On numerical testing of the regularity of Semidefinite problems

Eloísa Macedo*

* *Center for Research & Development in Mathematics and Applications (CIDMA), Department of Mathematics, University of Aveiro, Portugal*
*E-mail:*macedo@ua.pt

Abstract

This paper is devoted to study regularity of Semidefinite Programming (SDP) problems. Current methods for SDP rely on assumptions of regularity such as constraint qualifications and well-posedness. Absence of regularity may compromise characterization of optimality and algorithms may present numerical difficulties. Prior that solving problems, one should evaluate the expected efficiency of algorithms. Therefore, it is important to have simple procedures that verify regularity. Here we use an algorithm to test regularity of linear SDP problems in terms of Slater's condition. We present numerical tests using problems from SDPLIB and compare our results with those from others available in literature.

Key words: Constraint qualifications, optimality conditions, regularity, semi-infinite programming, semidefinite programming, well-posedness.

1 Introduction

Semidefinite Programming (SDP) deals with problems of minimization of a linear objective function subject to constraints in the form of linear matrix inequalities and can be considered as a generalization of Linear Programming (LP), where real matrices are used instead of real variables. SDP is an active area of Optimization and has various applications in control and approximation theory, sensor network localization, principal component analysis, etc. (see [1, 18] for further details).

The methods developed to solve SDP problems, the duality theory, and most of optimality conditions are based on certain assumptions of regularity [6, 12, 18]. In the absence of regularity, characterization of optimality of feasible solutions may fail. With respect to algorithms, the regularity of a problem is a condition that guarantees their stability and efficiency [6].

In the literature, different concepts are being associated to the notion of regularity. Usually, an optimization problem is considered to be regular if certain constraint qualification (CQ) is satisfied ([9]). The most efficient optimality conditions are formulated under the fulfilment of some CQ. The regularity conditions play also an important role in deriving duality relations, sensitivity/stability analysis and convergence of computational methods [11]. One of the most stronger regularity conditions is the Slater CQ (see [4, 12]) that consists in the nonemptiness of the interior of the feasible set.

Another kind of regularity in Optimization and Numerical Methods is known as well-posedness of a problem. There exist different definitions of well-posedness. According to [5, 13], an optimization problem is well-posed in the sense of Hadamard if it has a unique solution that depends continuously on data. According to [5, 16], a problem is well-posed in the sense of Tikhonov if it has a unique solution that is stable, meaning that small perturbations of data cause small variations on solution.

In [13], it is shown that under the Slater CQ, Hadamard's well-posedness is equivalent to that of Tikhonov. In [13], other definitions of well-posedness are presented as well (Levitin-Polyak well-posedness and strong well-posedness). It is noticed, in particular, that "in finite dimensions uniqueness of the solution to a convex minimizing problem ... is enough to guarantee its Tikhonov well-posedness ... This is no longer valid in infinite dimensions ...". Theoretical study of well-posedness of certain optimization problems is a rather difficult issue. In practice, well-posedness is often verified in terms of convergence of certain minimizing sequences of some optimization problems, that is not always easy to verify.

A problem that is not well-posed is called ill-posed. Ill-posed problems are quite common in applications and, according to [10], the ill-posedness may occur due to the lack of precise mathematical formulations.

In [7, 10], a practical characterization of well-posedness of SDP problems is proposed and it is based on a so called condition number defined by Renegar in [15]. The Renegar's condition number is defined as a scale-invariant reciprocal of a problem instance to be infeasible. A SDP problem is considered to be well-posed if its Renegar condition number is finite, and ill-posed, otherwise. In [10], the calculus of this condition number is connected with upper bounds for the optimal values of SDP problems. The approach proposed in [10] for characterization of regularity of a SDP problem is constructive and is based on obtaining rigorous bounds and also error bounds for the optimal values, by properly postprocessing the output of a SDP solver. The main feature of this approach is that computation of the rigorous bounds takes into account all rounding errors and the possible small errors presented in the input data.

There exist some studies dedicated to interrelation between regularity and well-posedness of optimization problems. In [13] different notions of well-posedness of general convex problems are studied and compared; in [8, 10, 17] the relationship between well-posedness and regularity in the sense of Slater is considered for SDP problems.

In this paper, we apply a simple algorithmic procedure suggested in [14] to verify the Slater CQ for problems from SDPLIB, a linear SDP database [3]. We will compare the obtained results with tests of well-posedness described in [7, 10].

The paper is structured as follows. The basic notions are presented in Section 2. The study of different notions of regularity of linear SDP problems and the description of the algorithm to test the Slater CQ for these problems are carried out in Section 3. The numerical results are presented in Section 4. The final section is devoted to the conclusions of our work and future research.

2 Linear Semidefinite Programming Problem and its Semi-Infinite presentation

In what follows, we denote by $\mathcal{S}(s) \subset \mathbb{R}^{s \times s}$, $s \in \mathbb{N}$ the subspace of $s \times s$ real symmetric matrices. The set $\mathcal{S}(s)$ can be considered as a vector space with the trace inner product defined by

$$tr(AB) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{ji}$$

for $A, B \in \mathcal{S}(s)$. A matrix $A \in \mathcal{S}(s)$ is positive semidefinite if $x^T A x \geq 0, \forall x \in \mathbb{R}^s$. Given $A \in \mathcal{S}(s)$, to denote that A is positive (negative) semidefinite, we write: $A \succeq 0$ ($A \preceq 0$). Let $\mathcal{P}(s) \subset \mathcal{S}(s)$ be the cone of positive semidefinite symmetric $s \times s$ matrices.

Consider the following linear SDP problem:

$$\min c^T x, \quad \text{s. t. } \mathcal{A}(x) \preceq 0, \tag{1}$$

where $x \in \mathbb{R}^n$, $\mathcal{A}(x) := \sum_{i=1}^{n} A_i x_i + A_0$, $A_i \in \mathcal{S}(s)$, $i = 0, 1, ..., n$. Problem (1) is a convex problem and its (convex) feasible set is $\mathcal{X} = \{x \in \mathbb{R}^n : \mathcal{A}(x) \preceq 0\}$. We will refer to this problem as primal SDP problem.

The Lagrangian dual problem to problem (1) is given by

$$\max \; tr\,(A_0 Z)\,, \quad \text{s. t. } -tr\,(A_i Z) = c_i, \forall i = 1, \ldots, n, \quad Z \succeq 0, \tag{2}$$

where $Z \in \mathcal{P}(s)$. The feasible set of problem (2) is $\mathcal{Z} = \{Z \in \mathcal{P}(s) : -tr\,(A_i Z) = c_i, i = 1, ..., n\}$.

Notice here that some authors consider that the primal SDP problem has the form (2), and the dual problem has the form (1). This is not an issue, since it is possible to transform the problem in the form (1) to the form (2) and vice-versa (see [18]).

The duality results in SDP are more subtle than in Linear Programming (LP). The following property of LP problems still holds for SDP, inducing a lower bound on the value of the primal problem:

Theorem 1 *[Weak Duality] Given a pair of primal and dual feasible solutions $x \in \mathcal{X}, Z \in \mathcal{Z}$ of SDP problems* (1) *and* (2)*, the inequality $c^T x \geq tr(A_0 Z)$ always holds.*

Definition 1 *Denote by p^* the optimal value of the objective function of the primal SDP problem* (1) *and by d^* the optimal value of the objective function of the dual problem* (2)*. The difference $p^* - d^*$ is called duality gap.*

Unlike LP, a nonzero duality gap can occur in SDP and to guarantee strong duality some additional assumptions have to be made [1, 17].

Semi-Infinite Programming (SIP) studies optimization problems in the form

$$\min c^T x, \quad \text{s. t. } f(x,t) \le 0, t \in T,$$

where the index set T is some subset of $\mathbb{R}^s$ containing an infinite number of elements.

Consider the linear SDP problem (1). It is easy to see that this problem is equivalent to the following convex semi-infinite problem ([14]):

$$\min c^T x, \quad \text{s. t. } l^T \mathcal{A}(x) l \le 0, \forall l \in L := \{l \in \mathbb{R}^s : \|l\| = 1\}. \tag{3}$$

The feasible set of problem (3) is $\left\{x \in \mathbb{R}^n : l^T \mathcal{A}(x) l \le 0, \forall l \in L\right\}$ and it coincides with the feasible set of problem (1).

In [14], a new approach to optimality conditions for convex SIP and linear SDP was suggested. This approach is based on the notions of immobile indices for SIP problems and subspace of immobile indices for SDP.

Definition 2 *Given a convex SIP problem* (3)*, an index $l^* \in L$ is called immobile if ${l^*}^T \mathcal{A}(x) l^* = 0, \forall x \in \mathcal{X}$.*

It is proved in [14] that, given a pair of equivalent problems (1) and (3), the set of immobile indices $L^* = \left\{l \in L : l^T \mathcal{A}(x) l = 0, \forall x \in \mathcal{X}\right\}$ of problem (3) can be presented in the form $L^* = L \cap \mathcal{M}$, where $\mathcal{M}$ is a subspace of $\mathbb{R}^s$ defined by

$$\mathcal{M} := \left\{l \in \mathbb{R}^s : l^T \mathcal{A}(x) l = 0, \forall x \in \mathcal{X}\right\} = \{l \in \mathbb{R}^s : \mathcal{A}(x) l = 0, \forall x \in \mathcal{X}\}, \tag{4}$$

and is called subspace of immobile indices of the SDP problem (1).

3 Regularity of SDP Problems

3.1 Constraint Qualifications

Constraint qualifications are essential for deriving primal-dual characterization of solutions of optimization problems, and play an important role in duality theory, sensitivity and stability analysis, and in the convergence properties of computational algorithms ([11]).

The Slater CQ is one of the most widely known CQ in finite and infinite optimization that guarantees the vanishing of the dualiaty gap, therefore many authors assume in their studies that this condition holds (see [4, 8, 12, 17]).

Definition 3 *The constraints of problem* (1) *satisfy the Slater CQ if the feasible set $\mathcal{X}$ has a nonempty interior, i.e., $\exists\ \bar{x} \in \mathbb{R}^n : \mathcal{A}(\bar{x}) \prec 0$.*

Here, $A \prec 0$ ($A \succ 0$) denotes that matrix $A \in S(s)$ is negative (positive) definite. The Slater CQ is sometimes called strict feasibility [17] or Slater regularity condition [12].

The analogous definition can be introduced for the dual SDP problem.

Definition 4 *The constraints of the dual SDP problem* (2) *satisfy the Slater CQ if there exists a matrix $Z \in \mathcal{P}(s)$, such that $-tr\,(A_i Z) = c_i, \forall i = 1, \dots, n$ and $Z \succ 0$.*

In this paper, we will consider a SDP problem (1) to be regular if its constraints satisfy the Slater CQ and nonregular otherwise.

If it is assumed that problem (1) is regular, then the optimal values of problems (1) and (2) coincide and the following property is satisfied ([4, 17]):

Theorem 2 *[Strong Duality] Under the Slater CQ for the linear SDP problem* (1)*, if the primal optimal value is finite, then the duality gap vanishes and the (dual) optimal value of* (2) *is attained.*

The first order necessary and sufficient optimality conditions for regular linear SDP can be formulated in the form of the following theorem from ([2]):

Theorem 3 *If problem* (1) *satisfies the Slater CQ, then $x^* \in \mathcal{X}$ is an optimal solution if and only if there exists a matrix $Z^* \in \mathcal{P}(s)$ such that*

$$tr\,(Z^* A_i) + c_i = 0, i = 1, ..., n, \ tr\,(Z^* \mathcal{A}(x^*)) = 0. \tag{5}$$

In the absence of the Slater CQ, a duality gap in SDP can exist and conditions (5) (also called complementary conditions) may fail ([4, 14, 17]). Hence, the complete characterization of optimality of feasible solutions (either primal or dual) may fail (see the examples provided in [1, 14, 17]). Since many algorithms are based on solving systems of type (5), the failure of the strong duality can result in numerical difficulties, and thus, it is important to know in advance if the problem is regular. Notice also that unlike LP, the primal and dual SDP problems do not necessarily satisfy or not the Slater CQ simultaneously ([17]).

For the SIP problem in form (3), the Slater CQ is defined as follows.

Definition 5 *The SIP problem* (3) *satisfies the Slater CQ if there exists a feasible point* $\bar{x} \in \mathbb{R}^n$ *such that the inequalities* $l^T \mathcal{A}(\bar{x}) l < 0$ *hold, for all indices* $l \in L$.

It is easy to verify that the equivalent SDP and SIP problems (1) and (3) satisfy or not the Slater CQ simultaneously [14]. The following propositions are proved in [14].

Proposition 1 *The convex SIP problem* (3) *satisfies the Slater CQ if and only if the set* L^* *is empty.*

Proposition 2 *The SDP problem* (1) *satisfies the Slater CQ if and only if the set of immobile indices in the corresponding SIP problem* (3) *is empty.*

From Proposition 1, it follows that problem (1) is regular if and only if the subspace of immobile indices $\mathcal{M}$ defined in (4) is null, i.e., $\mathcal{M} = \{0\}$. Notice that the dimension s^* of the subspace $\mathcal{M}$ can be considered as the irregularity degree of problem (1) and

- if $s^* = 0$, then the problem is regular (satisfies the Slater CQ);
- if $s^* = 1$, then the problem is nonregular, with minimal irregularity degree;
- if $s^* = s$, then the problem is nonregular, with maximal irregularity degree.

In [14], it is shown that the subspace of immobile indices plays an important role in characterization of optimality of SDP problems and a new CQ-free optimality criterion is formulated, based on the explicit determination of the subspace $\mathcal{M}$ of immobile indices. A constructive algorithm (the DIIS algorithm) that finds a basis of the subspace $\mathcal{M}$ is described and justified in [14].

In this paper, we will use the DIIS algorithm just to check if a given SDP problem in the form (1) is regular.

3.1.1 Algorithm DIIS

Consider a linear SDP problem (1) and suppose that its feasible set is nonempty. The DIIS Algorithm proposed in [14] constructs a basis $M = (m_i, i = 1, ..., s^*)$ of the subspace of immobile indices $\mathcal{M}$. At the k-th iteration, I^k denotes some auxiliary set of indices and M^k denotes an auxiliary set of vectors. Suppose that $s > 1$, with $s \in \mathbb{N}$.

The brief description of the algorithm is as follows:

DIIS Algorithm

`input:` $s \times s$ matrices $A_j, j = 0, 1, ..., n$. Set $k := 1$, $I^1 := \emptyset$, $M^1 := \emptyset$
`repeat`
Given k, I^k , M^k:
compute $p_k := s - |I^k|$
solve the quadratic system:

$$\begin{cases} \sum_{i=1}^{p_k} l_i^T A_j l_i + \sum_{i \in I^k} \gamma_i^T A_j m_i = 0, \ j = 0, 1, \dots, n, \\ \sum_{i=1}^{p_k} \|l_i\|^2 = 1, \\ l_i^T m_j = 0 \ , \ j \in I^k, \ i = 1, \dots, \ p_k, \end{cases} \tag{6}$$

where $l_i \in \mathbb{R}^s, i = 1, ..., p_k$ and $\gamma_i \in \mathbb{R}^s, i \in I^k$
`if` system (6) does not have a solution, `then` stop and return the current M^k.
`else` given the solution $\{l_i \in \mathbb{R}^s, i = 1, \dots, p_k, \gamma_i \in \mathbb{R}^s, i \in I^k\}$ of (6):
construct the maximal subset of linearly independent vectors
$\{m_1, \dots, m_{s_k}\} \subset \{l_1, \dots, l_{p_k}\}$
update: $\triangle I^k := \{|I^k| + 1, \dots, |I^k| + s_k\}$
$M^{k+1} := M^k \cup \{m_j, \ j \in \triangle I^k\}$
$I^{k+1} := I^k \cup \triangle I^k$.
`do` $k := k + 1$

The outputs of the algorithm are:

- if the Slater CQ is satisfied: the algorithm stops at the first iteration, $k = 1$, $\mathcal{M} = \{0\}$ and $dim(\mathcal{M}) = 0$;
- if the Slater CQ is violated: the algorithm returns a basis $M = M^k$, such that $rank(M) = dim(\mathcal{M}) = s^* > 0$.

In [14], it is proved that the DIIS algorithm constructs a basis of the vector subspace $\mathcal{M}$ in a finite number of iterations.

It is easy to see that the algorithm's implementation is simple and the main numerical procedure on each iteration is solving the quadratic system (6).

3.2 Well-posedness in SDP

In [7] and [10], two constructive approaches to the classification of well-posed SDP problems are proposed, both based on the concept of well-posedness in the sense defined by Renegar in [15]. The Renegar condition number C of a problem's instance is defined as a scale-invariant reciprocal of the distance to infeasibility (the smallest data perturbation that results in either primal or dual infeasibility). If the distance to infeasibility is close to zero, then $C = \infty$ and the problem is said to be ill-posed; otherwise, it is well-posed.

The approach described in [7] is based on the estimation of lower and upper bounds of the Renegar's condition number C of a SDP problem. The distance to primal infeasibility is obtained by solving several auxiliary SDP problems of compatible size to the original SDP problem and the distance to dual infeasibility is found by solving one single SDP auxiliary problem. According to Proposition 3 in [7], the estimation of the norm of data can be done with the help of its upper and lower bounds using straightforward matrix norms and maximum eigenvalue computations. Notice that it is necessary to choose adequate norms for computing the distances to primal and dual infeasibility. The SDP problem is ill-posed if C approaches to infinity.

In [10], the characterization of well-posedness of SDP problems is based on the calculus of rigorous lower and upper bounds of their optimal values. It is shown that for the ill-posed problems (in the sense of the infinite condition number), the rigorous upper bound $\bar{p}^*$ of the primal objective function is infinite. In [10], an algorithm for computing this upper bound is described. On its iterations, some auxiliary perturbed "midpoint" SDP problems are solved using a SDP solver and special interval matrices are constructed on the basis of their solutions. The constructed interval matrices must contain a primal feasible solution of the perturbed "midpoint" problem and satisfy the conditions (4.1) and (4.2) of Theorem 4.1 from [10]. If such interval matrix can be computed, then the optimal value of the primal objective function of the SDP problem is bounded from above by $\bar{p}^*$, which is the primal objective function value considering the obtained interval matrix. Besides requiring a SDP solver for computing the approximate solutions of the perturbed problems, the approach proposed in [10] also needs verified solvers for interval linear systems and eigenvalue problems.

Therefore, we can conclude that technically, testing of regularity of SDP problems with the help of the DIIS algorithm from [14] is much easier than testing their well-posedness using the methods from [7] and [10].

3.3 Relations between regularity and well-posedness in SDP

Nevertheless the definitions of regularity and well-posedness of SDP problems introduced above are different, there exist a deep connection between them. According to [17], the lack of regularity implies ill-posedness of the problem.

The following lemma can be easily proved.

Lemma 1 *If the linear SDP problem in the form* (1) *is not regular, then it is ill-posed.*

Indeed, since the Slater CQ is not satisfied, all the feasible solutions of the problem lie on the boundary of the feasible set. Therefore, any small perturbations may lead to the loss of feasibility, meaning that the problem is ill-posed.

The reciprocal is not true. The following example shows that there exist regular problems that are ill-posed.

Example 1 *Consider the primal SDP problem*

$$\begin{array}{ll} \min & x_1 - x_2 - x_3 \\ \text{s.t.} & \begin{bmatrix} 1 & -1 & 0 \\ -1 & x_2 & 0 \\ 0 & 0 & x_3 \end{bmatrix} \succeq 0. \end{array} \tag{7}$$

This problem can be easily written in form (1)*:*

$$\begin{array}{ll} \min & x_1 - x_2 - x_3 \\ \text{s.t.} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} x_1 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} x_2 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} x_3 + \begin{bmatrix} -1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \preceq 0. \end{array}$$

The dual problem to (7) *has the form*

$$\begin{array}{ll} \max & y_1 + y_2 \\ \text{s.t.} & \begin{bmatrix} -y_2 & \frac{1+y_1}{2} & -y_3 \\ \frac{1+y_1}{2} & -1 & -y_4 \\ -y_3 & -y_4 & -1 \end{bmatrix} \succeq 0. \end{array} \tag{8}$$

The constraints of the primal problem satisfy the Slater's CQ, since there exists a strictly feasible solution: e.g., $X = \begin{bmatrix} -1 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ *and* $X \prec 0$.

However, problem (7) *is not well-posed, since its dual problem is infeasible, the distance to dual infeasibility is zero and the Renegar condition number is infinite.*

4 Regularity Tests for SDP Problems

4.1 Test Problems for SDP

For the numerical tests we have used problems from the literature and from the SDPLIB suite, a collection of 92 Linear SDP test problems, provided by Brian Borchers ([3]). This SDP data base contains problems ranging in size from 6 variables and 13 constraints up to 7000 variables and 7000 constraints. The problems are drawn from a variety of applications, such as truss topology design, control systems engineering and relaxations of combinatorial optimization problems.

4.2 Regularity Tests

To test the regularity of linear SDP problems we used the DIIS Algorithm that was implemented in Matlab 7.10.0.499 (R2010a) on a computer with an Intel Core i7-2630QM, 2.0GHz, with Windows 7 64 bits and 6 Gb RAM.

We have chosen 26 test problems from SDPLIB that were also tested in [7] and [10], in terms of their well-posedness. For these problems we checked regularity using the DIIS algorithm.

The results are displayed in Table 1. The first column of the table contains the instance's name used in SDPLIB data base; the next two columns refer to the number of variables, n, and the dimension of the constraints matrices, s. The next column represents the results of the regularity tests that find the dimension of the immobile index subspace, s^*. If $s^* = 0$, then the problem is regular. Column 5 contains the lower and upper bounds of the condition number C reported in [7] and the last column presents the upper bound for the primal objective function from [10].

Table 1: Numerical results using the DIIS Algorithm to test regularity (the SDP problem satisfies the Slater CQ if $s^* = 0$), the lower and upper bounds of the Renegar condition number from [7] and the upper bound of the optimal value from [10] (if C or $\bar{p}^*$ is finite, then the problem is well-posed).

Problem	n	s	s^*	C		$\bar{p}^*$
				lower bound	upper bound	
control1	21	15	0	8.3×10^5	1.8×10^6	-1.7782×10^1
control2	66	30	0	3.9×10^6	1.3×10^7	-8.2909×10^0
control3	136	45	0	2.0×10^6	1.2×10^7	-1.3615×10^1
hinf1	13	14	0	∞	∞	∞
hinf2	13	16	0	3.5×10^5	5.6×10^5	-7.1598×10^0
hinf3	13	16	0	∞	∞	∞
hinf4	13	16	0	∞	∞	∞
hinf5	13	16	16	∞	∞	∞
hinf6	13	16	16	∞	∞	∞
hinf7	13	16	16	∞	∞	∞
hinf8	13	16	0	∞	∞	∞
hinf9	13	16	16	2.0×10^7	3.6×10^7	∞
hinf10	21	18	0	∞	∞	∞
hinf11	31	22	0	∞	∞	∞
hinf12	43	24	0	∞	∞	∞
hinf13	57	30	30	∞	∞	∞
hinf14	73	34	0	∞	∞	∞
hinf15	91	37	37	∞	∞	∞
qap5	136	26	0	∞	∞	∞
qap6	229	37	0	∞	∞	∞
qap7	358	50	0	∞	∞	∞
qap8	529	65	0	∞	∞	∞
theta1	104	50	0	2.0×10^2	2.1×10^2	-2.3000×10^1
truss1	6	13	0	2.2×10^2	3.0×10^2	9.0000×10^0
truss3	27	31	0	7.4×10^2	1.9×10^3	9.1100×10^0
truss4	12	19	0	3.6×10^2	7.7×10^2	9.0100×10^0

Table 2: Regularity and well-posedness according to [7].

		Regularity (Slater CQ)	
		Regular	*Nonregular*
Classification	*well-posed*	8	1
	ill-posed	12	5

From Table 1 we see that 20 from 26 tested problems are regular, i.e., their constraints satisfy the Slater CQ. Moreover, the tests provide valuable information about the irregularity degree for a nonregular SDP problem. Notice that for the tested problems, all the nonregular problems in Table 1 have maximal irregularity degree.

Table 3: Regularity and well-posedness according to [10].

		Regularity (Slater CQ)	
		Regular	*Nonregular*
Classification	*well-posed*	8	0
	ill-posed	12	6

Tables 2 and 3 compare the results of testing SDP problems in terms of their regularity and well-posedness.

In Table 2, the lines correspond to well-posed and ill-posed problems according to the test from [7] and the columns correspond to regular and nonregular problems. On the intersection we have the number of problems that satisfy both corresponding conditions. Table 3 is constructed in a similar way, but the lines correspond to the number of the well-posed and ill-posed problems classified on the basis of the experiments in [10]. From Table 2 we can see that 13 from the tested problems are regular and well-posed or nonregular and ill-posed, simultaneously, and 12 of the ill-posed problems are regular. The only exception is problem *hinf9* that is nonregular and well-posed according to [7]. This contradiction to Lemma 1 can be explained by the fact that the numerical procedures are based on approximated calculus and may be not precise. Comparing our regularity results with those from [10] (w.r.t. ill-posedness), regarding Table 3 we conclude that for 14 problems these results coincide, i.e., the problems are regular and well-posed, or nonregular and ill-posed, simultaneously.

Notice that the numerical results of well-posedness obtained in [7] and in [10] do not coincide: problem *hinf9* is well-posed according to [7] and ill-posed according to [10]. This can be connected with the fact that nevertheless the condition number C is finite, it is rather big and the problem is close to be ill-posed, and it may also be due to the tests were performed in nonexact arithmetic and/or with different numerical procedures.

Finally, notice that in [10], it is reported that problem *hinf8* is well-posed, although the results presented in [10] (and also in [7]) have shown that this problem is ill-posed. Our numerical tests show that this problem is regular.

Therefore, the numerical experiences have showed that the DIIS algorithm can be efficiently used to study the regularity of SDP problems. The comparison of these tests with those from [7] and [10] confirm the conclusions about relation between regularity and well-posedness in SDP.

4.3 Conclusions and Future Work

The DIIS Algorithm permits to verify easily if a given SDP problem in the form (1) satisfies the Slater CQ. The numerical tests show that the DIIS algorithm is an efficient procedure to check the regularity of small to medium-scale SDP problems. For problems of bigger dimension the program ran out of memory. The DIIS algorithm can be used to check if a given SDP problem in form (1) is regular or not in a single iteration. The main advantage of using this numerical procedure is that one does not need to solve any SDP problem: the DIIS algorithm only deals with a quadratic system (6). To solve this system, we used in our tests the Levenberg-Marquardt method that is implemented in the routine `fsolve` from Matlab.

In our procedure, we have not checked the feasibility of the SDP problems: we worked under the assumption that their feasible sets were not empty.

To permit a more universal and precise use of the DIIS algorithm, some improvements should be made in the future. Thus, the algorithm can be modified to handle large-scale SDP problems. The procedure of solving the quadratic system (6) should be as much exact as possible, so it is reasonable to develop here specific algorithms that precisely verify the situations when the system is inconsistent. It is also important to implement the procedure that verifies feasibility of the SDP problems. Finally, it is reasonable to implement the DIIS algorithm in other programming language than Matlab (e.g., C++), since the implementation in Matlab can cause some loss of efficiency.

Notice that the tests of well-posedness of SDP problems using the procedures described in [7] and [10] present much more difficulties compared with the numerical test presented in this paper. Thus, using the method described in [7], one has to estimate the Renegar condition number that depends on the computation of three quantities that can be more or less numerically hard, depending on the choice of norms. To compute estimations of distance to primal and dual infeasibility, one has to solve several auxiliary SDP problems whose structure and size are compatible with the original SDP primal and dual problems. This may be computationally hard and not precise, since in nonregular cases, the auxiliary problems are ill-posed and the solutions found by the numerical methods may be not correct. The procedure in [10] contains parameters and to find them an auxiliary perturbed SDP problem must be solved, that turns into a difficult problem in the case of nonregular problems. To compute the rigorous upper bound, it is required to find certain interval matrices satisfying conditions of Theorem 4.1. Specially adapted solvers for interval linear systems and eigenvalue problems as well as a SDP solver for computing approximate solutions for the perturbed "midpoint" problem are needed.

On the basis of the results of the numerical experiences we can make the following conclusions: it is important to introduce a unified treatment of regularity for SDP problems and to have numerical tools to verify regularity of problems and establish clear relationship between different notions of regularity.

In the future, we intend to provide more precise and extensive regularity tests with SDP and SIP problems, and compare them with the available results of well-posedness tests.

Acknowledgements

This work was supported by *FEDER* funds through *COMPETE*–Operational Programme Factors of Competitiveness ("Programa Operacional Factores de Competitividade") and by Portuguese funds through the *Center for Research and Development in Mathematics and Applications* (University of Aveiro) and the Portuguese Foundation for Science and Technology ("FCT–Fundação para a Ciência e a Tecnologia"), within project PEst-C/MAT/UI4106/2011 with COMPETE number FCOMP-01-0124-FEDER-022690.

References

[1] Anjos, M.F. and Lasserre, J.B. (Eds.) (2012). Handbook of Semidefinite, Conic and Polynomial Optimization: Theory, Algorithms, Software and Applications. *International Series in Operational Research and Management Science*, 166, Springer.

[2] Bonnans, J. F. and Shapiro, A. (2000). Perturbation Analysis of Optimization Problems. *Springer-Verlag*, New-York.

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

[4] Cheung, Y., Schurr, S. and Wolkowicz, H. (2013). Preprocessing and Reduction for Degenerate Semidefinite Programs. Research Report CORR 2011-02, http://www.optimization-online.org/DB_FILE/2011/02/2929.pdf, Revised in January 2013.

[5] Dontchev, A. L., Zolezzi, T. (1993). Well-Posed Optimization Problems. *Lecture Notes in Mathematics, Springer-Verlag*, Berlin, 1543.

[6] Freund, R. M. (1995). Complexity of an Algorithm for Finding an Approximate Solution of a Semi-Definite Program, with no Regularity Condition. Technical Report OR 302-94, Op. Research Center, MIT, Revised in December 1995.

[7] Freund, R. M., Ordóñez, F. and Toh, K. C. (2007). Behavioral Measures and their Correlation with IPM Iteration Counts on Semi-Definite Programming Problems. *Mathematical Programming*, Springer-Verlag, New York, 109(2):445–475.

[8] Helmberg, C. (2000). Semidefinite Programming for Combinatorial Optimization. ZIB Report, Berlin.

[9] Hernández-Jiménez, B., Rojas-Medar, M.A., Osuna-Gómez, R., Beato-Moreno, A. (2009). Generalized convexity in non-regular programming problems with inequality-type constraints. Journal of Mathematical Analysis and Applications, 352(2):604–613.

[10] Jansson, C., Chaykin, D. and Keil, C. (2007). Rigorous Error Bounds for the Optimal Value in Semidefinite Programming. *SIAM Journal on Numerical Analysis archive*, 46(1):180–200.

[11] Klatte, D. (2009). First Order Constraint Qualifications. *Encyclopedia of Optimization*, Floudas, C. A. and Pardalos, P. M. (Eds.), Second Edition, Springer, 1055–1060.

[12] Klerk, E. de (2004). Aspects of Semidefinite Programming - Interior Point Algorithms and Selected Applications. *Applied Optimization*, 65, Kluwer.

[13] Konsulova, A. S. and Revalski,J. P. (1994). Constrained convex optimization problems - well-posedness and stability. *Numerical Functional Analysis and Optimization*, 15(7-8):889–907.

[14] Kostyukova, O. I. and Tchemisova, T.V. (2012). Optimality Criterion without Constraint Qualification for Linear Semidefinite Problems. Journal of Mathematical Sciences, 182(2):126–143.

[15] Renegar, J. (1994). Some Perturbation-Theory for Linear-Programming. *Mathematical Programming*, 65(1):73–91.

[16] Tikhonov, A. N. and Arsenin, V. Y. (1977). Solutions of ill-posed problems. *John Wiley and Sons.*

[17] Wolkowicz, H. (2009). Duality for semidefinite programming. *Encyclopedia of Optimization*, Floudas, C. A. and Pardalos, P. M. (Eds.), Second Edition, Springer, 811–814.

[18] Wolkowicz, H., Saigal, R. and Vandenberghe, L. (2000). Handbook of semidefinite programming: theory, algorithms, and applications. *Kluwer.*