
<td>15.8</td>
<td>7.3</td>
<td>1.01</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>5000</td>
<td>34</td>
<td>7.8</td>
<td>1.00</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>100</td>
<td>100</td>
<td>7.0</td>
<td>8.4</td>
<td>2.00</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>100</td>
<td>1000</td>
<td>16.9</td>
<td>7.1</td>
<td>1.4</td>
</tr>
</tbody>
</table>

| $10^{-3}d$ | 5    | 100  | 55         | 23.0      | 8.9           |
| $10^{-3}d$ | 5    | 1000 | 30         | 23.1      | 8.4           |
| $10^{-3}d$ | 5    | 5000 | 56         | 22        | 8             |
| $10^{-3}d$ | 15   | 100  | 221        | 93        | 25            |
| $10^{-3}d$ | 15   | 1000 | 57         | 88        | 24.1          |
| $10^{-3}d$ | 15   | 5000 | 54         | 87        | 24            |
| $10^{-3}d$ | 100  | 100  | 802        | 432       | 44            |
| $10^{-3}d$ | 100  | 1000 | 155        | 369       | 35            |

Table 5.2
Average number of iterations needed to solve (3.10) to relative precision $10^{-6}$

<table>
<thead>
<tr>
<th>$b$</th>
<th>$k$</th>
<th>$n$</th>
<th>Ellipsoid</th>
<th>Acc. Ellipsoid</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>100</td>
<td>202</td>
<td>30.3</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>1000</td>
<td>212</td>
<td>31.2</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>5</td>
<td>5000</td>
<td>216</td>
<td>32</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>100</td>
<td>1467</td>
<td>140</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>1000</td>
<td>1862</td>
<td>148</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>15</td>
<td>5000</td>
<td>-</td>
<td>149</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>100</td>
<td>100</td>
<td>-</td>
<td>6125</td>
</tr>
<tr>
<td>$10^{-1}d$</td>
<td>100</td>
<td>1000</td>
<td>-</td>
<td>5764</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>5</td>
<td>100</td>
<td>266</td>
<td>41.5</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>5</td>
<td>1000</td>
<td>265</td>
<td>40.7</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>5</td>
<td>5000</td>
<td>258</td>
<td>41</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>15</td>
<td>100</td>
<td>2918</td>
<td>168</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>15</td>
<td>1000</td>
<td>2897</td>
<td>166</td>
</tr>
<tr>
<td>$10^{-3}d$</td>
<td>15</td>
<td>5000</td>
<td>-</td>
<td>166</td>
</tr>
</tbody>
</table>

element in $\mathcal{V}$ if and only if

$$x^*V_1x = x^*V_2x = 0, \quad x \in \mathbb{C}^n \implies x = 0.$$  \hfill (5.2)

Namely, the condition (5.2) is clearly necessary for the existence of positive definite linear combinations. For sufficiency, (5.2) is equivalent to the origin not being in the numerical range of the matrix $V = V_1 + iV_2$. This was observed in [1]. The numerical range satisfies $F(e^{-i\gamma}V) = e^{-i\gamma}F(V)$ for any $\gamma \in \mathbb{R}$. The Hermitian part of $e^{-i\gamma}V$ is $\cos(\gamma)V_1 + \sin(\gamma)V_2$ covering projectively all the possible real linear combinations of $V_1$ and $V_2$ when $\gamma$ varies. By the properties of the numerical range, if $0 \not\in F(V_1 + iV_2)$, then there must exist $\gamma \in \mathbb{R}$ such that $\cos(\gamma)V_1 + \sin(\gamma)V_2$ is positive definite [19].}$
The location of the numerical range determines the positive definite linear combinations completely as follows. (Recall that the numerical range is convex.)

**Corollary 5.2.** Let \( \theta_1 \leq \theta_2 \) be the angles of the smallest cone centred at the origin containing \( F(V_1 + iV_2) \) with \( \theta_2 - \theta_1 < \pi \). Then, with \( \gamma = \frac{\theta_2 + \theta_1}{2} \), exactly

\[
\cos(\gamma - \theta)V_1 + \sin(\gamma - \theta)V_2
\]

for \( \theta \in \left( -\frac{1}{2}(\pi + \theta_1 - \theta_2), \frac{1}{2}(\pi + \theta_1 - \theta_2) \right) \) are positive definite.

Assume that \( V_1 \) and \( V_2 \) are orthonormal with respect to the inner product (2.1). The matrices

\[
\cos(\theta_2 - \pi/2)V_1 + \sin(\theta_2 - \pi/2)V_2 \quad \text{and} \quad \cos(\theta_1 + \pi/2)V_1 + \sin(\theta_1 + \pi/2)V_2
\]
determine the boundaries of the cone \( \mathcal{V} \cap \mathbb{P}_n \), so that the angle between them yields its sharpness. This angle is independent of the matrices \( V_1 \) and \( V_2 \) spanning \( \mathcal{V} \), as long as they are orthonormal. (If \( \theta_2 - \theta_1 \approx \pi \), then the set of positive definite elements of \( \mathcal{V} \) is needle-like.) This follows from the use of trigonometric identities and the fact that any other orthonormal basis is obtained as the rows of

\[
\begin{bmatrix}
\cos(\beta) & -\sin(\beta) \\
\sin(\beta) & \cos(\beta)
\end{bmatrix}
\begin{bmatrix}
V_1 \\
V_2
\end{bmatrix}
\]

with \( \beta \in \mathbb{R} \). Observe thus that the structure of \( \mathcal{V} \cap \mathbb{P}_n \) is independent of \( n \).

**Appendix B: Related problems.** Locating a positive definite element in a Hermitian matrix subspace resembles a class of convex optimization problems known as *semidefinite programs*, which can be formulated as

\[
\min \quad \mathbf{c}^T \mathbf{u} \\
\text{s.t.} \quad \mathbf{V}_0 + \sum_{i=1}^{k} u_i \mathbf{V}_i \succeq 0,
\]

where \( \mathbf{V}_0, \ldots, \mathbf{V}_k \) are Hermitian matrices. The feasibility problem of finding a positive semidefinite matrix from an affine Hermitian subspace is also known as a *linear matrix inequality*. Semidefinite programs and linear matrix inequalities for linear subspaces (that is \( \mathbf{V}_0 = 0 \)) are not much of interest since they are either trivially solved by the zero matrix or unbounded. The problem (3.10) is not a special case of a semidefinite program, but an optimization problem on \( \mathbb{S}_k^{-1} \).

The perceptron algorithm is used, in a bit different manner, to locate a positive definite element in Hermitian matrix subspaces by Tong [28]. This subject has also been studied by Zaidi [30].

**REFERENCES**