Plea for a semidefinite optimization solver in complex numbers

Jean Charles Gilbert, Cédric Josz

Plea for a semidefinite optimization solver in complex numbers

J. Ch. Gilbert [†] and C. Josz [‡]

December 27, 2016

Numerical optimization in complex numbers has drawn much less attention than in real numbers. A widespread opinion is that, since a complex number is a pair of real numbers, the best strategy to solve a complex optimization problem is to transform it into real numbers and to solve the latter by a real number solver. This paper defends another point of view and presents three arguments to convince the reader that skipping the transformation phase and using a complex number algorithm, if any, can be much more efficient. This is demonstrated for the particular case of a semidefinite optimization problem solved by a feasible predictor-corrector primal-dual interior-point method. In that case, the complex number formulation has the advantage of (i) having a smaller memory storage, (ii) having a faster iteration, and (iii) requiring less iterations. The computing time saving is rooted in the fact that some operations (like the matrix-matrix product) are much faster for complex operands than for their double size real counterparts, so that the conclusions of this paper could be valid for other problems in which these operations count a great deal in the computing time. The iteration saving has its origin in the smaller (though complex) dimension of the problem in complex numbers.

Keywords: Complex numbers, convergence, optimal power flow, predictor-corrector interior-point algorithm, $\mathbb{R}$-linear and $\mathbb{C}$-linear systems of equations, semidefinite optimization.

AMS MSC 2010: 65E05, 65K05, 90C22, 90C46, 90C51.

Table of contents

[†]INRIA Paris, 2 rue Simone Iff, CS 42112, 75589 Paris Cedex 12, France. E-mail: `Jean-Charles.Gilbert@inria.fr`.

[‡]LAAS, 7, avenue du Colonel Roche, BP 54200, 31031 Toulouse cedex 4, France. E-mail: `Cedric.Josz@gmail.com`.

1 Introduction

Numerical linear algebra in complex numbers has drawn much less attention than in real numbers. For example, the two reference books, those by Golub and Van Loan [9] and by Higham [10], do not even quote the notion of Hermitian matrices in their respective index and discuss very little the numerical methods for complex matrices. The same observation can be made for numerical optimization in complex numbers (see the recent contributions [8, 36, 32, 37, 35, 15, 11] and the references therein for some exceptions). Actually, it is not uncommon to encounter computer scientists asserting that, since a complex number can be viewed as a pair of real numbers, the best strategy to solve a complex number problem is to transform it into real numbers and to run real number algorithms to solve the resulting problem; working directly in complex numbers would therefore be useless folklore. This paper defends another point of view and presents several arguments to convince the reader that developing algorithms in complex numbers can *sometimes* yield methods that are significantly faster than solving their transformed real number counterpart. This claim is often true when the computation involves matrices, which, in the real number version, have their size twice as large as in the complex formulation.

This paper is not at such a high level of generality but demonstrates that a semidefinite optimization (SDO) problem, naturally or possibly defined in complex numbers, should most often also be solved by an SDO solver in complex numbers, if computing time prevails. We are even more specific, since this viewpoint is only theoretically and numerically validated for a primal-dual path-following interior-point algorithm, starting sufficiently close to the central path, whose iteration is a predictor-corrector step based on the Nesterov-Todd direction, and that uses a direct linear system solver; in that case, the speedup is approximately two. Such a claim in such a restrictive domain may sound of marginal relevance, but this acceleration should be observed in many other algorithms, provided some distinctive basic operations (such as those listed in section 5.4) form the bottleneck of the considered numerical method. In the case of the above described interior-point method, the most expensive and frequent basic operation for large problems is the product of matrices. Since, on paper, a speedup of two is reachable for this operation, this benefit naturally and approximately extends to the whole algorithm when it solves sufficiently large problems. This is a "theoretical" claim, based on the evaluation of the number of elementary operations. In practice, the block structure of the memory, the multiple cores of a particular machine, and the decisions taken by the compiler or interpreter may modify, up or down, this speedup estimation. It is therefore necessary to validate this one experimentally. We have done so thanks to a home-made Matlab implementation of the above sketched interior-point algorithm for real/complex SDO problems; the piece of software Sdolab. A comparison with SeDuMi 1.3 is also presented and discussed.

This work has its source in and is motivated by our experience [14, 12] with the alternating current optimal power flow (OPF) problem, which is naturally defined in complex numbers [3, 2, 1]. One of the emblematic instance of the OPF problem consists in minimizing the Joule heating losses in a transmission electricity network, while satisfying the electricity demand by determining appropriately the active powers of the generators installed at given buses (or nodes) of the network. Structurally, one can view the problem as (or reduce to) a nonconvex quadratic optimization problem with nonconvex quadratic constraints in complex numbers [13; § 6]. Recalling that any polynomial optimization problem can be written that way [30, 31, 5], one understands the potential difficulty of such a problem, which is actually NP-hard. In practice, approximate solutions can be obtained by taking the Shor/Lagrangian relaxation of the problem [30, 25, 19, 20, 21] or more and more precise approximate solutions can be computed by memory-greedy moment-sos relaxations [27, 17, 18, 14, 24], as long as the computer accepts the resulting increase in the relaxed problem dimensions. These relaxations may be written like SDO problems in real or complex numbers. Some advantages of the latter formalism have been identified in [15]. As the present paper demonstrates it, having an efficient SDO solver in complex numbers would still increase the appeal of the formulation of the relaxations in complex number.

In the SDO context, the benefit of dealing directly with complex data, instead of transforming the SDO problem into real numbers, was already observed in [33; section 25.3.8.3] (see also the draft [34; section 3.5]), but the present paper considers complex SDO models that are more general and goes further in the analysis of the problem. The primal SDO model in [33] has constraints of the form $\langle A_k, X\rangle = b_k$, where the data A_k and the unknown X are Hermitian matrices, $\langle\cdot,\cdot\rangle$ is the standard scalar product on the vector space of Hermitian matrices, and the given scalar b_k is therefore a real number. In the model of this paper, which occurs in the OPF problem briefly described above, X is still searched as a positive semidefinite Hermitian matrix, but A_k may be an arbitrary complex square matrix (hence b_k is now a complex number). Of course, as we shall see in section 4.2.1, one can transform the latter model into the former, but an SDO solver runs faster if this transformation is not made.

The benefits of solving an SDO problem in complex numbers with a complex interior-point SDO solver instead of its real number analogue can be summarized in three points:

- the approach requires less memory storage (since the matrices have smaller order, see observation 4.4),
- the presented predictor-corrector interior-point solver requires less iterations (observation 5.2),
- each iteration is approximately twice faster in complex numbers (see section 5.4).

These findings alone are sufficient to wish more effort on the development of SDO solvers in complex numbers, which justifies the title of this paper. On the other hand, since it is shown below that the feasible and solution sets are nonempty and bounded simultaneously in the real and complex formulations, there is no clear advantage in using the real or complex formulation of the SDO problem with respect to the well-posedness of a primal, dual, or primal-dual interior-point method (observation 4.10).

The paper is organized as follows. In section 2, we recall some basic concepts in complex analysis. Section 3 presents a rather general form of the complex SDO problem and the way a dual problem can be obtained by Lagrangian duality. Section 4 describes in details how the complex primal and dual SDO problems can be transformed into SDO problems in real numbers, using vector space isometries and ring homomorphisms. Various properties linking the complex SDO problem and its real counterpart are established; in particular, it is shown that both problems have their feasible and solution sets simultaneously bounded. Section 5 deals with algorithmic issues. The predictor-corrector algorithm for solving the complex SDO problem is introduced and it is shown that the generated iterates are not in correspondence with those generated by the same algorithm on the corresponding real version of the problem. In passing, an argument is given to show why the algorithm should require less iterations to converge when it solves the problem in complex number rather than the problem in real numbers. The computation of the complex NT direction is described with some details and it is shown that this computation is well defined under a regularity assumption (the surjectivity of the linear map used to define the affine constraint). This long section ends with a theoretical comparison of the computation efforts of some key operations occurring in complex and real numbers, which explains why an iteration of the complex predictor-corrector algorithm runs faster than in its real twin.

2 Fragments of complex analysis

2.1 Complex numbers and vectors

We denote by $\mathbb{R}$ the set of real numbers and by $\mathbb{C}$ the set of complex numbers. The *pure imaginary number* is written $\mathrm{i} := \sqrt{-1}$. The *real part* of a complex number x is denoted by $\Re(x)$ and its *imaginary part* by $\Im(x)$; hence $x = \Re(x) + \mathrm{i}\Im(x)$. The *conjugate* of x is denoted by $\overline{x} := \Re(x) - \mathrm{i}\Im(x)$ and its *modulus* by $|x| = (\overline{x}x)^{1/2} = (\Re(x)^2 + \Im(x)^2)^{1/2} \in \mathbb{R}$.

We denote by $\mathbb{C}^m$ the $\mathbb{C}$-vector space (hence with scalars in $\mathbb{C}$) formed of the m-uples of complex numbers (sometimes considered as an $m \times 1$ matrix) and by $\mathbb{C}^m_{\mathbb{R}}$ the $\mathbb{R}$-vector space (hence with scalars in $\mathbb{R}$) formed of the same vectors. Recall that the dimension of the $\mathbb{C}$-vector space $\mathbb{C}^m$ is m, while the dimension of the $\mathbb{R}$-vector space $\mathbb{C}^m_{\mathbb{R}}$ is $2m$. The space $\mathbb{C}^m_{\mathbb{R}}$ is useful, since the primal SDO problem (section 3.1) is defined with an $\mathbb{R}$-linear map (i.e., a map that is linear when the scalars are taken in $\mathbb{R}$) whose values are complex vectors. We endow $\mathbb{C}^m_{\mathbb{R}}$ with the scalar product $\langle \cdot, \cdot \rangle_{\mathbb{C}^m_{\mathbb{R}}} : \mathbb{C}^m_{\mathbb{R}} \times \mathbb{C}^m_{\mathbb{R}} \to \mathbb{R}$, defined at $(u,v) \in \mathbb{C}^m_{\mathbb{R}} \times \mathbb{C}^m_{\mathbb{R}}$ by

$$\langle u, v \rangle_{\mathbb{C}^m_{\mathbb{R}}} = \Re(u^{\mathsf{H}} v) = \Re\left(\sum_{i=1}^m \overline{u}_i v_i\right) = \Re(u)^{\mathsf{T}} \Re(v) + \Im(u)^{\mathsf{T}} \Im(v). \tag{2.1}$$

This choice of scalar product will have a direct impact on the form of the adjoint of an operator with values in $\mathbb{C}^m_{\mathbb{R}}$ (see section 3.2.2).

2.2 General complex matrices

The set of real $m \times n$ matrices is denoted by $\mathbb{R}^{m \times n}$ and the set of complex $m \times n$ matrices by $\mathbb{C}^{m \times n}$. We denote by A^{T} the transpose of a matrix $A \in \mathbb{C}^{m \times n}$ (i.e., $(A^{\mathsf{T}})_{ij} = A_{ji}$), by $\overline{A}$ its conjugate (i.e., $(\overline{A})_{ij} = \overline{A_{ij}}$), and by $A^{\mathsf{H}} := (\overline{A})^{\mathsf{T}}$ its conjugate transpose.

We assume that $\mathbb{C}^{m \times n}$ is equipped with the Hermitian scalar product

$$\langle \cdot, \cdot \rangle_{\mathbb{C}^{m \times n}} : (A, B) \in \mathbb{C}^{m \times n} \times \mathbb{C}^{m \times n} \mapsto \langle A, B \rangle_{\mathbb{C}^{m \times n}} := \operatorname{tr} A^{\mathsf{H}} B \in \mathbb{C}, \tag{2.2}$$

making $\mathbb{C}^{m \times n}$ a Hermitian vector space. As any Hermitian scalar product, this one is *left-sesquilinear*, implying that for $\alpha \in \mathbb{C}$: $\langle A, \alpha B \rangle = \alpha \langle A, B \rangle$, while $\langle \alpha A, B \rangle = \overline{\alpha} \langle A, B \rangle$. The associated norm is the Frobenius norm $\| \cdot \|_F : A \in \mathbb{C}^{m \times n} \mapsto \|A\|_F$, where

$$\|A\|_F := \langle A, A \rangle_{\mathbb{C}^{m \times n}}^{1/2} = \left(\operatorname{tr} A^{\mathsf{H}} A\right)^{1/2} = \left(\sum_{ij} |A_{ij}|^2\right)^{1/2}.$$

The Hermitian scalar product (2.2) has the following properties: for A and $B \in \mathbb{C}^{m \times n}$ there hold

$$\langle B, A \rangle_{\mathbb{C}^{m \times n}} = \overline{\langle A, B \rangle_{\mathbb{C}^{m \times n}}} = \langle A^{\mathsf{H}}, B^{\mathsf{H}} \rangle_{\mathbb{C}^{n \times m}}. \tag{2.3}$$

2.3 Hermitian matrices

A matrix $A \in \mathbb{C}^{n \times n}$ is *Hermitian* if $A^{\mathsf{H}} = A$. The set of complex Hermitian matrices is denoted by $\mathcal{H}^n$, while the set of real symmetric matrices is denoted by $\mathcal{S}^n$ and the set of real skew symmetric matrices by $\mathcal{Z}^n$. We also denote by $\mathcal{S}^n_+$ (resp. $\mathcal{S}^n_{++}$) the cone of $\mathcal{S}^n$ formed of the positive semidefinite (resp. definite) matrices. For a Hermitian matrix $A \in \mathcal{H}^n$, there hold $\Re(A) \in \mathcal{S}^n$ and $\Im(A) \in \mathcal{Z}^n$. The set $\mathcal{H}^n$ is an $\mathbb{R}$-vector space but not a $\mathbb{C}$-linear space, since the identity matrix I is Hermitian but not $\mathrm{i}I$ (this observation will have some consequences below and it already implies that all the notions of convex analysis are naturally well defined in $\mathcal{H}^n$). The trace $\operatorname{tr} AB = \sum_{ij} A_{ij} B_{ij}$ of two Hermitian matrices A and B is a real number and the map

$$\langle \cdot, \cdot \rangle_{\mathcal{H}^n} : (A, B) \in \mathcal{H}^n \times \mathcal{H}^n \mapsto \langle A, B \rangle_{\mathcal{H}^n} := \operatorname{tr} AB \in \mathbb{R} \tag{2.4}$$

is a scalar product, making $\mathcal{H}^n$ a Euclidean space.

If $A \in \mathcal{H}^n$ and $v \in \mathbb{C}^n$, then $v^{\mathsf{H}} A v$ is a real number (since its conjugate transpose does not change its value). Therefore, for $A \in \mathcal{H}^n$, it makes sense to say that

$$\begin{aligned} A \text{ is positive semidefinite (notation } A \succcurlyeq 0) &\iff v^{\mathsf{H}} A v \geqslant 0, \quad \forall v \in \mathbb{C}^n, \\ A \text{ is positive definite (notation } A \succ 0) &\iff v^{\mathsf{H}} A v > 0, \quad \forall v \in \mathbb{C}^n \setminus \{0\}. \end{aligned}$$

The set of positive semidefinite (resp. definite) Hermitian matrices is denoted by $\mathcal{H}^n_+$ (resp. $\mathcal{H}^n_{++}$). The set $\mathcal{H}^n_+$ is a closed convex cone of $\mathcal{H}^n$; it is also self-dual, meaning that its dual cone $(\mathcal{H}^n_+)^+ := \{K \in \mathcal{H}^n : \langle K, H \rangle \geqslant 0 \text{ for all } H \in \mathcal{H}^n_+\}$ is equal to itself: $(\mathcal{H}^n_+)^+ = \mathcal{H}^n_+$.

A matrix A is said to be *skew-Hermitian* if $A^{\mathsf{H}} = -A$. Any matrix $A \in \mathbb{C}^{n \times n}$ can be uniquely written as the sum of a Hermitian matrix $\mathcal{H}(A)$ and a skew-Hermitian matrix $\mathcal{Z}(A)$:

$$A = \mathcal{H}(A) + \mathcal{Z}(A), \qquad \text{where } \mathcal{H}(A) := \tfrac{1}{2}(A + A^{\mathsf{H}}) \text{ and } \mathcal{Z}(A) := \tfrac{1}{2}(A - A^{\mathsf{H}}). \tag{2.5}$$

One can therefore write

$$A = \mathcal{H}(A) - \mathrm{i}\,(\mathrm{i}\mathcal{Z}(A)), \tag{2.6}$$

in which $\mathrm{i}\mathcal{Z}(A)$ is also Hermitian. Using this identity and the sesquilinearity of the scalar product $\langle \cdot, \cdot \rangle_{\mathbb{C}^{n \times n}}$, one gets for $A \in \mathbb{C}^{n \times n}$ and $X \in \mathcal{H}^n$:

$$\langle A, X \rangle_{\mathbb{C}^{n \times n}} = \underbrace{\langle \mathcal{H}(A), X \rangle_{\mathcal{H}^n}}_{\text{real number}} + \mathrm{i}\, \underbrace{\langle \mathrm{i}\mathcal{Z}(A), X \rangle_{\mathcal{H}^n}}_{\text{real number}}, \tag{2.7}$$

which provides the decomposition of the complex number $\langle A, X \rangle_{\mathbb{C}^{n \times n}}$ in its real and imaginary parts.

3 The complex SDO problem

Some semidefinite optimization (SDO) problems like the OPF problem [3, 2, 1] are naturally defined in complex numbers, since they deal with the optimization of systems that are more conveniently described in complex numbers. This section introduces the complex (number) SDO problem in a rather general form (section 3.1), as well as its Lagrangian dual (section 3.2), and presents the various forms that can take a standard regularity assumption (proposition 3.2).

3.1 Primal problem

The primal form of the SDO problem consists in finding a Hermitian matrix $X \in \mathcal{H}^n$ minimizing a linear function on a set that is the intersection of the cone $\mathcal{H}^n_+$ of positive semidefinite Hermitian matrices and an affine subspace. It can therefore be written

$$(P) \quad \begin{cases} \inf_X \langle C, X \rangle_{\mathcal{H}^n} \\ \mathcal{A}(X) = b \\ X \succcurlyeq 0, \end{cases} \tag{3.1}$$

where $C \in \mathcal{H}^n$, $\mathcal{A}$ is an $\mathbb{R}$-linear map defined on $\mathcal{H}^n$ with values in

$$\mathbb{F} := \mathbb{F}_r \times \mathbb{F}_c, \qquad \text{where } \mathbb{F}_r := \mathbb{R}^{m_r} \text{ and } \mathbb{F}_c := \mathbb{C}^{m_c}_{\mathbb{R}}, \tag{3.2}$$

and $b \in \mathbb{F}$. Since $\langle C, X\rangle_{\mathcal{H}^n}$ is a real number, the problem has a single objective (to put it another way, the real-valued objective, which is linear on $\mathcal{H}^n$, is represented by the matrix $C \in \mathcal{H}^n$). The linear map $\mathcal{A}$ cannot be $\mathbb{C}$-linear since $\mathcal{H}^n$ is only an $\mathbb{R}$-linear space. Hence its arrival space $\mathbb{F}$ must also be considered as an $\mathbb{R}$-linear space, which is the reason why the complex space $\mathbb{C}^{m_c}_{\mathbb{R}}$ defined in section 2.1 appears in the product space $\mathbb{F}$. The space $\mathbb{F}_r := \mathbb{R}^{m_r}$ is introduced to reflect the fact that some constraints are naturally expressed in real numbers. Of course a real number is just a particular complex number, so that one could have get rid of $\mathbb{F}_r$, but then the surjectivity assumption considered as regularity assumption below (see proposition 3.2) could not be satisfied.

The feasible set of problem (P) is denoted by $\mathcal{F}_P$, its solution set by $\mathcal{S}_P$, and its optimal value by $\operatorname{val}(P)$. We make the notation shorter by introducing the index sets

$$K_r := [1\!:\!m_r] \qquad \text{and} \qquad K_c := [m_r{+}1\!:\!m_r{+}m_c],$$

so that $b \in \mathbb{F}$ can be decomposed in (b_{K_r}, b_{K_c}) with $b_{K_r} \in \mathbb{F}_r$ and $b_{K_c} \in \mathbb{F}_c$. We also introduce the partial $\mathbb{R}$-linear maps

$$(\mathcal{A}_r := \pi_r \circ \mathcal{A}) : \mathcal{H}^n \to \mathbb{F}_r, \qquad \text{and} \qquad (\mathcal{A}_c := \pi_c \circ \mathcal{A}) : \mathcal{H}^n \to \mathbb{F}_c, \tag{3.3}$$

where π_r and π_c are the canonical projectors $\pi_r : b = (b_{K_r}, b_{K_c}) \in \mathbb{F} \mapsto b_{K_r} \in \mathbb{F}_r$ and $\pi_c : b = (b_{K_r}, b_{K_c}) \in \mathbb{F} \mapsto b_{K_c} \in \mathbb{F}_c$.

By the Riesz-Fréchet representation theorem, for $k \in K_r$, the $\mathbb{R}$-linear map $X \in \mathcal{H}^n \mapsto [\mathcal{A}(X)]_k \in \mathbb{R}$ can be represented by a Hermitian matrix, say $A_k \in \mathcal{H}^n$, in the sense that

$$\forall\, X \in \mathcal{H}^n : \quad [\mathcal{A}(X)]_k = \langle A_k, X\rangle_{\mathcal{H}^n}.$$

Similarly, for $k \in K_c$, the $\mathbb{R}$-linear maps $X \in \mathcal{H}^n \mapsto \Re([\mathcal{A}(X)]_k) \in \mathbb{R}$ and $X \in \mathcal{H}^n \mapsto \Im([\mathcal{A}(X)]_k) \in \mathbb{R}$ can be represented by Hermitian matrices, say H^r_k and $H^i_k \in \mathcal{H}^n$:

$$\forall\, X \in \mathcal{H}^n : \quad \Re([\mathcal{A}(X)]_k) = \langle H^r_k, X\rangle_{\mathcal{H}^n} \text{ and } \Im([\mathcal{A}(X)]_k) = \langle H^i_k, X\rangle_{\mathcal{H}^n}.$$

Hence

$$\forall\, X \in \mathcal{H}^n : \quad [\mathcal{A}(X)]_k = \langle H^r_k, X\rangle_{\mathbb{C}^{n\times n}} + \mathrm{i}\langle H^i_k, X\rangle_{\mathbb{C}^{n\times n}} = \langle H^r_k - \mathrm{i}H^i_k, X\rangle_{\mathbb{C}^{n\times n}}.$$

Let us introduce $A_k := H^r_k - \mathrm{i}H^i_k \in \mathbb{C}^{n\times n}$. Although A_k is formed from the Hermitian matrices H^r_k and H^i_k, the decomposition (2.6) shows that it has no particular structure, so that A_k is an arbitrary $n \times n$ complex matrix. In conclusion, we have shown that $\mathcal{A}$ has the following representation

$$[\mathcal{A}(X)]_k = \begin{cases} \langle A_k, X\rangle_{\mathcal{H}^n} & \text{for } k \in K_r \\ \langle A_k, X\rangle_{\mathbb{C}^{n\times n}} & \text{for } k \in K_c, \end{cases} \tag{3.4}$$

where $A_k \in \mathcal{H}^n$ when $k \in K_r$ and $A_k \in \mathbb{C}^{n\times n}$ when $k \in K_c$.

3.2 Lagrangian duality

3.2.1 Dual problem

The Lagrange dual of the complex SDO problem (P) in (3.1) can be obtained like in real numbers. Let $\langle\cdot,\cdot\rangle_{\mathbb{F}}$ be a scalar product on $\mathbb{F}$ (see below). Then problem (P) can be written

$$\inf_{X \succcurlyeq 0} \sup_{y \in \mathbb{F}} \langle C, X\rangle_{\mathcal{H}^n} - \langle y, \mathcal{A}(X) - b\rangle_{\mathbb{F}}.$$

Therefore, its Lagrange dual is the problem

$$\sup_{y\in\mathbb{F}} \inf_{X\succcurlyeq 0} \langle C, X\rangle_{\mathcal{H}^n} - \langle y, \mathcal{A}(X) - b\rangle_{\mathbb{F}} = \sup_{y\in\mathbb{F}} \left(\langle b, y\rangle_{\mathbb{F}} + \inf_{X\succcurlyeq 0} \langle C - \mathcal{A}^*(y), X\rangle_{\mathcal{H}^n} \right).$$

Now, using the self-duality of $\mathcal{H}^n_+$, one gets

$$\inf_{X\succcurlyeq 0} \langle C - \mathcal{A}^*(y), X\rangle_{\mathcal{H}^n} = \begin{cases} 0 & \text{if } C - \mathcal{A}^*(y) \succcurlyeq 0, \\ -\infty & \text{otherwise.} \end{cases}$$

Therefore, the Lagrange dual of (P) is the following problem in $(y, S) \in \mathbb{F} \times \mathcal{H}^n$:

$$(D) \quad \begin{cases} \sup_{(y,S)} \langle b, y\rangle_{\mathbb{F}} \\ \mathcal{A}^*(y) + S = C \\ S \succcurlyeq 0. \end{cases} \tag{3.5}$$

The structure of (D) is similar to the Lagrange dual obtained in the real formulation of the SDO problem.

The feasible set of problem (D) is denoted by $\mathcal{F}_D$, its solution set by $\mathcal{S}_D$, and its optimal value by $\operatorname{val}(D)$.

3.2.2 Adjoint

To write a more specific version of the dual problem (3.5), one using the representation matrices A_k of $\mathcal{A}$ given by (3.4), one must explicit how the adjoint operator $\mathcal{A}^* : \mathbb{F} \to \mathcal{H}^n$ of $\mathcal{A} : \mathcal{H}^n \to \mathbb{F}$ appearing in (D) can expressed in terms of these matrices A_k. This adjoint depends on the scalar product equipping $\mathbb{F}$. We take the standard scalar product of the Cartesian product $\mathbb{F} = \mathbb{F}_r \times \mathbb{F}_c$, namely

$$\langle \cdot, \cdot\rangle_{\mathbb{F}} : (a, b) \in \mathbb{F}^2 \mapsto \langle a, b\rangle_{\mathbb{F}} = \langle a_{K_r}, a_{K_r}\rangle_{\mathbb{F}_r} + \langle a_{K_c}, a_{K_c}\rangle_{\mathbb{F}_c}, \tag{3.6}$$

where $a = (a_{K_r}, a_{K_c})$, $b = (b_{K_r}, b_{K_c})$, $\langle \cdot, \cdot\rangle_{\mathbb{F}_r}$ is the Euclidean scalar product of $\mathbb{F}_r := \mathbb{R}^{m_r}$, and $\langle \cdot, \cdot\rangle_{\mathbb{F}_c}$ is the scalar product of $\mathbb{F}_c := \mathbb{C}^{m_c}_{\mathbb{R}}$ defined in (2.1).

We denote by $\mathcal{A}_i^* : \mathbb{F}_i \to \mathcal{H}^n$, for $i \in \{r, c\}$, the adjoint map of the $\mathbb{R}$-linear map $\mathcal{A}_i : \mathcal{H}^n \to \mathbb{F}_i$ defined in (3.3).

Proposition 3.1 (adjoint of $\mathcal{A}$) *Suppose that $\mathcal{A} : \mathcal{H}^n \to \mathbb{F}$ is defined by (3.4) and that $\mathbb{F}$ is equipped with the scalar product (3.6). Then, for $y = (y_{K_r}, y_{K_c}) \in \mathbb{F}_r \times \mathbb{F}_c = \mathbb{F}$, there holds:*

$$\mathcal{A}^*(y) = \mathcal{A}_r^*(y_{K_r}) + \mathcal{A}_c^*(y_{K_c}), \tag{3.7a}$$

where

$$\mathcal{A}_r^*(y_{K_r}) = \sum_{k\in K_r} y_k A_k, \tag{3.7b}$$

$$\mathcal{A}_c^*(y_{K_c}) = \mathcal{H}\left(\sum_{l\in K_c} y_l A_l \right) = \frac{1}{2} \sum_{l\in K_c} \left(y_l A_l + \overline{y}_l A_l^{\mathsf{H}} \right) \tag{3.7c}$$

$$= \sum_{l\in K_c} \Re(y_l)\mathcal{H}(A_l) + \sum_{l\in K_c} \Im(y_l)(\mathrm{i}\mathcal{Z}(A_l)). \tag{3.7d}$$

PROOF. Take any $y = (y_{K_r}, y_{K_c}) \in \mathbb{F}_r \times \mathbb{F}_c = \mathbb{F}$ and $X \in \mathcal{H}^n$. Formula (3.7a) follows from

$$\begin{aligned}
\langle \mathcal{A}^*(y), X\rangle_{\mathcal{H}^n} &= \langle y, \mathcal{A}(X)\rangle_{\mathbb{F}} \qquad [\text{definition of } \mathcal{A}^*] \\
&= \langle y_{K_r}, \mathcal{A}_r(X)\rangle_{\mathbb{F}_r} + \langle y_{K_c}, \mathcal{A}_c(X)\rangle_{\mathbb{F}_r} \qquad [(3.6) \text{ and } (3.3)] \\
&= \langle \mathcal{A}_r^*(y_{K_r}), X\rangle_{\mathcal{H}^n} + \langle \mathcal{A}_c^*(y_{K_c}), X\rangle_{\mathcal{H}^n}.
\end{aligned}$$

To get (3.7b), just write

$$\begin{aligned}
\langle \mathcal{A}_r^*(y_{K_r}), X\rangle_{\mathcal{H}^n} &= \langle y_{K_r}, \mathcal{A}_r(X)\rangle_{\mathbb{F}_r} \qquad [\text{definition of } \mathcal{A}_r^*] \\
&= \textstyle\sum_{k\in K_r} y_k[\mathcal{A}(X)]_k \qquad [\text{Euclidean scalar product on } \mathbb{F}_r] \\
&= \textstyle\sum_{k\in K_r} y_k\langle A_k, X\rangle_{\mathcal{H}^n} \qquad [(3.4)] \\
&= \langle \textstyle\sum_{k\in K_r} y_k A_k, X\rangle_{\mathcal{H}^n}.
\end{aligned}$$

The first identity in (3.7c) is obtained by

$$\begin{aligned}
\langle \mathcal{A}_c^*(y_{K_c}), X\rangle_{\mathcal{H}^n} &= \langle y_{K_c}, \mathcal{A}_c(X)\rangle_{\mathbb{F}_c} \qquad [\text{definition of } \mathcal{A}_c^*] \\
&= \Re(\textstyle\sum_{l\in K_c} \overline{y}_l[\mathcal{A}(X)]_l) \qquad [(2.1)] \\
&= \Re(\textstyle\sum_{l\in K_c} \overline{y}_l\langle A_l, X\rangle_{\mathbb{C}^{n\times n}}) \qquad [(3.4)] \\
&= \Re(\langle \textstyle\sum_{l\in K_c} y_l A_l, X\rangle_{\mathbb{C}^{n\times n}}) \qquad [(2.2)] \\
&= \langle \mathcal{H}(\textstyle\sum_{l\in K_c} y_l A_l), X\rangle_{\mathcal{H}^n} \qquad [(2.7)].
\end{aligned}$$

The second identity in (3.7c) now comes from the definition (2.5) of the $\mathcal{H}$ operator. Finally, (3.7d) follows immediately from the last expression in (3.7c). □

The surjectivity of the linear map $\mathcal{A}$ is a standard regularity assumption of a real SDO problem. The same assumption intervenes several times below. The next proposition makes explicit what this means in terms of the matrices A_k introduced by (3.4). In this proposition and below, for $\mathbb{K}_i$ being $\mathbb{R}$ or $\mathbb{C}$, we say that a set of complex matrices $\{A_1, \dots, A_m\}$ is $\mathbb{K}_1\times\cdots\times\mathbb{K}_m$-*linearly independent* if any $\alpha \in \mathbb{K}_1 \times \cdots \times \mathbb{K}_m$ satisfying $\sum_{i=1}^m \alpha_i A_i = 0$ vanishes. Of course, matrices that are $\mathbb{K}_1 \times \cdots \times \mathbb{K}_m$-linearly independent are also $\mathbb{K}_1' \times \cdots \times \mathbb{K}_m'$-linearly independent if $\mathbb{K}_i' \subset \mathbb{K}_i$ for all $i \in [1:m]$ (in particular, the $\mathbb{C}^m$-linear independence implies the $\mathbb{R}^m$-linear independence).

Proposition 3.2 (surjectivity of $\mathcal{A}$) *The following properties are equivalent:*
(i) *the $\mathbb{R}$-linear operator $\mathcal{A} : \mathcal{H}^n \to \mathbb{F}$ is surjective,*
(ii) *any $y \in \mathbb{F}$ satisfying $\sum_{k\in K_r} y_k A_k + \mathcal{H}(\sum_{l\in K_c} y_l A_l) = 0$ vanishes,*
(iii) *$\{A_k\}_{k\in K_r} \cup \{A_l\}_{l\in K_c} \cup \{A_l^{\mathsf{H}}\}_{l\in K_c}$ are $(\mathbb{R}^{m_r} \times \mathbb{C}^{2m_c})$-linearly independent,*
(iv) *$\{A_k\}_{k\in K_r} \cup \{\mathcal{H}(A_l)\}_{l\in K_c} \cup \{\mathrm{i}\mathcal{Z}(A_l)\}_{l\in K_c}$ are $\mathbb{R}^{m_r+2m_c}$-linearly independent.*

PROOF. $[(i) \Leftrightarrow (ii)]$ The surjectivity of the $\mathbb{R}$-linear map $\mathcal{A}$ is equivalent to the injectivity of its adjoint $\mathcal{A}^*$, which by (3.7b) and (3.7c) is equivalent to (ii).

$[(ii) \Leftrightarrow (iii)]$ Since the condition $\sum_{k\in K_r} y_k A_k + \mathcal{H}(\sum_{l\in K_c} y_l A_l) = 0$ reads $\sum_{k\in K_r}(2y_k)A_k + \sum_{l\in K_c} y_l A_l + \sum_{l\in K_c} \overline{y}_l A_l^{\mathsf{H}} = 0$, it is clear the $(iii) \Rightarrow (ii)$. Let us now show the converse, assuming that (ii) holds and that

$$\sum_{k\in K_r} y_k A_k + \sum_{l\in K_c} y_l A_l + \sum_{l\in K_c} y_l' A_l^{\mathsf{H}} = 0, \tag{3.8}$$

for some $(y_{K_r}, y_{K_c}, y'_{K_c}) \in \mathbb{R}^{m_r} \times \mathbb{C}^{m_c} \times \mathbb{C}^{m_c}$. Taking the conjugate transpose and adding, one gets

$$\sum_{k\in K_r} (2y_k)A_k + \sum_{l\in K_c} \Big((y_l + \overline{y'_l})A_l + \overline{(y_l + \overline{y'_l})} A_l^{\mathsf{H}} \Big) = 0.$$

By (ii), $y_{K_r} = 0$ and $y_{K_c} + \overline{y'_{K_c}} = 0$. Then (3.8) yields

$$\sum_{l\in K_c} \Big(y_l A_l - \overline{y_l} A_l^{\mathsf{H}} \Big) = 0.$$

Multiplying by i, one gets

$$\sum_{l\in K_c} \Big((\mathrm{i}y_l)A_l + \overline{(\mathrm{i}y_l)} A_l^{\mathsf{H}} \Big) = 0.$$

By (ii) again, $y_{K_c} = 0$. Hence $y'_{K_c} = 0$ and $y = 0$.

[$(ii) \Leftrightarrow (iv)$] By (3.7d), the condition $\mathcal{A}^*(y) \coloneqq \sum_{k\in K_r} y_k A_k + \mathcal{H}(\sum_{l\in K_c} y_l A_l) = 0$ can be written

$$\sum_{k\in K_r} y_k A_k + \sum_{l\in K_c} \Re(y_l)\mathcal{H}(A_l) + \sum_{l\in K_c} \Im(y_l)(\mathrm{i}\mathcal{Z}(A_l)) = 0.$$

Since $\Re(y_l)$ and $\Im(y_l)$ are arbitrary real numbers, the equivalence follows. □

Remarks 3.3 1) The surjectivity of $\mathcal{A}$ implies a constraint on the number of affine constraints: since the $\mathbb{R}$-dimension of $\mathcal{H}^n$ is $n + 2((n-1)n/2) = n^2$ (n real numbers on the diagonal and $(n-1)n/2$ complex numbers on the strictly lower triangular part) and the one of $\mathbb{F}$ is $m_r + 2m_c$, there must hold

$$m_r + 2m_c \leqslant n^2. \tag{3.9}$$

2) Property (iv) is equivalent to saying that the constraint matrices of problem (4.9) below, obtained by transforming the constraints of problem (3.1) into real valued constraints are $\mathbb{R}$-linearly independent. Equivalently, the operator $\mathcal{A}$ is surjective if and only if the operator $\tilde{\mathcal{A}}$ defined in (4.15) below is surjective. □

4 Complex and real SDO formulations

Since a complex number is a pair of real numbers, the complex SDO problem (3.1) can certainly be transformed into real SDO problem. One of the goals of this paper is to compare the complex and real number SDO formulations and to highlight the advantages of the former when the problem is naturally raised in complex numbers. Before showing this, we have to be precise on the way such a complex number SDO problem is transformed into real numbers. This subject is examined in this section. Section 4.1 presents the operators that make possible to switch between the two formulations, as well as some of their properties. Section 4.2 provides one possible way of transforming the complex SDO problem into real numbers and presents a few properties linking the two formulations.

4.1 Switching between complex and real formulations

The transformation of the complex SDO problem of section 3.1 into a real SDO problem in section 4.2 is better done using operators establishing a correspondence between complex objects (vector and matrices) and their real counterparts. This section introduces these operators. These have the remarkable properties of being isometries or homomorphisms. The isometries of section 4.1.1 are more natural for transforming the complex SDO problem into real numbers and, as a result, for making correspondences between the feasible and solution sets, as well as the optimal values, which are identical. The homomorphisms of section 4.1.2 are more appropriate for establishing comparisons between the complex objects generated by the interior-point solvers in the complex and real spaces, presented in section 5.

4.1.1 Vector space isometries

One of the first questions that should be examined deals with the appropriate way of transforming into real numbers the condition $M \succcurlyeq 0$ appearing in (3.1) for a Hermitian matrix $M \in \mathcal{H}^n$. This condition can be written $v^{\mathsf{H}} M v \geqslant 0$ for all $v \in \mathbb{C}^n$ or

$$(\Re(v) - \mathrm{i}\Im(v))^{\mathsf{T}}(\Re(M) + \mathrm{i}\Im(M))(\Re(v) + \mathrm{i}\Im(v)) \geqslant 0, \qquad \text{for all } v \in \mathbb{C}^n. \tag{4.1}$$

Without surprise, this suggests us to transform a complex vector $v \in \mathbb{C}^n$ into a real vector $(\Re(v), \Im(v))$ made of its real and imaginary parts. The associated transformation operator is $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}} : \mathbb{C}^n_{\mathbb{R}} \to \mathbb{R}^{2n}$ defined at $v \in \mathbb{C}^n_{\mathbb{R}}$ by

$$\boxed{\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) = \begin{pmatrix} \Re(v) \\ \Im(v) \end{pmatrix}.}$$

We prefer defining $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}$ on $\mathbb{C}^n_{\mathbb{R}}$ instead of $\mathbb{C}^n$ since it is an $\mathbb{R}$-linear map and, with the scalar product (2.1) on $\mathbb{C}^n_{\mathbb{R}}$ and the Euclidean scalar product on $\mathbb{R}^{2n}$, $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}$ is an isometry:

$$\forall\, v, w \in \mathbb{C}^n_{\mathbb{R}} : \qquad \langle \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v), \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(w) \rangle_{\mathbb{R}^{2n}} = \langle v, w \rangle_{\mathbb{C}^n_{\mathbb{R}}}.$$

Note that, if the size of $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) \in \mathbb{R}^{2n}$ is twice that of $v \in \mathbb{C}^n_{\mathbb{R}}$, the storage of these two vectors requires the same amount of memory.

A straightforward computation shows that (4.1) can be written

$$\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)^{\mathsf{T}} \begin{pmatrix} \Re(M) & -\Im(M) \\ \Im(M) & \Re(M) \end{pmatrix} \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) \geqslant 0, \qquad \text{for all } v \in \mathbb{C}^n.$$

It is therefore natural [33, 15] to introduce as matrix transformation operator the $\mathbb{R}$-linear map $\mathfrak{J}_{\mathbb{C}^{m\times n}} : \mathbb{C}^{m\times n} \to \mathbb{R}^{(2m)\times(2n)}$ defined at $M \in \mathbb{C}^{m\times n}$ by

$$\boxed{\mathfrak{J}_{\mathbb{C}^{m\times n}}(M) = \frac{1}{\sqrt{2}} \begin{pmatrix} \Re(M) & -\Im(M) \\ \Im(M) & \Re(M) \end{pmatrix}.}$$

Here and below, the index of the various operators $\mathfrak{J}$ refers to its starting space. The presence of the factor $1/\sqrt{2}$ is motivated by the fact that $\mathfrak{J}_{\mathbb{C}^{m\times n}}$ satisfies then the isometric-like identity

$$\forall\, M, N \in \mathbb{C}^{m\times n} : \qquad \langle \mathfrak{J}_{\mathbb{C}^{m\times n}}(M), \mathfrak{J}_{\mathbb{C}^{m\times n}}(N) \rangle_{\mathbb{R}^{(2m)\times(2n)}} = \Re(\langle M, N \rangle_{\mathbb{C}^{m\times n}}),$$

where we have used the standard scalar product $\langle A,B\rangle_{\mathbb{R}^{m\times n}} = \operatorname{tr} A^{\mathsf{T}}B$ on $\mathbb{R}^{m\times n}$. Note also that

$$\forall M \in \mathbb{C}^{m\times n}: \quad \mathfrak{J}_{\mathbb{C}^{n\times m}}(M^{\mathsf{H}}) = \mathfrak{J}_{\mathbb{C}^{m\times n}}(M)^{\mathsf{T}}. \tag{4.2}$$

We see that the restriction of $\mathfrak{J}_{\mathbb{C}^{m\times n}}$ to $\mathcal{H}^n$ has values in $\mathcal{S}^{2n}$, since $\mathfrak{R}(M) \in \mathcal{S}^n$ and $\mathfrak{I}(M) \in \mathcal{Z}^n$ when $M \in \mathcal{H}^n$. This restriction is denoted by $\mathfrak{J}_{\mathcal{H}^n} : \mathcal{H}^n \to \mathcal{S}^{2n}$. It satisfies the isometry property

$$\forall\, M,N \in \mathcal{H}^n: \qquad \langle \mathfrak{J}_{\mathcal{H}^n}(M), \mathfrak{J}_{\mathcal{H}^n}(N)\rangle_{\mathcal{S}^{2n}} = \langle M,N\rangle_{\mathcal{H}^n}, \tag{4.3}$$

which shows that the adjoint $\mathfrak{J}_{\mathcal{H}^n}^*$ of $\mathfrak{J}_{\mathcal{H}^n}$ is a left inverse of $\mathfrak{J}_{\mathcal{H}^n}$:

$$\mathfrak{J}_{\mathcal{H}^n}^*\mathfrak{J}_{\mathcal{H}^n} = I_{\mathcal{H}^n}. \tag{4.4}$$

In particular, $\mathfrak{J}_{\mathcal{H}^n}$ is injective. It is routing computation to verify that

$$\mathfrak{J}_{\mathcal{H}^n}^* \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} = \frac{1}{\sqrt{2}}(S_{11}+S_{22}) + \frac{\mathrm{i}}{\sqrt{2}}(S_{21}-S_{12}), \tag{4.5}$$

in which we have assumed that S_{11}, $S_{22} \in \mathcal{S}^n$ and $S_{12}^{\mathsf{T}} = S_{21}$.

4.1.2 Ring homomorphism

As observed in [15], the operator $\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}} := \sqrt{2}\,\mathfrak{J}_{\mathbb{C}^{m\times n}}$ having at $M \in \mathbb{C}^{m\times n}$ the value

$$\boxed{\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M) = \begin{pmatrix} \mathfrak{R}(M) & -\mathfrak{I}(M) \\ \mathfrak{I}(M) & \mathfrak{R}(M) \end{pmatrix}}$$

is a ring homomorphism, since it satisfies $\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M_1+M_2) = \hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M_1)+\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M_2)$, $\hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(I) = I$, and the property in point 1 of the next proposition. This last powerful property and its consequences will allow us to compare the interior-point algorithms in the complex and real spaces. We give its five line proof, to be comprehensive and because it plays a crucial role in the sequel.

Proposition 4.1 ($\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}$ is a ring homomorphism) 1) *For any $M \in \mathbb{C}^{m\times n}$ and $N \in \mathbb{C}^{n\times p}$, there holds $\hat{\mathfrak{J}}_{\mathbb{C}^{m\times p}}(MN) = \hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M)\,\hat{\mathfrak{J}}_{\mathbb{C}^{n\times p}}(N)$.*
2) *If $M \in \mathbb{C}^{n\times n}$ is nonsingular, then $\hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(M)$ is nonsingular and $\hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(M^{-1}) = \hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(M)^{-1}$.*

Proof. 1) Let $M = M_{\mathrm{r}} + \mathrm{i}M_{\mathrm{i}}$ and $N = N_{\mathrm{r}} + \mathrm{i}N_{\mathrm{i}}$, with M_{r} and $M_{\mathrm{i}} \in \mathbb{R}^{m\times n}$, and N_{r} and $N_{\mathrm{i}} \in \mathbb{R}^{n\times p}$. There holds $MN = (M_{\mathrm{r}}N_{\mathrm{r}} - M_{\mathrm{i}}N_{\mathrm{i}}) + \mathrm{i}(M_{\mathrm{r}}N_{\mathrm{i}} + M_{\mathrm{i}}N_{\mathrm{r}})$ and therefore

$$\hat{\mathfrak{J}}_{\mathbb{C}^{m\times p}}(MN) = \begin{pmatrix} M_{\mathrm{r}}N_{\mathrm{r}} - M_{\mathrm{i}}N_{\mathrm{i}} & -M_{\mathrm{r}}N_{\mathrm{i}} - M_{\mathrm{i}}N_{\mathrm{r}} \\ M_{\mathrm{r}}N_{\mathrm{i}} + M_{\mathrm{i}}N_{\mathrm{r}} & M_{\mathrm{r}}N_{\mathrm{r}} - M_{\mathrm{i}}N_{\mathrm{i}} \end{pmatrix},$$

which has the same value as

$$\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M)\,\hat{\mathfrak{J}}_{\mathbb{C}^{n\times p}}(N) = \begin{pmatrix} M_{\mathrm{r}} & -M_{\mathrm{i}} \\ M_{\mathrm{i}} & M_{\mathrm{r}} \end{pmatrix}\begin{pmatrix} N_{\mathrm{r}} & -N_{\mathrm{i}} \\ N_{\mathrm{i}} & N_{\mathrm{r}} \end{pmatrix} = \begin{pmatrix} M_{\mathrm{r}}N_{\mathrm{r}} - M_{\mathrm{i}}N_{\mathrm{i}} & -M_{\mathrm{r}}N_{\mathrm{i}} - M_{\mathrm{i}}N_{\mathrm{r}} \\ M_{\mathrm{r}}N_{\mathrm{i}} + M_{\mathrm{i}}N_{\mathrm{r}} & M_{\mathrm{r}}N_{\mathrm{r}} - M_{\mathrm{i}}N_{\mathrm{i}} \end{pmatrix}.$$

2) Applying point 1 to the identity $MM^{-1} = I$ yields $\hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(M)\,\hat{\mathfrak{J}}_{\mathbb{C}^{n\times n}}(M^{-1}) = I_{2n}$, hence the result. □

Note that

$$\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) = \hat{\mathfrak{J}}_{\mathbb{C}^{n\times 1}}(v)\, e^1, \tag{4.6}$$

where $e^1 := (1\ \ 0)^{\mathsf{T}}$ is the first basis vector of $\mathbb{R}^2$. Therefore, for $M \in \mathbb{C}^{m\times n}$ and $v \in \mathbb{C}^n$:

$$\mathfrak{J}_{\mathbb{C}^m_{\mathbb{R}}}(Mv) = \Big(\hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M)\Big)\Big(\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)\Big), \tag{4.7}$$

since by (4.6) and point 1 of the previous proposition, there holds $\mathfrak{J}_{\mathbb{C}^m_{\mathbb{R}}}(Mv) = \hat{\mathfrak{J}}_{\mathbb{C}^{m\times 1}}(Mv)\, e^1 = \hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M)\hat{\mathfrak{J}}_{\mathbb{C}^{n\times 1}}(v)\, e^1 = \hat{\mathfrak{J}}_{\mathbb{C}^{m\times n}}(M)\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)$.

Proposition 4.2 ($\hat{\mathfrak{J}}_{\mathcal{H}^n}$ on a psd matrix) *Let $M \in \mathcal{H}^n$. Then*
1) *$M \succcurlyeq 0$ if and only if $\hat{\mathfrak{J}}_{\mathcal{H}^n}(M) \succcurlyeq 0$, or equivalently $\hat{\mathfrak{J}}_{\mathcal{H}^n}(\mathcal{H}^n_+) = \mathcal{S}^{2n}_+ \cap \mathcal{R}(\hat{\mathfrak{J}}_{\mathcal{H}^n})$,*
2) *$M \succ 0$ if and only if $\hat{\mathfrak{J}}_{\mathcal{H}^n}(M) \succ 0$, or equivalently $\hat{\mathfrak{J}}_{\mathcal{H}^n}(\mathcal{H}^n_{++}) = \mathcal{S}^{2n}_{++} \cap \mathcal{R}(\hat{\mathfrak{J}}_{\mathcal{H}^n})$,*
3) *if $M \succcurlyeq 0$, then $\hat{\mathfrak{J}}_{\mathcal{H}^n}(M^{1/2}) = \hat{\mathfrak{J}}_{\mathcal{H}^n}(M)^{1/2}$.*

Proof. Using point 1 of proposition 4.1, as well as (4.2) and (4.6), we have for a vector $v \in \mathbb{C}^n$:

$$\begin{aligned} v^{\mathsf{H}} M v &= (e^1)^{\mathsf{T}} \hat{\mathfrak{J}}_{\mathbb{C}^{1\times 1}}(v^{\mathsf{H}} M v) e^1 \\ &= (e^1)^{\mathsf{T}} \hat{\mathfrak{J}}_{\mathbb{C}^{n\times 1}}(v)^{\mathsf{T}}\, \hat{\mathfrak{J}}_{\mathcal{H}^n}(M)\, \hat{\mathfrak{J}}_{\mathbb{C}^{n\times 1}}(v) e^1 \\ &= \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)^{\mathsf{T}}\, \hat{\mathfrak{J}}_{\mathcal{H}^n}(M)\, \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v). \end{aligned}$$

Since $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)$ can represent an arbitrary vector in $\mathbb{R}^{2n}$, and $v \neq 0$ if and only if $\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) \neq 0$, the first parts of points 1 and 2 follow.

The second parts of points 1 and 2 follow directly from the first parts. The equivalence of the two parts also results from the injectivity of $\hat{\mathfrak{J}}_{\mathcal{H}^n}$.

Consider point 3. If $M \succcurlyeq 0$, it has a unique positive semi-definite square root in $\mathcal{H}^n_+$ [9; § 4.2.10], denoted $M^{1/2} \succcurlyeq 0$, and $\hat{\mathfrak{J}}_{\mathcal{H}^n}(M^{1/2}) \succcurlyeq 0$ by point 2. Furthermore

$$\hat{\mathfrak{J}}_{\mathcal{H}^n}(M^{1/2})\, \hat{\mathfrak{J}}_{\mathcal{H}^n}(M^{1/2}) = \hat{\mathfrak{J}}_{\mathcal{H}^n}(M),$$

point 1 of proposition 4.1. Hence $\hat{\mathfrak{J}}_{\mathcal{H}^n}(M^{1/2}) = \hat{\mathfrak{J}}_{\mathcal{H}^n}(M)^{1/2}$ (uniqueness of the square root in $\mathcal{S}^{2n}_+$). □

Proposition 4.3 ($\hat{\mathfrak{J}}^*_{\mathcal{H}^n}$ on a psd matrix) *$\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\mathcal{S}^{2n}_+) = \mathcal{H}^n_+$ and $\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\mathcal{S}^{2n}_{++}) = \mathcal{H}^n_{++}$.*

Proof. Consider the first identity. [$\subset$] Let $\tilde{M} \in \mathcal{S}^{2n}_+$ and $v \in \mathbb{C}^n$. First observe that with $v_{\mathrm{r}} := \Re(v)$ and $v_{\mathrm{i}} = \Im(v)$, one gets

$$\hat{\mathfrak{J}}_{\mathcal{H}^n}(vv^{\mathsf{H}}) = \begin{pmatrix} v_{\mathrm{r}}v_{\mathrm{r}}^{\mathsf{T}} + v_{\mathrm{i}}v_{\mathrm{i}} & v_{\mathrm{r}}v_{\mathrm{i}}^{\mathsf{T}} - v_{\mathrm{i}}v_{\mathrm{r}}^{\mathsf{T}} \\ v_{\mathrm{i}}v_{\mathrm{r}}^{\mathsf{T}} - v_{\mathrm{r}}v_{\mathrm{i}}^{\mathsf{T}} & v_{\mathrm{r}}v_{\mathrm{r}}^{\mathsf{T}} + v_{\mathrm{i}}v_{\mathrm{i}} \end{pmatrix} = \begin{pmatrix} v_{\mathrm{r}} \\ v_{\mathrm{i}} \end{pmatrix}\begin{pmatrix} v_{\mathrm{r}} \\ v_{\mathrm{i}} \end{pmatrix}^{\mathsf{T}} + \begin{pmatrix} -v_{\mathrm{i}} \\ v_{\mathrm{r}} \end{pmatrix}\begin{pmatrix} -v_{\mathrm{i}} \\ v_{\mathrm{r}} \end{pmatrix}^{\mathsf{T}}.$$

Therefore

$$v^{\mathsf{H}}\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\tilde{M})v = \langle \hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\tilde{M}), vv^{\mathsf{H}} \rangle = \langle \tilde{M}, \hat{\mathfrak{J}}_{\mathcal{H}^n}(vv^{\mathsf{H}}) \rangle = \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v)^{\mathsf{T}}\tilde{M}\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(v) + \mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(\mathrm{i}v)^{\mathsf{T}}\tilde{M}\mathfrak{J}_{\mathbb{C}^n_{\mathbb{R}}}(\mathrm{i}v),$$

which is nonnegative. Hence $\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\tilde{M}) \in \mathcal{H}^n_+$.

$[\supset]$ Let $M \in \mathcal{H}^n_+$. Since $M = \mathfrak{J}^*_{\mathcal{H}^n} \mathfrak{J}_{\mathcal{H}^n}(M)$ by (4.4) and $\mathfrak{J}_{\mathcal{H}^n}(M) \in \mathcal{S}^{2n}_+$ by point 1 of proposition 4.2, it follows that $M \in \mathfrak{J}^*_{\mathcal{H}^n}(\mathcal{S}^{2n}_+) = \hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\mathcal{S}^{2n}_+)$.

The second identity can be proven similarly. □

Note that $\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\tilde{M}) \succcurlyeq 0$ with $\tilde{M} \in \mathcal{S}^{2n}$ does not imply that $\tilde{M} \succcurlyeq 0$. As counter-example with $n = 1$, observe using (4.5) that $\hat{\mathfrak{J}}^*_{\mathcal{H}^n}(\tilde{M}) = 2 \succcurlyeq 0$ for the matrix

$$\tilde{M} = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \not\succcurlyeq 0.$$

4.2 Transformation into real numbers

4.2.1 Transformation of the primal problem into real numbers

Let us consider first the transformation of the primal problem (3.1) into real numbers. The complex constraints of this problem, those with index k in K_c, can be transformed into real constraints by using (2.7). This yields the following equivalent problem

$$(P') \quad \begin{cases} \inf \langle C, X \rangle_{\mathcal{H}^n} \\ \langle A_k, X \rangle_{\mathcal{H}^n} = b_k, \quad \forall\, k \in K_r \\ \langle \mathcal{H}(A_k), X \rangle_{\mathcal{H}^n} = \mathfrak{R}(b_k), \quad \forall\, k \in K_c \\ \langle \mathrm{i}\mathcal{Z}(A_k), X \rangle_{\mathcal{H}^n} = \mathfrak{I}(b_k), \quad \forall\, k \in K_c \\ X \succcurlyeq 0. \end{cases} \tag{4.8}$$

Since one can transform complex constraints into real ones, one can wonder whether it is useful to make more effort to deal with additional complex constraints (we have already said at the end of the paragraph containing (3.2) why it is not appropriate to discard the real affine constraints).

Problem (4.8) is still expressed in complex numbers, since C, A_k, $\mathcal{H}(A_k)$, $\mathrm{i}\mathcal{Z}(A_k)$, and X are complex Hermitian matrices. Using the isometry property (4.3), it looks natural to take as real number version of the primal problem, the following problem in $\tilde{X} \in \mathcal{S}^{2n}$:

$$\begin{cases} \inf \langle \mathfrak{J}_{\mathcal{H}^n}(C), \tilde{X} \rangle_{\mathcal{S}^{2n}} \\ \langle \mathfrak{J}_{\mathcal{H}^n}(A_k), \tilde{X} \rangle_{\mathcal{S}^{2n}} = b_k, \quad \forall\, k \in K_r \\ \langle \mathfrak{J}_{\mathcal{H}^n}(\mathcal{H}(A_k)), \tilde{X} \rangle_{\mathcal{S}^{2n}} = \mathfrak{R}(b_k), \quad \forall\, k \in K_c \\ \langle \mathfrak{J}_{\mathcal{H}^n}(\mathrm{i}\mathcal{Z}(A_k)), \tilde{X} \rangle_{\mathcal{S}^{2n}} = \mathfrak{I}(b_k), \quad \forall\, k \in K_c \\ \tilde{X} \succcurlyeq 0, \end{cases} \tag{4.9}$$

where $\tilde{X}$ plays the role of $\mathfrak{J}_{\mathcal{H}^n}(X)$ but, as we shall see in proposition 4.5, is not guaranteed to have that value. Instead of using the isomorphism $\mathfrak{J}_{\mathcal{H}^n}$ of section 4.1.1, one could have used the ring homomorphism $\hat{\mathfrak{J}}_{\mathcal{H}^n}$ of section 4.1.2, but with the additional factor 2 in the right-hand side of the affine constraints.

Observation 4.4 The first advantage of dealing directly with the complex SDO problem (3.1), with (3.4), rather than with its real number transformation (4.9) can already be observed at this point: *the former requires less memory storage*. Indeed the unknown matrix $X \in \mathcal{H}^n$ takes twice less memory storage than its real number counterpart $\tilde{X} \in \mathcal{S}^{2n}$ ($2n^2$ real numbers against $4n^2$ if the full matrices are stored). The same observation holds for the matrices C and A_k (with $k \in K_r$), when they are transformed into $\mathfrak{J}_{\mathcal{H}^n}(C)$ and $\mathfrak{J}_{\mathcal{H}^n}(A_k)$. For the matrices A_k (with $k \in K_c$), the ratio is 4, since the $2n^2$ real numbers of A_k become the $8n^2$ real numbers of $\mathfrak{J}_{\mathcal{H}^n}(\mathcal{H}(A_k))$ and $\mathfrak{J}_{\mathcal{H}^n}(\mathrm{i}\mathcal{Z}(A_k))$, if the full matrices are stored. □

Problem (4.9) can be written compactly as a standard primal SDO problem in real numbers as follows

$$(\tilde{P}) \quad \begin{cases} \inf \langle \tilde{C}, \tilde{X} \rangle_{\mathcal{S}^{2n}} \\ \tilde{\mathcal{A}}(\tilde{X}) = \tilde{b} \\ \tilde{X} \succcurlyeq 0, \end{cases} \tag{4.10}$$

where $\tilde{C} := \mathfrak{J}_{\mathcal{H}^n}(C)$, while the map $\tilde{\mathcal{A}} : \mathcal{S}^{2n} \to \mathbb{R}^{m_r+2m_c}$ and the vector $\tilde{b} \in \mathbb{R}^{m_r+2m_c}$, whose meanings are easily guessed from (4.9), are now defined with additional objects that will be useful below. The feasible set of problem $(\tilde{P})$ is denoted by $\mathcal{F}_{\tilde{P}}$, its solution set by $\mathcal{S}_{\tilde{P}}$, and its optimal value by $\operatorname{val}(\tilde{P})$.

The form of the right-hand side $\tilde{b}$ of the affine constraints in (4.9) suggests us to introduce the $\mathbb{R}$-linear bijection

$$\mathfrak{J}_{\mathbb{F}} := (a_{K_r}, a_{K_c}) \in \mathbb{F} := \mathbb{R}^{m_r} \times \mathbb{C}^{m_c} \mapsto \big(a_{K_r}, \mathfrak{R}(a_{K_c}), \mathfrak{I}(a_{K_c})\big) \in \mathbb{R}^{m_r+2m_c}, \tag{4.11}$$

since then

$$\tilde{b} = \mathfrak{J}_{\mathbb{F}}(b). \tag{4.12}$$

Note that, with the scalar product (3.6) defined on $\mathbb{F}$ and the Euclidean scalar product on $\mathbb{R}^{m_r+2m_c}$, there holds

$$\langle \mathfrak{J}_{\mathbb{F}}(a), \mathfrak{J}_{\mathbb{F}}(b) \rangle_{\mathbb{R}^{m_r+2m_c}} = \langle a, b \rangle_{\mathbb{F}}, \tag{4.13}$$

so that $\mathfrak{J}_{\mathbb{F}}$ is also an isometry. It is also nonsingular and satisfies

$$\mathfrak{J}_{\mathbb{F}}^*(\tilde{a}) = \mathfrak{J}_{\mathbb{F}}^{-1}(\tilde{a}) = (\tilde{a}_{K_r}, \tilde{a}_{K_c} + \mathrm{i}\tilde{a}'_{K_c}), \tag{4.14}$$

where we have assumed that $\tilde{a}$ reads $((\tilde{a}_k)_{k\in K_r}, (\tilde{a}_k)_{k\in K_c}, (\tilde{a}'_k)_{k\in K_c}) \in \mathbb{R}^{m_r+2m_c}$.

We now claim that the map $\tilde{\mathcal{A}} : \mathcal{S}^{2n} \to \mathbb{R}^{m_r+2m_c}$ defined by

$$\tilde{\mathcal{A}} := \mathfrak{J}_{\mathbb{F}} \circ \mathcal{A} \circ \mathfrak{J}_{\mathcal{H}^n}^* \tag{4.15}$$

allows us to write the constraints of (4.9) as $\tilde{\mathcal{A}}(\tilde{X}) = \tilde{b}$ like in (4.10). Indeed, by definition of $\mathcal{A}$ in (3.4), there holds for $\tilde{X} \in \mathcal{S}^{2n}$:

$$(\mathcal{A} \circ \mathfrak{J}_{\mathcal{H}^n}^*)(\tilde{X}) = \begin{pmatrix} \langle A_k, \mathfrak{J}_{\mathcal{H}^n}^*(\tilde{X}) \rangle_{\mathcal{H}^n}, \ k \in K_r \\ \langle A_k, \mathfrak{J}_{\mathcal{H}^n}^*(\tilde{X}) \rangle_{\mathbb{C}^{n\times n}}, \ k \in K_c \end{pmatrix}.$$

Therefore, by the definition of $\mathfrak{J}_{\mathbb{F}}$ in (4.11) and by (2.7), we get

$$(\mathfrak{J}_{\mathbb{F}} \circ \mathcal{A} \circ \mathfrak{J}_{\mathcal{H}^n}^*)(\tilde{X}) = \begin{pmatrix} \langle A_k, \mathfrak{J}_{\mathcal{H}^n}^*(\tilde{X}) \rangle_{\mathcal{H}^n}, \ k \in K_r \\ \langle \mathcal{H}(A_k), \mathfrak{J}_{\mathcal{H}^n}^*(\tilde{X}) \rangle_{\mathcal{H}^n}, \ k \in K_c \\ \langle \mathrm{i}\mathcal{Z}(A_k), \mathfrak{J}_{\mathcal{H}^n}^*(\tilde{X}) \rangle_{\mathcal{H}^n}, \ k \in K_c \end{pmatrix} = \begin{pmatrix} \langle \mathfrak{J}_{\mathcal{H}^n}(A_k), \tilde{X} \rangle_{\mathcal{S}^{2n}}, \ k \in K_r \\ \langle \mathfrak{J}_{\mathcal{H}^n}(\mathcal{H}(A_k)), \tilde{X} \rangle_{\mathcal{S}^{2n}}, \ k \in K_c \\ \langle \mathfrak{J}_{\mathcal{H}^n}(\mathrm{i}\mathcal{Z}(A_k)), \tilde{X} \rangle_{\mathcal{S}^{2n}}, \ k \in K_c \end{pmatrix}.$$

Since the right-hand side provides the constraints of (4.9), (4.15) holds. From (4.15) and (4.4), one deduces that

$$\tilde{\mathcal{A}} \circ \mathfrak{J}_{\mathcal{H}^n} := \mathfrak{J}_{\mathbb{F}} \circ \mathcal{A}. \tag{4.16}$$

In the next proposition, we highlight some of the links between the feasible and solution sets of (P) and $(\tilde{P})$. In particular it is shown how $\mathfrak{J}_{\mathcal{H}^n}^*$ can be used to recover a solution to (P) from a solution to its real number counterpart $(\tilde{P})$; this will be useful in the experiments. The proposition does not assume either that the feasible set or solution set of (P) or $(\tilde{P})$ is nonempty, or even that the optimal values are finite.

Proposition 4.5 (feasible and solution sets of (P) and $(\tilde{P})$) *There hold*

$$\mathfrak{I}_{\mathcal{H}^n}(\mathcal{F}_P) = \mathcal{F}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n}) \quad \text{and} \quad \mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{F}_{\tilde{P}}) = \mathcal{F}_P, \tag{4.17}$$
$$\mathfrak{I}_{\mathcal{H}^n}(\mathcal{S}_P) = \mathcal{S}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n}) \quad \text{and} \quad \mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{S}_{\tilde{P}}) = \mathcal{S}_P, \tag{4.18}$$

and $\mathrm{val}(P) = \mathrm{val}(\tilde{P})$.

Proof. 1) Consider first (4.17).

- $[\mathfrak{I}_{\mathcal{H}^n}(\mathcal{F}_P) \subset \mathcal{F}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})]$ Let $X \in \mathcal{F}_P$ and set $\tilde{X} := \mathfrak{I}_{\mathcal{H}^n}(X)$. By (4.16) and (4.12), it results that $\tilde{\mathcal{A}}(\tilde{X}) = \mathfrak{I}_{\mathbb{F}}(\mathcal{A}(X)) = \mathfrak{I}_{\mathbb{F}}(b) = \tilde{b}$. By point 1 in proposition 4.2, $\tilde{X} \succcurlyeq 0$. Hence $\tilde{X} \in \mathcal{F}_{\tilde{P}}$.
- $[\mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{F}_{\tilde{P}}) \subset \mathcal{F}_P]$ Let $\tilde{X} \in \mathcal{F}_{\tilde{P}}$ and set $X := \mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X})$. By the affine constraint of $(\tilde{P})$, (4.15), and (4.12), there holds $\mathfrak{I}_{\mathbb{F}}(\mathcal{A}(X)) = \tilde{b} = \mathfrak{I}_{\mathbb{F}}(b)$; hence $\mathcal{A}(X) = b$ by the injectivity of $\mathfrak{I}_{\mathbb{F}}$. By $\tilde{X} \succcurlyeq 0$ and proposition 4.3, $X \succcurlyeq 0$. Hence $\tilde{X} \in \mathcal{F}_P$.
- $[\mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{F}_{\tilde{P}}) \supset \mathcal{F}_P]$ Just apply $\mathfrak{I}^*_{\mathcal{H}^n}$ to both sides of the first proven inclusion $\mathfrak{I}_{\mathcal{H}^n}(\mathcal{F}_P) \subset \mathcal{F}_{\tilde{P}}$ and use (4.4).
- $[\mathfrak{I}_{\mathcal{H}^n}(\mathcal{F}_P) \supset \mathcal{F}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})]$ Let $\tilde{X} \in \mathcal{F}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})$. Then $\tilde{X} = \mathfrak{I}_{\mathcal{H}^n}(X)$ for some $X \in \mathcal{H}^n$ and one just has to prove that $X \in \mathcal{F}_P$. This comes from $X = \mathfrak{I}^*_{\mathcal{H}^n}\mathfrak{I}_{\mathcal{H}^n}(X) = \mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}) \in \mathcal{F}_P$, by $\tilde{X} \in \mathcal{F}_{\tilde{P}}$ and the second proven inclusion.

2) Consider now (4.18).

- $[\mathfrak{I}_{\mathcal{H}^n}(\mathcal{S}_P) \subset \mathcal{S}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})]$ Let $X \in \mathcal{S}_P$ and set $\tilde{X} := \mathfrak{I}_{\mathcal{H}^n}(X)$. By (4.17), $\tilde{X} \in \mathcal{F}_{\tilde{P}}$. Furthermore, by optimality, $\langle C, X\rangle_{\mathcal{H}^n} \leqslant \langle C, X'\rangle_{\mathcal{H}^n}$ for all $X' \in \mathcal{F}_P$. Using the isometry identity (4.3), it follows that
$$\forall\, X' \in \mathcal{F}_P: \quad \langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}\rangle_{\mathcal{S}^{2n}} \leqslant \langle \mathfrak{I}_{\mathcal{H}^n}(C), \mathfrak{I}_{\mathcal{H}^n}(X')\rangle_{\mathcal{S}^{2n}}. \tag{4.19}$$
Now let $\tilde{X}' \in \mathcal{F}_{\tilde{P}}$. Then $\mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}') \in \mathcal{F}_P$ by (4.17) and
$$\begin{aligned}
\langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}'\rangle_{\mathcal{S}^{2n}} &= \langle \mathfrak{I}_{\mathcal{H}^n}\mathfrak{I}^*_{\mathcal{H}^n}\mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}'\rangle_{\mathcal{S}^{2n}} \qquad [(4.4) \text{ and } C \in \mathcal{H}^n]\\
&= \langle \mathfrak{I}_{\mathcal{H}^n}(C), \mathfrak{I}_{\mathcal{H}^n}\mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}')\rangle_{\mathcal{S}^{2n}} \qquad [\text{adjoint definition}]\\
&\geqslant \langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}\rangle_{\mathcal{S}^{2n}} \qquad [\mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}') \in \mathcal{F}_P \text{ and } (4.19)].
\end{aligned}$$
Therefore $\tilde{X}$ is a solution to $(\tilde{P})$.
- $[\mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{S}_{\tilde{P}}) \subset \mathcal{S}_P]$ Let $\tilde{X} \in \mathcal{S}_{\tilde{P}}$ and set $X := \mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X})$. By (4.17), $X \in \mathcal{F}_P$. Furthermore, by optimality, $\langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}\rangle_{\mathcal{S}^{2n}} \leqslant \langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}'\rangle_{\mathcal{S}^{2n}}$ for all $\tilde{X}' \in \mathcal{F}_{\tilde{P}}$, so that
$$\forall\, \tilde{X}' \in \mathcal{F}_{\tilde{P}}: \quad \langle C, X\rangle_{\mathcal{H}^n} \leqslant \langle C, \mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}')\rangle_{\mathcal{H}^n}.$$
Now $\mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{F}_{\tilde{P}}) = \mathcal{F}_P$ by (4.17), hence X is a solution to (P).
- $[\mathfrak{I}^*_{\mathcal{H}^n}(\mathcal{S}_{\tilde{P}}) \supset \mathcal{S}_P]$ Just apply $\mathfrak{I}^*_{\mathcal{H}^n}$ to both sizes of the first proven inclusion $\mathfrak{I}_{\mathcal{H}^n}(\mathcal{S}_P) \subset \mathcal{S}_{\tilde{P}}$ and use (4.4).
- $[\mathfrak{I}_{\mathcal{H}^n}(\mathcal{S}_P) \supset \mathcal{S}_{\tilde{P}} \cap \mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})]$ Let $\tilde{X} := \mathfrak{I}_{\mathcal{H}^n}(X) \in \mathcal{S}_{\tilde{P}}$ for some $X \in \mathcal{H}^n$. Then $X = \mathfrak{I}^*_{\mathcal{H}^n}(\tilde{X}) \in \mathcal{S}_P$ by (4.4) and the second proven inclusion in point 2. Therefore $\tilde{X} = \mathfrak{I}_{\mathcal{H}^n}(X) \in \mathfrak{I}_{\mathcal{H}^n}(\mathcal{S}_P)$.

3) For X in $\mathcal{F}_P$, $\tilde{X} := \mathfrak{I}_{\mathcal{H}^n}(X)$ is in $\mathcal{F}_{\tilde{P}}$ by (4.17), so that $\langle C, X\rangle_{\mathcal{H}^n} = \langle \mathfrak{I}_{\mathcal{H}^n}(C), \tilde{X}\rangle_{\mathcal{S}^{2n}} \geqslant \mathrm{val}(\tilde{P})$. Hence $\mathrm{val}(P) \geqslant \mathrm{val}(\tilde{P})$. The reverse inequality is proven similarly. □

The next example shows that $\mathcal{F}_{\tilde{P}}$ and $\mathcal{S}_{\tilde{P}}$ may contain points that are not in $\mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})$. Therefore, the intersections with $\mathcal{R}(\mathfrak{I}_{\mathcal{H}^n})$ in the right-hand sides of the first identities in (4.17) and (4.18) cannot be removed.

Example 4.6 (a simple complex SDO problem) Consider one of the simplest complex SDO problems (3.1), in which $n = 2$, $m_r = 1$, $m_c = 0$,

$$C = I_2, \quad A = \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix} \in \mathcal{H}^2, \quad \text{and} \quad b = -2.$$

The problem can be rewritten in terms of the elements of $X \in \mathcal{H}^2$ as follows

$$\left\{ \begin{array}{l} \inf \; X_{11} + X_{22} \\ \mathfrak{I}(X_{12}) = 1 \\ X_{11} \geqslant 0, \quad X_{22} \geqslant 0, \quad X_{11}X_{22} \geqslant \mathfrak{R}(X_{12})^2 + 1. \end{array} \right. \tag{4.20}$$

This problem has the optimal value 2 and for unique solution

$$X^* = \begin{pmatrix} 1 & \mathrm{i} \\ -\mathrm{i} & 1 \end{pmatrix}.$$

The transformation of the problem into real numbers is the problem in $\tilde{X} \in \mathcal{S}^4$, obtained thanks to isomorphism $\mathfrak{J}_{\mathcal{H}^n}$ like in (4.9):

$$\left\{ \begin{array}{l} \inf \; \frac{1}{\sqrt{2}} (\tilde{X}_{11} + \tilde{X}_{22} + \tilde{X}_{33} + \tilde{X}_{44}) \\ -\tilde{X}_{14} + \tilde{X}_{23} + \tilde{X}_{32} - \tilde{X}_{41} = 2\sqrt{2} \\ \tilde{X} \geqslant 0. \end{array} \right. \tag{4.21}$$

We know from proposition 4.5 that

$$\mathfrak{J}_{\mathcal{H}^n}(X^*) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

is a solution to $(\tilde{P})$. We claim that $\mathfrak{J}_{\mathcal{H}^n}(X^*) + tD$ is also a solution to $(\tilde{P})$ if $t \in [0, 1]$ and

$$D = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & -1 & 0 \end{pmatrix} \in \mathcal{R}(\mathfrak{J}_{\mathcal{H}^n})^\perp = \mathcal{N}(\mathfrak{J}_{\mathcal{H}^n}^*),$$

see formula (4.5). Therefore, for this example, $\mathfrak{J}_{\mathcal{H}^n}(\mathcal{S}_P) \neq \mathcal{S}_{\tilde{P}}$.

Since $(\tilde{P})$ is a convex problem, it suffices to show that $\mathfrak{J}_{\mathcal{H}^n}(X^*) + D$ is in $\mathcal{S}_{\tilde{P}}$. Since $D \in \mathcal{N}(\mathfrak{J}_{\mathcal{H}^n}^*)$, there holds $\langle \mathfrak{J}_{\mathcal{H}^n}(C), \mathfrak{J}_{\mathcal{H}^n}(X^*) + D \rangle = \langle \mathfrak{J}_{\mathcal{H}^n}(C), \mathfrak{J}_{\mathcal{H}^n}(X^*) \rangle$, so that the objective of $(\tilde{P})$ takes at $\mathfrak{J}_{\mathcal{H}^n}(X^*) + D$ its optimal value. Let us now show that $\mathfrak{J}_{\mathcal{H}^n}(X^*) + D$ is feasible for $(\tilde{P})$. The affine constraint of $(\tilde{P})$ is satisfied by $\mathfrak{J}_{\mathcal{H}^n}(X^*) + D$, since $\langle \mathfrak{J}_{\mathcal{H}^n}(A), \mathfrak{J}_{\mathcal{H}^n}(X^*) + D \rangle = \langle A, X^* \rangle = b$. It remains to show that

$$\mathfrak{J}_{\mathcal{H}^n}(X^*) + D = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{pmatrix} \text{ is positive semi-definite.}$$

This property is a consequence of that fact that the matrix is symmetric and has only two eigenvalues, 0 and $4/\sqrt{2}$ (the eigenspace associated with the first eigenvalue is spanned by the three linearly independent vectors $(1, -1, 0, 0)$ $(0, 1, -1, 0)$, $(0, 0, 1, 1)$, and an eigenvector associated with the second eigenvalue is $(1, 1, 1, -1)$), which are nonegative. □

We now raise the question to know whether $\mathcal{S}_{\tilde{P}}$ may be unbounded while $\mathcal{S}_P$ is bounded (according to (4.18), $\mathcal{S}_{\tilde{P}}$ is is somehow "larger" than $\mathcal{S}_P$). This question is of interest for interior-point methods, since an unbounded solution set makes the solution difficult to compute by these algorithms. It would be unfortunate that a path-following interior-point method could be well defined for solving problem (P) but not for solving problem $(\tilde{P})$. The next proposition shows that this situation does not occur and therefore that *there is no advantage to find in the complex number SDO formulation on this topics.*

Proposition 4.7 (bounded solution sets for (P) and $(\tilde{P})$) *There hold*

$$\mathcal{F}_P \text{ is bounded} \iff \mathcal{F}_{\tilde{P}} \text{ is bounded,} \tag{4.22}$$

$$\mathcal{S}_P \text{ is bounded} \iff \mathcal{S}_{\tilde{P}} \text{ is bounded.} \tag{4.23}$$

PROOF. [(4.22)] [$\Rightarrow$] Let X be feasible for (P). Hence $\mathcal{J}_{\mathcal{H}^n}(X)$ is feasible for $(\tilde{P})$ by (4.17). Since $\mathcal{F}_{\tilde{P}}$ is a closed convex set, its boundedness will be proven if we show that $\mathcal{J}_{\mathcal{H}^n}(X)+t\tilde{D} \in \mathcal{F}_{\tilde{P}}$ for $\tilde{D} \in \mathcal{S}^{2n}$ and all $t \geqslant 0$ implies that $\tilde{D} = 0$ [28; theorem 8.4]. From the condition $\mathcal{J}_{\mathcal{H}^n}(X)+t\tilde{D} \succcurlyeq 0$ for all $t \geqslant 0$, we get

$$\tilde{D} \succcurlyeq 0.$$

On the other hand $\mathcal{J}_{\mathcal{H}^n}^*(\tilde{D}) = 0$ since $\mathcal{J}_{\mathcal{H}^n}^*(\mathcal{J}_{\mathcal{H}^n}(X)+t\tilde{D}) = X + t\mathcal{J}_{\mathcal{H}^n}^*(\tilde{D})$ is feasible for (P) for all $t \geqslant 0$, by (4.17), and $\mathcal{F}_P$ is bounded. Hence, by (4.5), $\tilde{D}$ is of the form

$$\tilde{D} = \begin{pmatrix} \tilde{D}_{11} & \tilde{D}_{12} \\ \tilde{D}_{12} & -\tilde{D}_{11} \end{pmatrix},$$

where the blocks $\tilde{D}_{ij} \in \mathcal{S}^n$. Since both $\tilde{D}_{11}$ and $-\tilde{D}_{11}$ must be positive semidefinite to preserve $\tilde{D} \succcurlyeq 0$, there must hold $\tilde{D}_{11} = 0$. Next $\tilde{D}_{12} = 0$ to preserve $\tilde{D} \succcurlyeq 0$. We have shown that $\tilde{D} = 0$ and therefore the boundedness of $\mathcal{F}_{\tilde{P}}$.

[(4.22)] [$\Leftarrow$] Since $\mathcal{F}_P = \mathcal{J}_{\mathcal{H}^n}^*(\mathcal{F}_{\tilde{P}})$ by (4.17), the boundedness of $\mathcal{F}_{\tilde{P}}$ implies that of $\mathcal{F}_P$.

[(4.23)] The proof is similar, using (4.18) instead of (4.17). □

4.2.2 Transformation of the dual problem into real numbers

The real Lagrange dual SDO problem reads

$$\left\{ \begin{array}{l} \sup \ \langle \mathcal{J}_{\mathbb{F}}(b), \tilde{y} \rangle_{\mathbb{R}^{m_r+2m_c}} \\ \sum_{k\in K_r} \tilde{y}_k \mathcal{J}_{\mathcal{H}^n}(A_k) + \sum_{l\in K_c} \tilde{y}_l \mathcal{J}_{\mathcal{H}^n}(\mathcal{H}(A_l)) + \sum_{l\in K_c} \tilde{y}'_l \mathcal{J}_{\mathcal{H}^n}(\mathrm{i}\mathcal{Z}(A_l)) + \tilde{S} = \mathcal{J}_{\mathcal{H}^n}(C) \\ \tilde{S} \succcurlyeq 0. \end{array} \right. \tag{4.24}$$

This one can be obtained by writing the Lagrange dual of the real SDO problem (4.10) or by translating the complex Lagrange dual (3.5) into real numbers (commutative diagram). In the latter case, $\tilde{y} = \mathcal{J}_{\mathbb{F}}(y)$, $\tilde{S}$ plays the role of $\mathcal{J}_{\mathcal{H}^n}(S)$, the objective is obtained by using the isometry identity (4.13) and the affine constraint is the result of applying $\mathcal{J}_{\mathcal{H}^n}$ to both sides of the affine constraint of (3.5).

Using the isometry

$$\mathcal{J}_{\mathbb{F}\times\mathcal{H}^n} : (y,S) \in \mathbb{F} \times \mathcal{H}^n \mapsto (\mathcal{J}_{\mathbb{F}}(y), \mathcal{J}_{\mathcal{H}^n}(S)) \in \mathbb{R}^{m_r+2m_c} \times \mathcal{S}^{2n}, \tag{4.25}$$

one can show results similar to propositions 4.5 and 4.7. All these yield observation 4.10.

Proposition 4.8 (feasible and solution sets of (D) and $(\tilde{D})$) *There hold* $\mathcal{J}_{\mathbb{F}\times\mathcal{H}^n}(\mathcal{F}_D) = \mathcal{F}_{\tilde{D}} \cap \mathcal{R}(\mathcal{J}_{\mathbb{F}\times\mathcal{H}^n})$, $\mathcal{J}^*_{\mathbb{F}\times\mathcal{H}^n}(\mathcal{F}_{\tilde{D}}) = \mathcal{F}_D$, $\mathcal{J}_{\mathbb{F}\times\mathcal{H}^n}(\mathcal{S}_D) = \mathcal{S}_{\tilde{D}} \cap \mathcal{R}(\mathcal{J}_{\mathbb{F}\times\mathcal{H}^n})$, $\mathcal{J}^*_{\mathbb{F}\times\mathcal{H}^n}(\mathcal{S}_{\tilde{D}}) = \mathcal{S}_D$, *and* $\operatorname{val}(D) = \operatorname{val}(\tilde{D})$.

Proposition 4.9 (bounded solution sets for (D) and $(\tilde{D})$) *$\mathcal{F}_D$ is bounded if and only if $\mathcal{F}_{\tilde{D}}$ is bounded. $\mathcal{S}_D$ is bounded if and only if $\mathcal{S}_{\tilde{D}}$ is bounded.*

Observation 4.10 There is no advantage in using the real or complex formulation of the SDO problem with respect to the well-posedness of a primal, dual, or primal-dual interior-point method, since the primal and dual solution sets are nonempty and bounded simultaneously in the two formulations. □

5 A feasible predictor-corrector algorithm

As already said, the goal of this plea is to highlight the advantages of solving a complex SDO problem directly with a complex SDO solver, not to present a competitive complex SDO solver. With that objective in mind, we have implemented in Matlab a simplified version of a predictor-corrector path-following interior-point algorithm, using the primal-dual Nesterov-Todd (NT) direction, in which the iterates always satisfy the affine constraint (sometimes qualified as *feasible algorithm*) and remain close to the central path. The algorithm is introduced in section 5.1. Section 5.2 shows why the convergence of the algorithm is unlikely deducible from the one of its real version applied to the associated real SDO problem. Section 5.3 gives the details on the computation of the NT directions computed by the algorithm *in complex numbers*. The comparison of the computational effort required by the complex and real algorithms is done in section 5.4.

We denote the *primal-dual strictly feasible sets* by

$$
\begin{aligned}
\mathcal{F}^s &:= \{(X,y,S) \in \mathcal{H}^n_{++} \times \mathbb{F} \times \mathcal{H}^n_{++} : \mathcal{A}(X) = b,\ \mathcal{A}^*(y) + S = C\},\\
\tilde{\mathcal{F}}^s &:= \{(\tilde{X},\tilde{y},\tilde{S}) \in \mathcal{S}^{2n}_{++} \times \mathbb{R}^{m_r+2m_c} \times \mathcal{S}^{2n}_{++} : \tilde{\mathcal{A}}(\tilde{X}) = \tilde{b},\ \tilde{\mathcal{A}}^*(\tilde{y}) + \tilde{S} = \tilde{C}\}.
\end{aligned}
$$

In all this section, we assume that

$$\mathcal{F}^s \neq \varnothing \text{ and } \mathcal{A} \text{ is surjective.} \tag{5.1}$$

By propositions 4.5, 4.8, 4.2, and 4.3, and by remark 3.3(3), this is equivalent to saying that $\tilde{\mathcal{F}}^s \neq \varnothing$ and $\tilde{\mathcal{A}}$ is surjective.

5.1 Description of the algorithm

This section presents an algorithm that is a faithful adaptation to complex numbers of a predictor-corrector algorithm for real SDO problems discussed and analyzed in details by de Klerk [4; § 7.6]. This one is descending from the homonymous algorithm described by Roos, Terlaky, and Vial [29; § 7.7.4] for linear optimization, itself inherited from [22, 23]. The algorithm is *primal-dual*, which means that the iterates are triplets $z := (X, y, S)$ made of primal $X \in \mathcal{H}^n$ and dual $(y, S) \in \mathbb{F} \times \mathcal{H}^n$ variables. It is also *feasible* in the sense that the iterates belong to $\mathcal{F}^s$. The iterates are forced to follow the *central path* $\mathcal{C}$, which is the image of the map $\mu \in \mathbb{R}_{++} \mapsto (X_\mu, y_\mu, S_\mu) \in \mathcal{H}^n_{++} \times \mathbb{F} \times \mathcal{H}^n_{++}$, giving the unique solution to the system

$$\begin{cases} \mathcal{A}^*(y) + S = C \\ \mathcal{A}(X) = b \\ XS = \mu I. \end{cases}$$

Uniqueness indeed occurs when (5.1) holds.

The NT direction $d = (d_X, d_y, d_S) \in \mathcal{H}^n \times \mathbb{F} \times \mathcal{H}^n$ at a point $z \in \mathcal{F}^s$ is the solution to

$$\mathcal{A}^*(d_y) + d_S = 0, \tag{5.2a}$$

$$\mathcal{A}(d_X) = 0, \tag{5.2b}$$

$$d_X + W d_S W = \mu S^{-1} - X, \tag{5.2c}$$

where $W \in \mathcal{H}^n$ is the scaling positive definite matrix defined by

$$W := X^{1/2}(X^{1/2} S X^{1/2})^{-1/2} X^{1/2} = S^{-1/2}(S^{1/2} X S^{1/2})^{1/2} S^{-1/2}, \tag{5.3}$$

in which the matrix square roots only act on Hermitian matrices (see [26, 16; 1997] and [4; § 7.1]). This is the unique matrix in $\mathcal{H}^n_{++}$ satisfying $WSW = X$. Note that uniqueness of the NT direction is actually ensured when the regularity assumption (5.1) holds.

When $\mu = 0$, the direction is said to be a *predictor direction* and is denoted by d^a (the superscript a is also standard and comes from *affine scaling*); when μ has the value

$$\bar{\mu}_{\mathcal{H}^n}(z) := \frac{\langle X, S \rangle_{\mathcal{H}^n}}{n}, \tag{5.4}$$

the direction is said to be a *corrector direction* and is denoted by d^c (the superscript c is standard and stands for *centering*). Each iteration of the algorithm is composed of a prediction phase followed by a correction phase (or vice versa). The goal of the predictor phase is to decrease as much as possible the positive value of $\bar{\mu}_{\mathcal{H}^n}$ along the direction d^a (it vanishes at a solution), which has the side effect of moving the intermediate iterate $z' := z + \alpha d^a$ away from the central path. The stepsize along the d^a is fixed to

$$\alpha := \frac{2}{1 + \sqrt{1 + \frac{13}{2\bar{\mu}_{\mathcal{H}^n}(z)} \| W^{-1/2} d^a_X d^a_S W^{1/2} + W^{1/2} d^a_S d^a_X W^{-1/2} \|_F}}. \tag{5.5}$$

The goal of the corrector phase is then to put back the next iterate in an appropriate neighborhood of the central path by moving along d^c with a unit stepsize. One often refers to Mizuno, Todd, and Ye [23] for the structure of the algorithm just described [29, 4]. Here is a more schematic description.

Algorithm 5.1 (feasible predictor-corrector) A tolerance $\varepsilon > 0$ on $\bar{\mu}(z)$ is given to determine when stopping the iterations. One iteration of the algorithm, from $z \in \mathcal{F}^s$ to $z_+ \in \mathcal{F}^s$, is formed of the following three phases.

1. *Stopping test.* If $\bar{\mu}(z) \leqslant \varepsilon$, stop.
2. *Prediction phase.* Compute the predictor NT direction d^a at z, as a solution of (5.2) with $\mu = 0$, and the stepsize $\alpha > 0$ given by (5.5). Compute the intermediate iterate

$$z' := z + \alpha d^a.$$

3. *Correction phase.* Compute the corrector NT direction d^c, as a solution of (5.2) with $z = z'$ and $\mu = \bar{\mu}(z')$. Take as next iterate

$$z_+ := z' + d^c.$$

To work properly, the intermediate iterates (z')'s must be maintained in some neighborhood $\mathcal{V}(\theta)$ of the central path that we now define. Since $W^{-1/2} XSW^{1/2}$ is Hermitian, one can take its square root and define the following scaled variable

$$V \equiv V(z) := \left(W^{-1/2} XSW^{1/2}\right)^{1/2} = W^{-1/2} XW^{-1/2} = W^{1/2} SW^{1/2}. \tag{5.6}$$

Observe now that for a point $z \in \mathcal{F}^s$ and the central path $\mathcal{C}$ if and only if $\bar{\mu}_{\mathcal{H}^n}(z)^{-1/2} V(z) = \bar{\mu}_{\mathcal{H}^n}(z)^{1/2} V(z)^{-1}$. Therefore, it makes sense to define the *proximity measure* of the central path as the map

$$\delta : z \in \mathcal{F}^s \mapsto \delta(z) := \frac{1}{2} \left\| \bar{\mu}_{\mathcal{H}^n}(z)^{-1/2} V(z) - \bar{\mu}_{\mathcal{H}^n}(z)^{1/2} V(z)^{-1} \right\|_F \in \mathbb{R}. \tag{5.7}$$

To this proximity measure, one associates the following neighborhoods of the central path, depending on the parameter $\theta \in \mathbb{R}_+$:

$$\mathcal{V}(\theta) := \{z \in \mathcal{F}^s : \delta(z) \leqslant \theta\}. \tag{5.8}$$

5.2 Non corresponding iterates

In view of the nice correspondences between the feasible and solution sets of problems (P) and $(\tilde{P})$ established in section 4.2, one could reasonably think that the iterates generated by the predictor-corrector algorithm 5.1, applied to problem (P), are in correspondence with those generated by the real version of this algorithm, applied to the real transformed SDO problem $(\tilde{P})$. The goal of this section is to show that this is unlikely. The indication is that some natural linear map, introduced in section 5.2.1, allows us to establish a correspondence between many objects generated during the particular examined iteration (sections 5.2.2, 5.2.3, 5.2.4, and 5.2.6), but not for the stepsizes in the predictor phase, which is larger in the complex space (section 5.2.5). As a result, the complex number algorithm should converge faster than its real analogue (observation 5.2).

5.2.1 A possible correspondence

Let us assume that the current iterate $z := (X, y, S) \in \mathcal{H}^n \times \mathbb{F} \times \mathcal{H}^n$ generated by algorithm 5.1 for solving the complex SDO problem (3.1) is in correspondence with the iterate $\tilde{z} := (\tilde{X}, \tilde{y}, \tilde{S}) \in \mathcal{S}^{2n} \times \mathbb{R}^{m_r + 2m_c} \times \mathcal{S}^{2n}$ generated by the real version of this algorithm for solving the real version (4.10) of (3.1), through the following rules

$$\tilde{X} = \mathcal{J}_{\mathcal{H}^n}(X), \qquad \tilde{y} = \mathcal{J}_{\mathbb{F}}(y), \qquad \text{and} \qquad \tilde{S} = \mathcal{J}_{\mathcal{H}^n}(S). \tag{5.9}$$

This correspondence is suggested by the fact that $\tilde{X}$ plays the role of $\mathfrak{J}_{\mathcal{H}^n}(X)$ in (4.9) and $(\tilde{y}, \tilde{S})$ plays the role of $\mathfrak{J}_{\mathbb{F}\times\mathcal{H}^n}(y,S)$ in (4.24). Propositions 4.5 and 4.8 reinforce the logic of this correspondence. In agreement with (5.9), we introduce the global correspondence map

$$\mathfrak{J} : z = (X,y,S) \in \mathcal{H}^n \times \mathbb{F} \times \mathcal{H}^n \mapsto \mathfrak{J}(z) := (\mathfrak{J}_{\mathcal{H}^n}(X), \mathfrak{J}_{\mathbb{F}}(y), \mathfrak{J}_{\mathcal{H}^n}(S)). \tag{5.10}$$

We now ask whether the correspondence (5.9) is still satisfied after one iteration of the predictor-corrector algorithm on the complex and real transformed problems.

To fix the notation, let us specify how the standard real version of algorithm 5.1 proceeds on the real transformed problem (4.10). The variable update is done by

$$\tilde{X}_+ := \tilde{X} + \tilde{\alpha}\tilde{d}_{\tilde{X}}, \qquad \tilde{y}_+ := \tilde{y} + \tilde{\alpha}\tilde{d}_{\tilde{y}}, \qquad \text{and} \qquad \tilde{S}_+ := \tilde{S} + \tilde{\alpha}\tilde{d}_{\tilde{S}}.$$

where $\tilde{d}$ is the solution to

$$\tilde{\mathcal{A}}^*(\tilde{d}_{\tilde{y}}) + \tilde{d}_{\tilde{S}} = 0, \tag{5.11a}$$

$$\tilde{\mathcal{A}}(\tilde{d}_{\tilde{X}}) = 0, \tag{5.11b}$$

$$\tilde{d}_{\tilde{X}} + \tilde{W}\tilde{d}_{\tilde{S}}\tilde{W} = \tilde{\mu}\tilde{S}^{-1} - \tilde{X}. \tag{5.11c}$$

The weighting positive definite matrix $\tilde{W} \in \mathcal{S}^{2n}$ is defined by the analog of (5.3), namely

$$\tilde{W} := \tilde{X}^{1/2}(\tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2})^{-1/2}\tilde{X}^{1/2} = \tilde{S}^{-1/2}(\tilde{S}^{1/2}\tilde{X}\tilde{S}^{1/2})^{1/2}\tilde{S}^{-1/2} \in \mathcal{S}^{2n} \tag{5.12}$$

and $\tilde{\mu} \in \mathbb{R}_{++}$ is a parameter. In the prediction phase at $\tilde{z}$, $\tilde{d}$ is denoted by $\tilde{d}^a$, μ is set to 0, and the stepsize $\tilde{\alpha} > 0$ is given by the analog of (5.5), namely

$$\tilde{\alpha} = \frac{2}{1 + \sqrt{1 + \frac{13}{2\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})}\|\tilde{W}^{-1/2}\tilde{d}^a_{\tilde{X}}\tilde{d}^a_{\tilde{S}}\tilde{W}^{1/2} + \tilde{W}^{1/2}\tilde{d}^a_{\tilde{S}}\tilde{d}^a_{\tilde{X}}\tilde{W}^{-1/2}\|_F}}, \tag{5.13}$$

where $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z}) := \langle \tilde{X}, \tilde{S}\rangle_{\mathcal{S}^{2n}}/(2n)$. In the correction phase at $\tilde{z}' := \tilde{z} + \tilde{\alpha}\tilde{d}^a$, $\tilde{d}$ is denoted by $\tilde{d}^c$, μ is set to $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z}')$, and $\tilde{\alpha} = 1$.

Since it is desirable to have $\tilde{X}+\tilde{\alpha}\tilde{d}^a_{\tilde{X}} = \mathfrak{J}_{\mathcal{H}^n}(X+\alpha d^a_X)$ and since $\mathfrak{J}_{\mathcal{H}^n}(X+\alpha d^a_X) = \tilde{X}+\alpha\mathfrak{J}_{\mathcal{H}^n}(d^a_X)$, it is natural to wonder whether

$$\tilde{d}_{\tilde{X}} = \mathfrak{J}_{\mathcal{H}^n}(d_X), \qquad \tilde{d}_{\tilde{y}} = \mathfrak{J}_{\mathbb{F}}(d_y), \qquad \text{and} \qquad \tilde{d}_{\tilde{S}} = \mathfrak{J}_{\mathcal{H}^n}(d_S), \tag{5.14a}$$

$$\tilde{\alpha} = \alpha, \tag{5.14b}$$

where $\tilde{d}$ and d stand for the prediction or correction directions.

5.2.2 Correspondence between $\tilde{W}$ and W

By the formulas (5.12) and (5.3) of $\tilde{W}$ and W, and by propositions 4.1 and 4.2, one gets

$$\tilde{W} = \hat{\mathfrak{J}}_{\mathcal{H}^n}(X)^{1/2}\left(\hat{\mathfrak{J}}_{\mathcal{H}^n}(X)^{1/2}\hat{\mathfrak{J}}_{\mathcal{H}^n}(S)\hat{\mathfrak{J}}_{\mathcal{H}^n}(X)^{1/2}\right)^{-1/2}\hat{\mathfrak{J}}_{\mathcal{H}^n}(X)^{1/2} = \hat{\mathfrak{J}}_{\mathcal{H}^n}(W). \tag{5.15}$$

5.2.3 Correspondence between $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})$ and $\bar{\mu}_{\mathcal{H}^n}(z)$

With the definitions of $\bar{\mu}_{\mathcal{H}^n}(z)$ in (5.4) and $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})$ after (5.13), and with (4.3), we have

$$\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z}) = \frac{\langle \tilde{X}, \tilde{S}\rangle_{\mathcal{S}^{2n}}}{2n} = \frac{\langle \mathfrak{J}_{\mathcal{H}^n}(X), \mathfrak{J}_{\mathcal{H}^n}(S)\rangle_{\mathcal{S}^{2n}}}{2n} = \frac{\langle X, S\rangle_{\mathcal{H}^n}}{2n} = \frac{1}{2}\bar{\mu}_{\mathcal{H}^n}(z). \tag{5.16}$$

5.2.4 Correspondence between the Newton directions $\tilde{d}$ and d

Let us now show that $\tilde{d}$ defined by (5.14a) is indeed the solution to the Newton system in the real space (5.11). Since, by the regularity assumption (5.1), the system (5.11) has a unique solution, it suffices to show that it is satisfied by that $\tilde{d}$.

- [(5.11a)] From (4.15), it results that $\mathfrak{J}_{\mathcal{H}^n} \circ \mathcal{A}^* \circ \mathfrak{J}_{\mathbb{F}}^* = \tilde{\mathcal{A}}^*$ and, from (4.14), that $\mathfrak{J}_{\mathcal{H}^n} \circ \mathcal{A}^* = \tilde{\mathcal{A}}^* \circ \mathfrak{J}_{\mathbb{F}}$. Then, applying $\mathfrak{J}_{\mathcal{H}^n}$ to equation (5.2a) yields indeed $\tilde{\mathcal{A}}^*(\mathfrak{J}_{\mathbb{F}}(d_y)) + \mathfrak{J}_{\mathcal{H}^n}(d_S) = 0$ or $\tilde{\mathcal{A}}^*(\tilde{d}_{\tilde{y}}) + \tilde{d}_{\tilde{S}} = 0$ if we assume (5.14a).
- [(5.11b)] From (4.15) and (4.4), it results that $\mathfrak{J}_{\mathbb{F}} \circ \mathcal{A} = \tilde{\mathcal{A}} \circ \mathfrak{J}_{\mathcal{H}^n}$. Then, applying $\mathfrak{J}_{\mathbb{F}}$ to equation (5.2b) yields indeed $\tilde{\mathcal{A}}(\mathfrak{J}_{\mathcal{H}^n}(d_X)) = 0$ or $\tilde{\mathcal{A}}(\tilde{d}_{\tilde{X}}) = 0$ if we assume (5.14a).
- [(5.11c)] By applying the ring homomorphism $\hat{\mathfrak{J}}_{\mathcal{H}^n} := \sqrt{2}\mathfrak{J}_{\mathcal{H}^n}$ to both sides of equation (5.2c) and using proposition 4.1, one gets
$$\hat{\mathfrak{J}}_{\mathcal{H}^n}(d_X) + \hat{\mathfrak{J}}_{\mathcal{H}^n}(W)\hat{\mathfrak{J}}_{\mathcal{H}^n}(d_S)\hat{\mathfrak{J}}_{\mathcal{H}^n}(W) = \mu\hat{\mathfrak{J}}_{\mathcal{H}^n}(S)^{-1} - \hat{\mathfrak{J}}_{\mathcal{H}^n}(X)$$
or, with (5.14a), (5.15), (5.9), and division by $\sqrt{2}$,
$$\tilde{d}_{\tilde{X}} + \tilde{W}\tilde{d}_{\tilde{S}}\tilde{W} = \frac{\mu}{2}\tilde{S}^{-1} - \tilde{X}$$
which is indeed (5.11c), either when $\mu = 0$ (prediction direction) or when $\mu = \bar{\mu}_{\mathcal{H}^n}(z')$ (correction direction, use (5.16) in that case), provided $\tilde{z}' = \mathfrak{J}_{\mathcal{H}^n}(\tilde{z})$ (this last correspondence has not been established yet, and will not occur actually; see the next section).

5.2.5 Correspondence between the stepsizes $\tilde{\alpha}$ and α

We still have to see how the stepsizes $\tilde{\alpha}$ and α correspond to each other. We have seen in the previous section that (5.14a) holds for the prediction phase. Then, using proposition 4.1 and (5.15), we get

$$\begin{aligned}
\tilde{W}^{-1/2}\tilde{d}^a_{\tilde{X}}\tilde{d}^a_{\tilde{S}}\tilde{W}^{1/2} &= \frac{1}{2}\hat{\mathfrak{J}}_{\mathcal{H}^n}(W^{-1/2})\hat{\mathfrak{J}}_{\mathcal{H}^n}(d^a_X)\hat{\mathfrak{J}}_{\mathcal{H}^n}(d^a_S)\hat{\mathfrak{J}}_{\mathcal{H}^n}(W^{1/2}) \\
&= \frac{1}{2}\hat{\mathfrak{J}}_{\mathcal{H}^n}(W^{-1/2}d^a_X d^a_S W^{1/2}) \\
&= \frac{1}{\sqrt{2}}\mathfrak{J}_{\mathcal{H}^n}(W^{-1/2}d^a_X d^a_S W^{1/2}).
\end{aligned}$$

Similarly, $\tilde{W}^{1/2}\tilde{d}^a_{\tilde{S}}\tilde{d}^a_{\tilde{X}}\tilde{W}^{-1/2} = (1/\sqrt{2})\,\mathfrak{J}_{\mathcal{H}^n}(W^{1/2}d^a_S d^a_X W^{-1/2})$. Therefore, since $\mathfrak{J}_{\mathcal{H}^n}$ is an isometry, one gets

$$\|\tilde{W}^{-1/2}\tilde{d}^a_{\tilde{X}}\tilde{d}^a_{\tilde{S}}\tilde{W}^{1/2} + \tilde{W}^{1/2}\tilde{d}^a_{\tilde{S}}\tilde{d}^a_{\tilde{X}}\tilde{W}^{-1/2}\|_F = \frac{1}{\sqrt{2}}\|W^{-1/2}d^a_X d^a_S W^{1/2} + W^{1/2}d^a_S d^a_X W^{-1/2}\|_F.$$

Using (5.16), one term in the denominator of the stepsize formula (5.13) reads

$$\begin{aligned}
&\left(\frac{13}{2\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})}\|\tilde{W}^{-1/2}\tilde{d}^a_{\tilde{X}}\tilde{d}^a_{\tilde{S}}\tilde{W}^{1/2} + \tilde{W}^{1/2}\tilde{d}^a_{\tilde{S}}\tilde{d}^a_{\tilde{X}}\tilde{W}^{-1/2}\|_F\right) \\
&\quad = \sqrt{2}\left(\frac{13}{2\bar{\mu}_{\mathcal{H}^n}(z)}\|W^{-1/2}d^a_X d^a_S W^{1/2} + W^{1/2}d^a_S d^a_X W^{-1/2}\|_F\right).
\end{aligned}$$

By the factor $\sqrt{2}$ in the right-hand side, the formulas (5.13) and (5.5) of the stepsizes $\tilde{\alpha}$ and α yield the inequality

$$\tilde{\alpha} < \alpha. \tag{5.17}$$

Hence the stepsizes taken in the prediction phases of the predictor-corrector algorithm when it solves the complex and associated real SDO problems are not identical.

5.2.6 Correspondence between the proximity measures to the central paths

The stepsize inequality (5.17) can be explained by the fact that the neighborhoods of the central path are larger for the complex problem than for the real transformed problem, in a sense that is now specified. One can introduce a scaled variable $\tilde{V} \equiv \tilde{V}(\tilde{z})$ in the real space as we did by (5.6) for the scaled variable $V \equiv V(z)$ in the complex space. The two variables are linked as follows:

$$\begin{aligned}
\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})^{-1/2}\,\tilde{V} &= \bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})^{-1/2}\,\tilde{W}^{-1/2}\tilde{X}\tilde{W}^{-1/2} \qquad [\text{definition of } \tilde{V}] \\
&= \bar{\mu}_{\mathcal{H}^n}(z)^{-1/2}\,\hat{\mathcal{J}}_{\mathcal{H}^n}(W)^{-1/2}\hat{\mathcal{J}}_{\mathcal{H}^n}(X)\hat{\mathcal{J}}_{\mathcal{H}^n}(W)^{-1/2} \qquad [(5.16),\ (5.15),\ (5.9)] \\
&= \bar{\mu}_{\mathcal{H}^n}(z)^{-1/2}\,\hat{\mathcal{J}}_{\mathcal{H}^n}\left(W^{-1/2}XW^{-1/2}\right) \qquad [\text{proposition } 4.1] && (5.18) \\
&= \sqrt{2}\,\mathcal{J}_{\mathcal{H}^n}\left(\bar{\mu}_{\mathcal{H}^n}(z)^{-1/2}\,V\right). && (5.19)
\end{aligned}$$

Furthermore, from (5.18) and proposition 4.1:

$$\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})^{1/2}\,\tilde{V}^{-1} = \bar{\mu}_{\mathcal{H}^n}(z)^{1/2}\,\hat{\mathcal{J}}_{\mathcal{H}^n}(V^{-1}) = \sqrt{2}\,\mathcal{J}_{\mathcal{H}^n}\left(\bar{\mu}_{\mathcal{H}^n}(z)^{1/2}\,V^{-1}\right). \tag{5.20}$$

One can now introduce a proximity measure $\tilde{\delta}$ in the real space, as we did for δ in the complex space by (5.7). The two proximity measures are linked as follows:

$$\begin{aligned}
\tilde{\delta}(\tilde{z}) &= \frac{1}{2}\left\|\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})^{-1/2}\,\tilde{V} - \bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})^{1/2}\,\tilde{V}^{-1}\right\|_F \qquad [\text{definition of } \tilde{\delta}\text{, like in } (5.7)] \\
&= \frac{1}{2}\left\|\sqrt{2}\,\mathcal{J}_{\mathcal{H}^n}\left(\bar{\mu}_{\mathcal{H}^n}(z)^{-1/2}\,V\right) - \sqrt{2}\,\mathcal{J}_{\mathcal{H}^n}\left(\bar{\mu}_{\mathcal{H}^n}(z)^{1/2}\,V^{-1}\right)\right\|_F \qquad [(5.19) \text{ and } (5.20)] \\
&= \frac{\sqrt{2}}{2}\left\|\bar{\mu}_{\mathcal{H}^n}(z)^{-1/2}\,V - \bar{\mu}_{\mathcal{H}^n}(z)^{1/2}\,V^{-1}\right\|_F \qquad [\mathcal{J}_{\mathcal{H}^n} \text{ is an isometry}] \\
&= \sqrt{2}\,\delta(z) \qquad [\text{definition of } \delta].
\end{aligned}$$

Therefore

$$\delta(z) \leqslant \theta \quad \Longleftrightarrow \quad \tilde{\delta}(\tilde{z}) \leqslant \sqrt{2}\,\theta. \tag{5.21}$$

5.2.7 Conclusion

Using the global correspondence map $\mathcal{J}$ defined in (5.10), the equivalence (5.21), and a size $\theta > 0$, the neighborhoods of the central paths $\mathcal{V}(\theta)$ given by (5.8) and its real analogue $\tilde{V}(\theta) := \{\tilde{z} \in \tilde{\mathcal{F}}^s : \tilde{\delta}(\tilde{z}) \leqslant \theta\}$ correspond to each other by

$$\mathcal{J}(\mathcal{V}(\theta)) = \{\mathcal{J}(z) : z \in \mathcal{F}^s,\ \delta(z) \leqslant \theta\} = \{\tilde{z} \in \tilde{\mathcal{F}}^s : \tilde{\delta}(\tilde{z}) \leqslant \sqrt{2}\,\theta\} = \tilde{\mathcal{V}}(\sqrt{2}\,\theta).$$

Therefore $\mathcal{J}(\mathcal{V}(\theta)) \supset \tilde{\mathcal{V}}(\theta)$, meaning that the transformed neighborhood $\mathcal{J}(\mathcal{V}(\theta))$ is larger than the neighborhood $\tilde{\mathcal{V}}(\theta)$ with the same size θ. This inclusion provides another way of interpreting (5.17), according to which, when started at corresponding iterates, the predictor-corrector algorithm can take larger stepsizes in the complex space than in the real space. This also means that the convergence of the complex interior-point SDO algorithm 5.1 requires a specific analysis, which is presented in [7].

Observation 5.2 Inequality (5.17) on the stepsizes suggests us that the considered predictor-corrector algorithm should converge more rapidly on the complex problem than on the real transformed problem. Indeed, the speed of convergence only depends on the reductions of $\bar{\mu}_{\mathcal{H}^n}(z)$ and $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})$ which are given at each iteration (during the predictor phase actually) by

$$\bar{\mu}_{\mathcal{H}^n}(z_+) = (1-\alpha)\bar{\mu}_{\mathcal{H}^n}(z) \qquad \text{and} \qquad \bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z}_+) = (1-\tilde{\alpha})\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z}).$$

Because $\alpha > \tilde{\alpha}$, the reduction of $\bar{\mu}_{\mathcal{H}^n}(z)$ towards zero for the complex problem is more important than that of $\bar{\mu}_{\mathcal{S}^{2n}}(\tilde{z})$ for its associated real problem, when z and $\tilde{z}$ correspond to each other by the isomorphism $\mathcal{J}$. This observation has been claimed with the conditional because it is based on the examination of a single iteration and that the non-corresponding iterates finally takes non-corresponding paths towards the solution, which are difficult to compare. □

5.3 Computing the NT direction

The most usual way of solving (5.2) consists in applying $\mathcal{A}$ to both sides of (5.2c) in order to eliminate d_X using (5.2b). This yields $\mathcal{A}(W d_S W) = \mathcal{A}(\mu S^{-1} - X)$. Next d_S is removed from this last equation by using (5.2a) to obtain a system in only d_y:

$$\mathcal{A}(W\mathcal{A}^*(d_y)W) = \mathcal{A}(X - \mu S^{-1}). \tag{5.22}$$

This linear system is generally solved by a direct method after computing and storing the matrix of the system. Let us look at the structure of that matrix when $\mathcal{A}$ is the $\mathbb{R}$-linear complex map specified by (3.4). Using (3.3) and (3.7a), we see that this system can be decomposed into

$$\mathcal{A}_r(W\mathcal{A}_r^*((d_y)_{\mathbb{F}_r})W) + \mathcal{A}_r(W\mathcal{A}_c^*((d_y)_{\mathbb{F}_c})W) = \mathcal{A}_r(X - \mu S^{-1}), \tag{5.23a}$$

$$\mathcal{A}_c(W\mathcal{A}_r^*((d_y)_{\mathbb{F}_r})W) + \mathcal{A}_c(W\mathcal{A}_c^*((d_y)_{\mathbb{F}_c})W) = \mathcal{A}_c(X - \mu S^{-1}). \tag{5.23b}$$

In matrix form, it becomes

$$\begin{pmatrix} \boxed{L^{11}} & L^{12} & L^{13} \\ L^{21} & \boxed{\boxed{L^{22}}} & \boxed{\boxed{L^{23}}} \end{pmatrix} \begin{pmatrix} (d_y)_{K_r} \\ (d_y)_{K_c} \\ \overline{(d_y)}_{K_c} \end{pmatrix} = \begin{pmatrix} \mathcal{A}_r(X - \mu S^{-1}) \\ \mathcal{A}_c(X - \mu S^{-1}) \end{pmatrix}, \tag{5.24}$$

where a simple box surrounds the *square real* submatrix L^{11} and double boxes surround the *square complex* submatrices L^{22} and L^{23}. The other submatrices L^{12}, L^{13}, and L^{21} have complex elements. For section 5.4, it is useful to make explicit the value of the elements of these matrices:

$$L^{11}_{kk'} = \langle G^{\mathsf{H}} A_k G, G^{\mathsf{H}} A_{k'} G\rangle_{\mathcal{H}^n} \in \mathbb{R}, \qquad L^{21}_{lk} = \langle G^{\mathsf{H}} A_l G, G^{\mathsf{H}} A_k G\rangle_{\mathbb{C}^{n\times n}} \in \mathbb{C}, \tag{5.25a}$$

$$L^{12}_{kl} = \tfrac{1}{2}\langle G^{\mathsf{H}} A_k G, G^{\mathsf{H}} A_l G\rangle_{\mathbb{C}^{n\times n}} \in \mathbb{C}, \qquad L^{13}_{kl} = \tfrac{1}{2}\langle G^{\mathsf{H}} A_k G, G^{\mathsf{H}} A_l^{\mathsf{H}} G\rangle_{\mathbb{C}^{n\times n}} \in \mathbb{C}, \tag{5.25b}$$

$$L^{22}_{ll'} = \tfrac{1}{2}\langle G^{\mathsf{H}} A_l G, G^{\mathsf{H}} A_{l'} G\rangle_{\mathbb{C}^{n\times n}} \in \mathbb{C}, \qquad L^{23}_{ll'} = \tfrac{1}{2}\langle G^{\mathsf{H}} A_l G, G^{\mathsf{H}} A_{l'}^{\mathsf{H}} G\rangle_{\mathbb{C}^{n\times n}} \in \mathbb{C}, \tag{5.25c}$$

where the indices $(k,k') \in K_r \times K_r$, $(l,l') \in K_c \times K_c$, and we have used the factorization $W = GG^{\mathsf{H}}$.

The next proposition gives properties of the matrices L^{ij}, in particular, conditions are given to ensure the nonsingularity of the matrix L^{11}. Its proof presents no difficulty if one uses the Gram structure of the matrices. Note that the $\mathbb{R}$-linearly independence of the matrices $\{A_k\}_{k\in K_r}$ used in point 1 and the $\mathbb{C}$-linearly independence of the matrices $\{A_l\}_{l\in K_c}$ used in point 3 are *weaker* assumptions than the surjectivity of $\mathcal{A}$ (see point (ii) in proposition 3.2).

Proposition 5.3 (properties of the matrices L^{ij}) 1) *The real square matrix L^{11} is symmetric and positive semidefinite. If the Hermitian matrices $\{A_k\}_{k\in K_r}$ are $\mathbb{R}$-linearly independent in $\mathcal{H}^n$, then L^{11} is positive definite.*

2) *The complex square matrix L^{22} is Hermitian and positive semidefinite. If the matrices $\{A_l\}_{l\in K_c}$ are $\mathbb{C}$-linearly independent in $\mathbb{C}^{n\times n}$, then L^{22} is positive definite.*

3) *The complex square matrix L^{23} is symmetric and $(L^{13})^{\mathsf{T}} = \frac{1}{2}\,L^{21} = (L^{12})^{\mathsf{H}}$.*

When L^{11} is nonsingular, it is natural to solve (5.24) by getting the following expression of $(d_y)_{K_r}$ from its first block row

$$(d_y)_{K_r} = -(L^{11})^{-1}\left(L^{12}(d_y)_{K_c} + L^{13}(\overline{d_y})_{K_c}\right) + (L^{11})^{-1}\left(\mathcal{A}_r(X - \mu S^{-1})\right)$$

and by substituting this vector in the second block row to obtain the system

$$Mv + N\overline{v} = p, \tag{5.26a}$$

where $v \coloneqq (d_y)_{K_c}$, $p \coloneqq \mathcal{A}_c(X - \mu S^{-1}) - L^{21}(L^{11})^{-1}(\mathcal{A}_r(X - \mu S^{-1}))$,

$$M \coloneqq L^{22} - L^{21}(L^{11})^{-1}L^{12}, \qquad \text{and} \qquad N \coloneqq L^{23} - L^{21}(L^{11})^{-1}L^{13}. \tag{5.26b}$$

The $\mathbb{R}$-linear complex system (5.26a) can certainly be solved by transforming it into a double size real system. For a computational efficiency reason, however, we prefer transforming it into a $\mathbb{C}$-linear complex system of the same dimension in the unknown variable v, hence, without the presence of its conjugate $\overline{v}$. This is possible when M is nonsingular (for necessary and sufficient conditions ensuring the possibility of doing this rewriting in the general case, see [6]), which occurs when some regularity conditions are satisfied (see proposition 5.4 below). Then, one can write $v = M^{-1}(p - N\overline{v})$ as a function of $\overline{v}$, take its conjugate $\overline{v} = \overline{M}^{-1}(\overline{p} - \overline{N}v)$, and substitute this last value in (5.26a), to get the following linear system in v:

$$\overline{M^s}v = p - N\overline{M}^{-1}\overline{p}, \tag{5.27a}$$

where

$$M^s \coloneqq \overline{M} - \overline{N}M^{-1}N. \tag{5.27b}$$

The reason for the notation M^s comes from the block elimination that has been realized, making M^s the Schur complement of M in some larger matrix. The next proposition shows that the surjectivity of $\mathcal{A}$ is a sufficient condition to have a system (5.27a) that is well defined and has a unique solution.

Proposition 5.4 (properties of the systems (5.26) and (5.27)) 1) *If the matrices $\{A_k\}_{k\in K_r} \cup \{A_l\}_{l\in K_c}$ are $(\mathbb{R}^{m_r} \times \mathbb{C}^{m_c})$-linearly independent in $\mathbb{C}^{n\times n}$, then the matrix M is well defined by (5.26b), Hermitian, and positive definite, and the matrix N is symmetric.*

2) *If $\mathcal{A}$ is surjective, then the matrix M^s is well defined by (5.27b), Hermitian, and positive definite.*

The proof is based on the block diagonalization of the Gram-like positive definite Hermitian matrices

$$\begin{pmatrix} L^{11} & 2L^{12} \\ L^{21} & 2L^{22} \end{pmatrix} \qquad \text{and} \qquad \begin{pmatrix} L^{11} & 2L^{12} & 2L^{13} \\ L^{21} & 2L^{22} & 2L^{23} \\ \overline{L^{21}} & 2\overline{L^{23}} & 2\overline{L^{22}} \end{pmatrix}.$$

5.4 Computational efforts in the real and complex algorithms

We are now ready to compare the number of operations required to perform an iteration of the interior-point algorithm 5.1 in real and complex numbers. To do so, we start by comparing the elementary operations.

Let us denote by $\mathtt{a}_{\mathbb{R}}$ and $\mathtt{a}_{\mathbb{C}}$ the computing time for the addition of two real and complex numbers, and by $\mathtt{p}_{\mathbb{R}}$ and $\mathtt{p}_{\mathbb{C}}$ the computing time for the product of two real and complex numbers. Since for a, b, c, and $d \in \mathbb{R}$, one has $(a + ib) + (c + id) = (a + c) + \mathrm{i}(b + d)$ and $(a + \mathrm{i}b)(c + \mathrm{i}d) = (ac - bd) + \mathrm{i}(ad + bc)$, there hold

$$\mathtt{a}_{\mathbb{C}} = 2\,\mathtt{a}_{\mathbb{R}} \qquad \text{and} \qquad \mathtt{p}_{\mathbb{C}} = 4\,\mathtt{p}_{\mathbb{R}} + 2\,\mathtt{a}_{\mathbb{R}}. \tag{5.28}$$

These elementary formulas allow us to estimate the computing time of a crucial operation that intervenes for computing the NT direction, namely the multiplication of matrices that are present in (5.25). This is actually the most time consuming part in our implementation as soon as the dimensions of the problem increase. The product of two $n \times n$ matrices requires the computation of n^2 elements, each of which is a scalar products of two vectors of size n. Since the latter requires n products and $n - 1$ additions, the total computing time is $n^2(n\,\mathtt{p} + (n-1)\,\mathtt{a}) \simeq n^3(\mathtt{p} + \mathtt{a})$. This results in the following estimations.

- The computing time of the product of two $n \times n$ complex (Hermitian) matrices can be evaluated by
$$\mathtt{mmp}_{\mathbb{C}}(n) \simeq n^3(\mathtt{p}_{\mathbb{C}} + \mathtt{a}_{\mathbb{C}}) = 4n^3(\mathtt{p}_{\mathbb{R}} + \mathtt{a}_{\mathbb{R}}), \tag{5.29}$$
where we have used (5.28).
- When the same matrices are transformed into real numbers, using the operator $\mathcal{J}_{\mathcal{H}^n}$ defined before (4.3), their size is doubled, so that the computing time of their product can be estimated by
$$\mathtt{mmp}_{\mathbb{R}}(n) \simeq (2n)^3(\mathtt{p}_{\mathbb{R}} + \mathtt{a}_{\mathbb{R}}) = 8n^3(\mathtt{p}_{\mathbb{R}} + \mathtt{a}_{\mathbb{R}}). \tag{5.30}$$

Therefore, one gets the ratio

$$\mathtt{mmp}_{\mathbb{R}}(n)/\mathtt{mmp}_{\mathbb{C}}(n) \simeq 2,$$

indicating that the computation is approximately twice faster in complex numbers.

In practice, the ratio $\mathtt{mmp}_{\mathbb{R}}(n)/\mathtt{mmp}_{\mathbb{C}}(n)$ may differ from the one given by this simple estimation, because the product of two matrices may depend on the ability of the function realizing this task to take advantage of the memory block size, the possible parallelization automatically realized by the compiler or interpreter, and the number of cores of a particular machine. Numerical experiment is therefore welcome. The one related in figure 5.1[1] provides a mean ratio around 1.8 for a single core, not too far from the crude estimation made above.

References

[1] F. Capitanescu (2016). Critical review of recent advances and further developments needed in AC optimal power flow. *Electric Power Systems Research*, 136, 57–68. [doi]. 2, 5

[2] F. Capitanescu, J.L. Martinez Ramos, P. Panciatici, D. Kirschen, A. Marano Marcolini, L. Platbrood, L. Wehenkel (2011). State-of-the-art, challenges, and future trends in security-constrained optimal power flow. *Electric Power Systems Research*, 81(8), 1731–1741. [doi]. 2, 5

[1]The numerical experiments of this paper have been realized in Matlab version 7.14.0.739 (R2012a), on a 2.8 GHz Intel Core i7, with OS X 10.9.5. When specified, the number of cores have been imposed from Matlab.

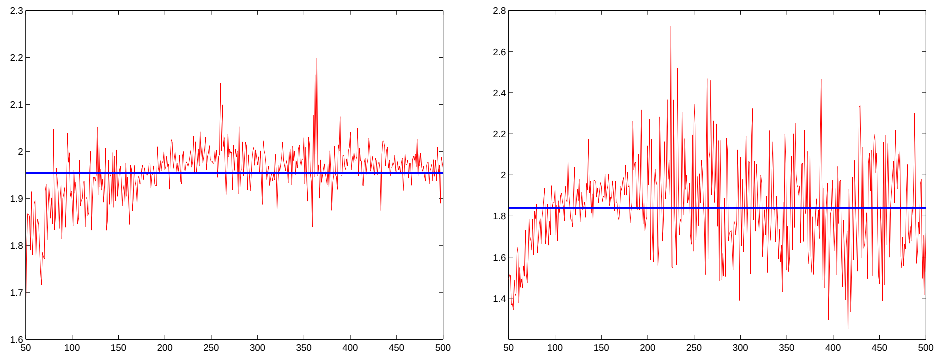

Figure 5.1: Ratio $\mathtt{mmp}_{\mathbb{R}}(n)/\mathtt{mmp}_{\mathbb{C}}(n)$ averaged on 10 runs for the product of matrices of respective order $2n$ and n, for $n \in [50\!:\!500]$. Left: one core (mean value 1.95). Right: 4 cores (mean value 1.84).

[3] M.J. Carpentier (1962). Contribution à l'étude du dispatching économique. *Bulletin de la Société Française des Électriciens*, 8(3), 431–447. 2, 5

[4] E. de Klerk (2002). *Aspects of Semidefinite Programming - Interior Point Algorithms and Selected Applications*. Kluwer Academic Publishers, Dordrecht. [doi]. 18, 19

[5] C. Ferrier (1998). Hilbert's 17th problem and best dual bounds in quadratic minimization. *Cybernetics and Systems Analysis*, 34, 696–709. [doi]. 2

[6] J.Ch. Gilbert (2016). On the equivalence between R-linear and C-linear systems of equations in complex numbers. Research report (to appear), INRIA-Paris, 2 rue Simone Iff, CS 42112, 75589 Paris Cedex 12, France. 25

[7] J.Ch. Gilbert (2016). Convergence of a feasible predictor-corrector interior-point algorithm for the semidefinite optimization problem in complex numbers. Research report (to appear), INRIA-Paris, 2 rue Simone Iff, CS 42112, 75589 Paris Cedex 12, France. 23

[8] M.X. Goemans, D.P. Williamson (2004). Approximation algorithms for MAX-3-CUT and other problems via complex semidefinite programming. *Journal of Computer and System Sciences*, 2, 442–470. [doi]. 2

[9] G.H. Golub, C.F. Van Loan (1996). *Matrix Computations* (third edition). The Johns Hopkins University Press, Baltimore, Maryland. 2, 12

[10] N.J. Higham (2002). *Accuracy and Stability of Numerical Algorithms* (second edition). SIAM Publication, Philadelphia. 2

[11] W. Huang, K. Gallivan, X. Zhang (2016). Solving PhaseLift by low-rank Riemannian optimization methods for complex semidefinite constraints. Technical report. [Optimization Online]. 2

[12] C. Josz (2016). *Application of Polynomial Optimization to Electricity Transmission Networks*. PhD Thesis, University Paris VI. 2

[13] C. Josz, S. Fliscounakis, J. Maeght, P. Panciatici (2015). AC power flow data in MATPOWER and QCQP format: iTesla, RTE snapshots, and PEGASE. Technical report. arXiv:1603.01533. 2

[14] C. Josz, J. Maeght, P. Panciatici, J.Ch. Gilbert (2015). Application of the moment-SOS approach to global optimization of the OPF problem. *IEEE Transactions on Power Systems*, 30(1), 463–470. [doi]. 2

[15] C. Josz, D.K. Molzahn (2015). Moment/sum-of-squares hierarchy for complex polynomial optimization. *SIAM Journal on Optimization* (in revision). arXiv:1508.02068. 2, 3, 10, 11

[16] M. Kojima, S. Shindoh, S. Hara (1997). Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, 7, 86–125. 19

[17] J.B. Lasserre (2001). Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11, 796–817. [doi]. 2

[18] J.B. Lasserre (2015). *An Introduction to Polynomial and Semi-Algebraic Optimization*. Cambridge Texts in Applied Mathematics. Cambridge University Press. 2

[19] J. Lavaei, S.H. Low (2012). Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, 27(1), 92–107. [doi]. 2

[20] S.H. Low (2014). Convex relaxation of optimal power flow – Part I: formulations and equivalence. *IEEE Transactions on Control of Network Systems*, 1(1), 15–27. [doi]. 2

[21] S.H. Low (2014). Convex relaxation of optimal power flow – Part II: exactness. *IEEE Transactions on Control of Network Systems*, 1(2), 177–189. [doi]. 2

[22] S. Mehrotra (1992). On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, 2, 575–601. 18

[23] S. Mizuno, M.J. Todd, Y. Ye (1993). On adaptive-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research*, 18(4), 964–981. [doi]. 18, 19

[24] D.K. Molzahn, I.A. Hiskens (2013). Moment-based relaxation of the optimal power flow problem. Technical report, University of Michigan. 2

[25] Y. Nesterov (2000). Squared functional systems and optimization problems. In J.B.G. Frenk, C. Roos, T. Terlaky, S. Zhang (editors), *High Performance Optimization*, pages 405–440. Kluwer Academic Publishers. 2

[26] Y.E. Nesterov, M.J. Todd (1997). Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22(1), 1–42. [doi]. 19

[27] P.A. Parrilo (2000). *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD Thesis, California Institute of Technology. 2

[28] R.T. Rockafellar (1970). *Convex Analysis*. Princeton Mathematics Ser. 28. Princeton University Press, Princeton, New Jersey. 17

[29] C. Roos, T. Terlaky, J.-Ph. Vial (1997). *Theory and Algorithms for Linear Optimization – An Interior Point Approach*. John Wiley & Sons, Chichester. 18, 19

[30] N.Z. Shor (1987). Class of global minimum bounds of polynomial functions. *Kibernetica*, 6, 9–11. [English translation: *Cybernetics*, 23 (1987) 731-734]. 2

[31] N.Z. Shor (1987). Quadratic optimization problems. *Soviet Journal of Computer and System Sciences*, 25, 1–11. Translated from *Izv. Akad. Nauk SSSR Tekhn. Kibernet.*, 1987, no. 1, 128-139. 2

[32] L. Sorber, M. Van Barel, L. De Lathauwer (2012). Unconstrained optimization of real functions in complex variables. *SIAM Journal on Optimization*, 22(3), 879–898. 2

[33] K.-C. Toh, M.J. Todd, R.H. Tütüncü (2012). On the implementation and usage of SDPT3 - A Matlab software package for semidefinite-quadratic-linear programming, version 4.0. In M.F. Anjos, J.B. Lasserre (editors), *Handbook on Semidefinite, Conic and Polynomial Optimization*, International Series in Operations Research and Management Science 166, pages 715–754. Springer. [doi]. 3, 10

[34] K.-C. Toh, R.H. Tütüncü, M.J. Todd (2006). On the implementation and usage of SDPT3 - a Matlab software package for semidefinite-quadratic-linear programming, version 4.0. Technical report. User's guide of SDPT3. 3

[35] I. Waldspurger, A. d'Aspremont, S. Mallat (2015). Phase recovery, maxcut and complex semidefinite programming. *Mathematical Programming*, 149(1-2), 47–81. [doi]. 2

[36] J. Watrous (2009). Semidefinite programs for completely bounded norms. *Theory of Computing*, 5(11), 217–238. [doi]. 2

[37] J. Watrous (2013). Simpler semidefinite programs for completely bounded norms. *Chicago Journal of Theoretical Computer Science*. Article 8. [doi]. 2