Stabilization of Affine Polynomial Families: an LMI Approach[1]

Didier Henrion[2,3,4]
Vladimír Kučera[5]
Michael Šebek[5]

Abstract

This paper is concerned with the open issue of finding a stable element in an affine polynomial family, a difficult design problem occurring when stabilizing a given SISO plant with a fixed order compensator whose parameters are subject to interval constraints. Using a rank-one LMI formulation of the problem, a tractable linearization heuristic is developed that proves efficient in practice, as illustrated by intensive numerical experiments.

Keywords

Stability, Affine Polynomial Family, LMI.

1 Introduction

A considerable interest in the stability of families of polynomials emerged in the early eighties after the popularization of *Kharitonov's Theorem* [21] in the control engineering literature. Kharitonov showed that every polynomial in the interval polynomial family

$$\{x_0 + x_1 s + x_2 s^2 + \cdots + x_n s^n \;:\; x_i^- \leq x_i \leq x_i^+,\; i = 0, 1, \ldots, n\}$$

[1]This work was supported by the Barrande Project No. 97/005-97/026 and by the Czech Ministry of Education under project No. LN00B096.

[2]Corresponding author. E-mail: henrion@laas.fr. Fax: +33 5 61 33 69 69.

[3]Laboratoire d'Analyse et d'Architecture des Systèmes, Centre National de la Recherche Scientifique, 7 Avenue du Colonel Roche, 31 077 Toulouse, cedex 4, France.

[4]Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 182 08 Prague 8, Czech Republic.

[5]Center of Applied Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Prague 6, Czech Republic.

is stable[1] if and only if four special members of the family, called Kharitonov polynomials, are stable. The coefficients x_i of these four polynomials are fixed to x_i^- or x_i^+. As an immediate important consequence, stability of an interval polynomial family can be checked efficiently with a polynomial-time algorithm.

Many researchers attempted to extend Kharitonov's result to other polynomial families, see [3] and [6] for a good overview. The most successful generalization is probably the so-called *Edge Theorem* [4], valid for an affine polynomial family

$$\{p_0(s) + x_1p_1(s) + x_2p_2(s) + \cdots + x_np_n(s) \ : \ x_i^- \leq x_i \leq x_i^+, \ i = 1, \ldots, n\} \qquad (1)$$

where each $p_i(s)$ is a given polynomial of degree m. The Edge Theorem states that the whole affine polynomial family is stable if and only if its edges are stable, where an edge is a subset of the family for which all coefficients x_i are fixed to x_i^- or x_i^+, with the exception of one coefficient that is allowed to vary within its interval. Based on this result, it was then shown that stability of an affine polynomial family can be checked efficiently with a polynomial-time algorithm [26].

As pointed out by Blondel and Tsitsiklis in their interesting recent survey [7], the problem of deciding whether a given family of polynomials is stable is equivalent to the problem of deciding whether the family contains an unstable polynomial. This is an *analysis* problem. As previously stated, in the case of interval and affine polynomial families, this analysis problem admits a polynomial-time solution. Similarly, one may be interested in deciding whether a given family of polynomials contains a stable polynomial, which is equivalent to the problem of deciding whether the whole family is unstable. Although closely related to the former, this latter problem is much more difficult. This is a *design* problem, for which no polynomial-time algorithm is available so far. In particular, it is known that both Kharitonov's Theorem and the Edge Theorem do not hold for instability. In other words, it is not true in general that instability of all four Kharitonov polynomials of an interval polynomial family ensures instability of the whole family. Similarly, it is not true in general that instability of all edges of an affine polynomial family ensures instability of the whole family.

Motivated by these negative results, this paper is precisely concerned with the problem of finding a stable polynomial $p(s)$ in a given affine polynomial family (1).

The relevance of this problem in control engineering stems from the basic design issue of stabilizing a fixed SISO plant with a fixed order compensator whose scalar parameters are restricted to lie within given intervals. As soon as the denominator polynomial of the transfer function of the compensator is made monic, interval constraints may account for structural requirements or physical limitations on compensator coefficients. Affine polynomial family (1) consists of every achievable characteristic polynomial of the closed-loop system. Coefficients of polynomials $p_i(s)$ represent given plant parameters. Components x_i represent unknown compensator parameters. For clarity, we underline that this problem is distinct from the (even more challenging) design issue of stabilizing with a fixed compensator a family of plants whose scalar parameters are restricted to lie within given intervals [6, 28, 14]. In turn, this latter problem is dual to non-fragile design [20], where

[1] In this paper, a stable polynomial has its zeros in the open left half-plane.

a given plant must be stabilized by any fixed-order compensator whose parameters lie within given intervals.

Note however that in this paper we essentially focus on numerical methods for finding a stable element in a given affine polynomial family, without referring to the corresponding control problems mentioned above. The interested reader may consult [14, 15, 17] for applications of these methods to control design.

In Section 2 of this paper, we show that finding a stable polynomial in an affine polynomial family can be cast into a Linear Matrix Inequality (LMI) feasibility problem with a rank-one constraint, a non-convex optimization problem frequently arising in robust analysis and design problems. In Section 3, intrinsic properties of this rank-one problem are then exploited to design a tractable linearization heuristic based on successive LMI optimization problems. Global convergence of the algorithm is not guaranteed, but intensive numerical experiments are reported in Section 4 that illustrate applicability of the method. The paper winds up with some concluding remarks and directions for further research.

The notations in this paper are fairly standard. Trace$X$ and Rank$X$ respectively stand for the trace and the rank of matrix X, the notations $X \succ 0$ and $X \succeq 0$ respectively mean that matrix X is positive definite and positive semidefinite.

2 Rank-one LMI Formulation

Definition 1 *Given a pair of polynomials $p(s) = p_0 + p_1 s + \cdots + p_m s^m$ and $q(s) = q_0 + q_1 s + \cdots + q_m \mathrm{s}^m$, the Bezoutian matrix of $p(s)$ and $q(s)$ is the m-by-m symmetric matrix $B(p(s), q(s))$ with entries*

$$b_{ij} = \sum_{k=i+j-m-1}^{i-1} (p_{i+j-k-1} q_k - q_{i+j-k-1} p_k)$$

where it is assumed that $p_k = q_k = 0$ if $k < 0$ or $k > m$.

Definition 2 *Given a polynomial $p(s) = p_{\mathrm{even}}(s^2) + \mathrm{s} p_{\mathrm{odd}}(s^2)$ of degree $\leq m$ (some leading coefficients of $p(s)$ may be zero), define $p^\star(s) = p_{\mathrm{even}}(s^2) - s p_{\mathrm{odd}}(s^2)$. Denote by b_{jk} the entries of $B(p(s), p^\star(s))$, the Bezoutian matrix of $p(s)$ and $p^\star(s)$. The Hermite-Fujiwara matrix of $p(s)$ of order m is the m-by-m matrix $H_m(p(s))$ with entries*

$$h_{ij} = (-1)^i b_{ij}, \quad i, j = 1, \ldots, m.$$

A fundamental property of the above matrix is captured by the following theorem, proved in [19, §3.1].

Theorem 1 *A polynomial $p(s)$ of degree m is stable if and only if the Hermite-Fujiwara matrix $H_m(p(s))$ is positive definite.*

The following property, easily proved by inspection [15], characterizes the Hermite-Fujiwara matrix of a linear combination of polynomials.

Lemma 1 *Given a real vector* $x = [x_0 \quad x_1 \quad \cdots \quad x_n]'$ *and* $n+1$ *polynomials* $p_i(s) = p_{i_{\text{even}}}(s^2) + \mathsf{s} p_{i_{\text{odd}}}(s^2)$, $i = 0, 1, \ldots, n$ *of degrees* $\leq m$, *it holds*

$$H_m(\sum_{i=0}^{n} x_i p_i(s)) = \sum_{i=0}^{n}\sum_{j=0}^{n} x_i x_j H_m(p_{i_{\text{even}}}(s^2) + s p_{j_{\text{odd}}}(s)).$$

For notational ease, let $H_{ij} = H_m(p_{i_{\text{even}}}(s^2) + s p_{j_{\text{odd}}}(s))$. To comply with the notation introduced in (1), throughout the paper it is assumed that $x_0^- = x_0^+ = 1$. The following important result readily follows from Theorem 1 and Lemma 1.

Theorem 2 *Any vector* x *solution to the inequalities*

$$\begin{array}{l} \sum_{i=0}^{n}\sum_{j=0}^{n} x_i x_j H_{ij} \succ 0 \\ x_i^- \leq x_i \leq x_i^+, \quad i = 0, 1, \ldots, n \end{array} \tag{2}$$

corresponds to a stable element in the affine polynomial family (1).

The above matrix inequality is bilinear in components x_i. In the control literature, such inequalities are known as bilinear matrix inequalities (BMIs). In [13], the authors showed that many of the problems considered in the robust control literature can be formulated as BMIs. In view of Theorem 2, this is also the case for the problem studied in this paper. Unfortunately, BMIs are highly non-convex optimization problems and solving a general BMI was shown to be NP-hard [27].

Inequalities (2) can be formulated as a special LMI problem where all the non-convexity is concentrated into a rank constraint. The motivation for doing so is that LMI problems are well-behaved convex problems for which many numerical methods have recently been developed, see [8, 23] for a good overview. The bilinear terms in the matrix inequality are collected into the entries

$$x_{ij} = x_i x_j, \quad i, j = 0, 1, \ldots, n$$

of a rank-one positive semidefinite matrix

$$X = xx' \succeq 0$$

so that inequalities (2) become a rank-one constrained LMI system

$$\begin{array}{l} \sum_{i=0}^{n}\sum_{j=0}^{n} x_{ij} H_{ij} \succ 0 \\ X \succeq 0, \quad \text{Rank} X = 1. \end{array} \tag{3}$$

It must be underlined that the idea of formulating non-convex control problems as an LMI problem with a rank constraint is not new, see for instance [9]. Beyond the scope of systems control, rank-constrained LMI problems also occur in mathematical programming and combinatorial optimization. In [25, 23] it is shown that all of the problems with polynomial objective and polynomial constraints (including quadratic programming, binary programming, integer programming and job-shop scheduling, to mention just a few) may actually be formulated as rank-one LMI optimization problems.

3 Linearization Method

Since rank-constrained LMI problems are non-convex problems for which no efficient resolution technique is available so far, we will be satisfied with a tractable heuristic for problem (3), that it to say, a polynomial time algorithm that may often find solutions at low computational cost.

First, we formulate problem (3) as a cone complementarity problem, following an idea originally published by El Ghaoui and co-workers in [10]. Consider the optimization problem

$$\begin{array}{ll} \min & \mathrm{Trace}\mathbf{XY} \\ \mathrm{s.t.} & \sum_{i=0}^{n}\sum_{j=0}^{n} x_{ij}H_{ij} \succ 0 \\ & x_i^- \leq x_i \leq x_i^+, \quad i = 0,1,\ldots,n \\ & \mathbf{X} = \left[\begin{array}{cc} X & x \\ x' & 1 \end{array}\right] \succeq 0, \quad \mathbf{Y} = \left[\begin{array}{cc} Y & y \\ y' & z \end{array}\right] \succeq 0, \quad Y \succ 0. \end{array} \tag{4}$$

Lemma 2 *Feasibility problem (3) has a solution if and only if the global optimum of optimization problem (4) is equal to zero.*

Proof: Since both matrices $\mathbf{X}$ and $\mathbf{Y}$ are positive semidefinite, it holds $\mathrm{Trace}\mathbf{XY} = 0$ if and only if

$$\mathbf{XY} = \left[\begin{array}{cc} XY + xy' & Xy + xz \\ x'Y + y' & x'y + z \end{array}\right] = 0.$$

Therefore

$$XY + xy' = (X - xx')Y = 0.$$

Since Y is positive definite, it follows that $X = xx'$ is the expected rank-one matrix. □

As in [10], a linearization method can be pursued to solve cone complementarity problem (4). It is based on the Frank and Wolfe feasible direction method.

Algorithm 1

Step 1: Let $k = 0$. Find a feasible point $\mathbf{X}_0, \mathbf{Y}_0$ for problem (4). This is a mere LMI feasibility problem. If there is no solution, exit: there is no stable element in affine polynomial family (1).

Step 2: Find the couple $\mathbf{X}_{k+1}, \mathbf{Y}_{k+1}$ that solves the LMI optimization problem

$$\begin{array}{ll} \min & \mathrm{Trace}(\mathbf{XY}_k + \mathbf{YX}_k) \\ \mathrm{s.t.} & \sum_{i=0}^{n}\sum_{j=0}^{n} x_{ij}H_{ij} \succ 0 \\ & x_i^- \leq x \leq x_i^+, \quad i = 0,1,\ldots,n \\ & \mathbf{X} = \left[\begin{array}{cc} X & x \\ x' & 1 \end{array}\right] \succeq 0, \quad \mathbf{Y} = \left[\begin{array}{cc} Y & y \\ y' & z \end{array}\right] \succeq 0, \quad Y \succ 0. \end{array}$$

Step 3: If $\|X_{k+1} - x_{k+1}x'_{k+1}\| < \varepsilon \|X_{k+1}\|$, where ε is the given relative precision, then exit: matrix X_{k+1} is a rank-one solution to problem (3) corresponding to a stable element in affine polynomial family (1). If $k > N$, where N is the given maximum number of iterations, then exit: no rank-one solution to problem (3) has been found and we cannot conclude about the existence of a stable element in affine polynomial family (1). Otherwise, let $k = k + 1$ and go to Step 2.

As pointed out in [10], the sequence $t_k = \mathrm{Trace}(\mathbf{X}_{k+1}\mathbf{Y}_k + \mathbf{Y}_{k+1}\mathbf{X}_k)$ is bounded below by zero and decreasing, thus Algorithm 1 always converges. When t_k is equal to zero then X_k is a rank-one matrix solution to problem (3). However, there is no guarantee that Algorithm 1 converges to zero.

The first step of the minimization algorithm with concave objective $\mathrm{Trace}(X - xx')$ proposed in [2] is a special case of problem (4) where Y is the identity matrix and $y = -x/2$. The LMI relaxation algorithm studied in [16] that minimizes the convex criterion $\mathrm{Trace}X$ is also a special case of problem (4) where Y is the identity matrix and y is zero. Algorithm 1 may therefore be considered as an improvement of the above mentioned methods.

4 Numerical Experiments

In order to assess the applicability of our method we carried out intensive numerical experiments. We tested Algorithm 1 on a collection of randomly chosen affine polynomial families always containing a stable element. In the sequel, $\mathbf{i}$ is the imaginary unit and $\mathcal{R}$ denotes the set of random variables with Gaussian distribution, zero mean and unit variance.

To solve LMI problems, we used the user-friendly interface LMITOOL 2.0 for MATLAB [11] and the interior-point algorithm SDPPACK [1]. The relative precision was set to $\varepsilon = 10^{-6}$. The maximum number of iterations was set to $N = 10$. The affine polynomial families were generated by the following procedure.

Procedure 1

Step 1: Generation of a random stable polynomial $p(s)$. Let

$$p(s) = (s - z_1)(s - z_2)\cdots(s - z_m)$$

with

- $z_i = -|a_i|$ if $c_i > 0$
- $z_i = -|a_i| + \mathbf{i}b_i$ and $z_{i+1} = -|a_i| - \mathbf{i}b_i$ if $c_i \leq 0$

where $a_i, b_i, c_i \in \mathcal{R}$ for $i = 1, 2, \ldots$

Step 2: Generation of a random polynomial family $p_0(s), p_1(s), \ldots, p_n(s)$ and random components $x_1, \ldots, x_n$ such that

$$p(s) = p_0(s) + x_1 p_1(s) + \cdots + x_n p_n(s).$$

Upon equating coefficient of like powers of s in this polynomial equation, we obtain the linear system of equations

$$\underbrace{\begin{bmatrix} p^0 \\ p^1 \\ \vdots \\ p^m \end{bmatrix}}_{p} = \underbrace{\begin{bmatrix} p_0^0 & p_1^0 & \cdots & p_n^0 \\ p_0^1 & p_1^1 & \cdots & p_n^1 \\ \vdots & \vdots & & \vdots \\ p_0^m & p_1^m & \cdots & p_n^m \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_n \end{bmatrix}}_{x}.$$

Vector p is known from Step 1 and a vector x satisfying the above equation always exists if we choose matrix P such that

$$P = [p \quad R_1] R_2$$

where $R_1 \in \mathcal{R}^{(m+1)\times n}$ and $R_2 \in \mathcal{R}^{(n+1)\times(n+1)}$. Coefficients of polynomial $p_i(s)$ are retrieved from the $(i+1)$th column of matrix P.

Step 3: Generation of the affine polynomial family (1) containing $p(s)$ as a stable element. Let

$$\begin{aligned} x_i^- &= x_i - |d_i| \\ x_i^+ &= x_i + |e_i| \end{aligned}$$

where $d_i, e_i \in \mathcal{R}$ for $i = 1, 2, \ldots, n$.

4.1 A Simple Example

To illustrate Procedure 1, we generate an affine polynomial family when $m = 3$ and $n = 2$. We randomly choose

$$\begin{aligned} p(s) &= (s + 1.0823)(s + 0.2379 + 1.0078\mathbf{i})(s + 0.2379 - 1.0078\mathbf{i}) \\ &= 1.1604 + 1.5871s + 1.5581s^2 + s^3. \end{aligned}$$

In order to generate the polynomial family $p_0(s)$, $p_1(s)$, $p_2(s)$ and associated components x_1, x_2, we build the random linear system

$$\underbrace{\begin{bmatrix} 1.1604 \\ 1.5871 \\ 1.5581 \\ 1 \end{bmatrix}}_{p} = \underbrace{\begin{bmatrix} -6.3235 & -1.6406 & -0.7758 \\ -3.2054 & -2.3062 & -0.5916 \\ -7.3550 & -2.9813 & -1.0016 \\ -6.0941 & 0.9822 & -0.5438 \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} 1 \\ 0.8668 \\ -11.4791 \end{bmatrix}}_{x}.$$

Letting

$$\begin{aligned} p_0(s) &= -6.3235 - 3.2054s - 7.3550s^2 - 6.0941s^3 \\ p_1(s) &= -1.6406 - 2.3062s - 2.9813s^2 + 0.9822s^3 \\ p_2(s) &= -0.7758 - 0.5916s - 1.0016s^2 - 0.5438s^3 \end{aligned}$$

and $x_1 = 0.8668$, $x_2 = -11.4791$, it holds

$$p(s) = p_0(s) + x_1 p_1(s) + x_2 p_2(s).$$

Intervals on components x_1 and x_2 are randomly chosen as

$$\begin{array}{rclrcl} x_1^- & = & 0.8454 & x_1^+ & = & 1.8708 \\ x_2^- & = & -12.4262 & x_2^+ & = & -11.1047. \end{array}$$

Thus, we are given an affine polynomial family (1) containing the stable element $p(s)$. One can check that this family also contains an unstable extreme element

$$\begin{array}{rcl} q(s) & = & p_0(s) + x_1^+ p_1(s) + x_2^- p_2(s) \\ & = & 0.2481 - 0.1687s - 0.4864s^2 + 2.5012s^3. \end{array}$$

Now let us illustrate Algorithm 1 on the above affine polynomial family. Inequalities (2) read

$$\begin{array}{l}
\left[\begin{array}{ccc} 40.5384 & 0 & 77.0712 \\ 0 & -29.9202 & 0 \\ 77.0712 & 0 & 89.6430 \end{array}\right] + x_1 \left[\begin{array}{ccc} 39.6836 & 0 & 7.5740 \\ 0 & 45.4621 & 0 \\ 7.5740 & 0 & 21.8883 \end{array}\right] + \\
x_2 \left[\begin{array}{ccc} 12.4563 & 0 & 16.3339 \\ 0 & -1.2098 & 0 \\ 16.3339 & 0 & 20.2071 \end{array}\right] + x_1^2 \left[\begin{array}{ccc} 7.5671 & 0 & -3.2230 \\ 0 & 16.9738 & 0 \\ -3.2230 & 0 & -5.8567 \end{array}\right] + \\
x_1 x_2 \left[\begin{array}{ccc} 5.5198 & 0 & 0.2603 \\ 0 & 7.8871 & 0 \\ 0.2603 & 0 & 1.2751 \end{array}\right] + x_2^2 \left[\begin{array}{ccc} 0.9181 & 0 & 0.8439 \\ 0 & 0.3413 & 0 \\ 0.8439 & 0 & 1.0894 \end{array}\right] \succ 0 \\
0.8454 \leq x_1 \leq 1.8708, \quad -12.4262 \leq x_2 \leq -11.1047
\end{array}$$

After one iteration, Algorithm 1 converged to a rank-one solution to problem (3). It is given by

$$X = \left[\begin{array}{c} 1 \\ 0.8454 \\ -11.047 \end{array}\right] \left[\begin{array}{c} 1 \\ 0.8454 \\ -11.047 \end{array}\right]'$$

and corresponds to another stable element in the affine polynomial family, namely the extreme polynomial

$$\begin{array}{rcl} p(s) & = & p_0(s) + x_1^- p_1(s) + x_2^+ p_2(s) \\ & = & 0.9051 + 1.4151s + 1.2470s^2 + 0.7753s^3. \end{array}$$

4.2 Intensive Numerical Experiments

Then, we evaluated convergence properties of Algorithm 1 on a collection of affine polynomial families randomly generated by Procedure 1 for various values of n and m. In Table 1, we reported the number of experiments as a function of the number k of iterations required to find a rank-one solution to problem (3). We generated 100 random affine polynomial families for each couple n, m. Recall that all the families are generated so as to contain at least one stable element.

n	m	$k=1$	$2 \leq k \leq 9$	$k > 10$
1	2	97	1	2
1	5	99	1	0
1	8	99	1	0
1	11	99	1	0
2	3	93	5	2
2	5	92	8	0
2	7	100	0	0
2	9	98	2	0
3	2	64	21	15
3	5	85	9	6
3	8	94	4	2
3	11	88	10	2
4	3	61	23	16
4	5	74	13	13
4	7	73	20	7
4	9	80	13	7

n	m	$k=1$	$2 \leq k \leq 9$	$k > 10$
6	2	46	21	33
6	5	41	24	35
6	8	69	13	18
6	11	60	15	25
8	3	44	27	29
8	5	36	22	42
8	7	34	14	52
8	9	45	6	49
9	2	33	21	46
9	5	30	16	54
9	8	36	7	57
9	11	39	14	47
10	3	31	25	44
10	5	23	16	61
10	7	28	14	58
10	9	31	10	59

Table 1. Number of iterations k for various values of n and m.

Despite the fact that Algorithm 1 is a heuristic with no convergence guarantee, we can see that the rate of failures (experiments for which k exceeds the maximum allowed number of iterations $N = 10$) is kept small for low values of n. Moreover, the algorithm generally converges fast, very often in one iteration ($k = 1$). This is consistent with the results published in [10, 24] where the cone complementarity linearization algorithm was shown to be one of the best methods to date for static output feedback design.

A surprising observation here is that the speed of convergence and the rate of failures do not depend significantly on m, the number of variables involved in the rank-one LMI.

5 Conclusion

We have addressed the difficult and open problem of finding a stable element in an affine polynomial family. Using a rank-one LMI formulation of the problem, we have developed a tractable LMI linearization heuristic that proves efficient in practice. Quite surprisingly, intensive numerical experiments reveal that the speed of convergence and the rate of failures of the procedure does not depend significantly on the number of variables (i.e. with the number of polynomials in the affine polynomial family) but rather on the dimension of the rank-one LMI (i.e. with the degree of the polynomial family). A problem that has not been addressed here but that deserves particular attention is the choice of the initial feasible point at the first step of the linearization algorithm. Experience reveals that the convergence of the procedure may depend on this choice.

As illustrated by our previous work [14, 15, 16, 17], the approach is general enough to handle other challenging design problems arising in robust control. We strongly believe that further work must be devoted to the study of algorithms for solving the rank-one LMI problem. It should be insightful to assess the applicability to our problem of the various rank minimization heuristics proposed in the literature. For instance, the cutting plane technique for rank-one linear programming proposed in [25] may prove useful. Recent theoretical results on the rank minimization problem [22] also deserve to be considered. Finally, it should also be highly instructive to study deterministic global optimization techniques [12, 18, 5, 2] that can take advantage of the special structure of the rank-one LMI problem and incorporate the local technique presented in this paper.

Acknowledgment

The first author would like to thank Laurent El Ghaoui (University of California at Berkeley) for providing a proof of Lemma 2. He is also grateful to Denis Arzelier and Dimitry Peaucelle (LAAS-CNRS, Toulouse) for insightful discussions.

References

[1] F. Alizadeh, J.-P. Haeberly, M. Nayakkankuppam and M. L. Overton, "SDPPACK Version 0.8 Beta", Rutgers University, Piscataway, New Jersey, 1997.

[2] P. Apkarian and H. D. Tuan, "Robust Control via Concave Minimization: Local and Global Algorithms", *IEEE Transactions on Automatic Control*, Vol. 45, No. 2, pp. 299–305, 2000.

[3] B. R. Barmish, "New Tools for Robustness of Linear Systems", *Macmillan*, New York, 1994.

[4] A. C. Barlett, C. V. Hollot and L. Huang, "Root Locations of an Entire Polytope of Polynomials: It Suffices to Check the Edges", *Mathematics of Control, Signals and Systems*, Vol. 11, pp. 61–71, 1988.

[5] E. Beran, L. Vandenberghe and S. Boyd, "A Global BMI Algorithm based on the Generalized Benders Decomposition", *Proceedings of the European Control Conference*, EUCA, Brussels, Belgium, 1997.

[6] S. P. Bhattacharyya, H. Chapellat and L. H. Keel, "Robust Control: The Parametric Approach", *Prentice Hall*, Upper Saddle River, New Jersey, 1995.

[7] V. D. Blondel and J. N. Tsitsiklis, "A Survey of Computational Complexity Results in Systems and Control", *Automatica*, Vol. 36, No. 9, pp. 1249–1274, 2000.

[8] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory", *SIAM Studies in Applied Mathematics*, Vol. 15, Philadelphia, Pennsylvania, 1994.

[9] L. El Ghaoui and P. Gahinet, "Rank Minimization under LMI Constraints: A Framework for Output Feedback Problems", *Proceedings of the European Control Conference*, EUCA, Groningen, The Netherlands, 1993.

[10] L. El Ghaoui, F. Oustry and M. Ait Rami, "A Cone Complementarity Linearization Algorithm for Static Output Feedback and Related Problems", *IEEE Transactions on Automatic Control*, Vol. 42, No. 8, pp. 1171–1176, 1997.

[11] L. El Ghaoui and J. L. Commeau, "LMITOOL 2.0 Package: An Interface to Solve LMI Problems", *E-Letters on Systems, Control and Signal Processing*, Issue 125, January 1999.

[12] C. A. Floudas and V. Visweswaran, "A Primal-Relaxed Dual Global Optimization Approach", *Journal of Optimization Theory and Applications*, Vol. 78, No. 2, pp. 187–225, 1993.

[13] K. C. Goh, M. G. Safonov and J. H. Ly, "Robust Synthesis via Bilinear Matrix Inequalities", *International Journal of Robust and Nonlinear Control*, Vol. 6, No. 9/10, pp. 1079–1095, 1996.

[14] D. Henrion and O. Bachelier, "Low-order Robust Controller Synthesis for Interval Plants", *International Journal of Control*, Vol. 74, No. 1, pp. 1–9, 2001.

[15] D. Henrion, S. Tarbouriech and M. Šebek, "Algebraic Approach to Robust Controller Design: A Geometric Interpretation", *Proceedings of the American Control Conference*, AACC, pp. 2703–2707, Philadelphia, Pennsylvania, 1998.

[16] D. Henrion, S. Tarbouriech and D. Arzelier, "LMI Approximations for the Radius of the Intersection of Ellipsoids", *Journal of Optimization Theory and Applications*, Vol. 108, No. 1, pp. 1–28, 2001.

[17] D. Henrion, S. Tarbouriech and M. Šebek, "Rank-one LMI Approach to Simultaneous Stabilization of Linear Systems", *Systems and Control Letters*, Vol. 38, No. 2, pp. 79–89, 1999.

[18] R. Horst and H. Tuy, "Global Optimization: Deterministic Approaches", Third Revised and Enlarged Edition, *Springer Verlag*, Berlin, 1996.

[19] E. I. Jury, "Inners and Stability of Dynamic Systems", *Wiley*, New York, 1974.

[20] L. H. Keel and S. P. Bhattacharyya, "Robust, Fragile or Optimal ?", *IEEE Transactions on Automatic Control*, Vol. 42, No. 8, pp. 1098–1105, 1997.

[21] V. L. Kharitonov, "Asymptotic Stability of an Equilibrium Position of a Family of Systems of Linear Differential Equations", *Differentsial'nye Uravneniya*, Vol. 14, pp. 2086–2088, 1978.

[22] M. Mesbahi, "On the Rank Minimization Problem and its Control Applications", *Systems and Control Letters*, Vol. 33, pp. 31–36, 1998.

[23] Yu. Nesterov and A. Nemirovski, "Interior-Point Polynomial Methods in Convex Programming", *SIAM Studies in Applied Mathematics*, Vol. 13, Philadelphia, Pennsylvania, 1994.

[24] M. C. de Oliveira and J. C. Geromel, "Numerical Comparison of Output Feedback Methods", *Proceedings of the American Control Conference*, AACC, pp. 72–76, Albuquerque, New Mexico, 1997.

[25] M. V. Ramana, "An Algorithmic Analysis of Multiquadratic and Semidefinite Programming Problems", Ph. D. Thesis, The Johns Hopkins University, Baltimore, Maryland, 1993.

[26] A. Sideris, "An Efficient Algorithm for Checking the Robust Stability of a Polytope of Polynomials", *Mathematics of Control, Signals and Systems*, Vol. 4, pp. 315–337, 1991.

[27] O. Toker and H. Özbay, "On the NP-hardness of Solving Bilinear Matrix Inequalities and Simultaneous Stabilization with Static Output Feedback", *Proceedings of the American Control Conference*, AACC, pp. 2525–2526, Seattle, Washington, 1995.

[28] K. Zhou, J. C. Doyle and K. Glover, Robust and Optimal Control, *Prentice Hall*, Englewood Cliffs, New Jersey, 1996.