Polynomial Matrices, LMIs and Static Output Feedback*

Didier Henrion[1,2]
Vladimír Kučera[3]

Abstract

In the polynomial approach to systems control, the static output feedback problem can be formulated as follows: given two polynomial matrices $D(s)$ and $N(s)$, find a constant matrix K such that polynomial matrix $D(s) + KN(s)$ is stable. In this paper, we show that solving this problem amounts to solving a linear matrix inequality with a non-convex rank constraint. Several numerical experiments illustrate that the heuristic cone complementarity linearization algorithm can efficiently handle this non-convexity.

Keywords

Linear Systems, Polynomial Approach, LMI, Static Output Feedback.

1 Introduction

The static output feedback problem is one of the basic open problems in control, see the survey [Syrmos et al., 1997]. Although many techniques were developed to address the problem, no efficient algorithmic solution is available. The objective of this paper is to describe another, novel approach to the problem and a new algorithm for its solution. Similarly to already existing techniques, the algorithm is not guaranteed to provide a solution, but may prove a useful, alternative design tool.

[1]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. FAX: +33 5 61 33 69 69. E-mail: henrion@laas.fr

[2]Also with the Institute of Information Theory and Automation, Czech Academy of Sciences, Pod vodárenskou věží 4, 182 08 Praha, Czech Republic.

[3]Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, 166 27 Praha 6, Czech Republic.

*This work was supported by the Ministry of Education of the Czech Republic under Research Project MSM 212300013.

Most of the methods available to address the static output feedback problem are based on state-space arguments. In this paper, we rather follow a polynomial, or algebraic approach to systems control, whose basics are described in [Kučera, 1979].

We show that that solving the static output feedback problem amounts to solving a feasibility problem over linear matrix inequalities (LMIs, see [Boyd et al., 1994]) with an additional non-convex constraint that some matrix has some given rank. This result was already obtained in a state-space setting [El Ghaoui and Gahinet, 1993]. A contribution of our paper is in showing that, unsurprisingly, this result is also valid in a polynomial setting.

In [El Ghaoui et al., 1997], an algorithm based on cone complementarity linearization was proposed to handle the non-convex rank constraint. The algorithm is a heuristic in the sense that there is no guarantee to find a solution even when one exists. However, in [de Oliveira and Geromel, 1997] it was shown that, statistically and comparatively, this algorithm was the best existing method (from the point of view of both computational complexity and convergence properties) to design a static output feedback. Another contribution of our paper is in showing that the cone complementarity linearization algorithm can also be applied successfully to the polynomial matrix version of the static output feedback problem.

After the statement of the problem in section 2, we derive its rank-constrained LMI formulation in section 3. We show briefly in section 4 that in the special case of static state feedback, the rank constraint can be removed resulting in a convex LMI problem. In section 5 we describe the cone complementarity linearization algorithm. Implementation aspects, intensive numerical experiments and illustrating examples taken from the literature are then proposed in section 6. The paper winds up with some concluding remarks and directions for future research.

2 Problem Statement

Let $P(s) = N(s)D^{-1}(s)$ be a polynomial matrix fraction description of a multivariable plant where $D(s)$ is a given non-singular polynomial matrix and $N(s)$ is a given polynomial matrix [Kučera, 1979]. Note that $N(s)$ and $D(s)$ are not necessarily right coprime. For the system in the negative feedback configuration shown in figure 1, the static output feedback problem consists in finding a constant matrix K stabilizing the plant $P(s)$.

The matrix transfer function of the closed-loop system is the rational matrix $N(s)(D(s)+KN(s))^{-1}$. In the absence of hidden modes, stability of the system is then equivalent to stability of the denominator polynomial matrix $D(s) + KN(s)$, resulting in the following algebraic, or polynomial formulation of the static output feedback problem:

Static Output Feedback Problem: *Given a non-singular $n \times n$ polynomial matrix $D(s)$ and a $m \times n$ polynomial matrix $N(s)$, find a constant matrix K such that the*

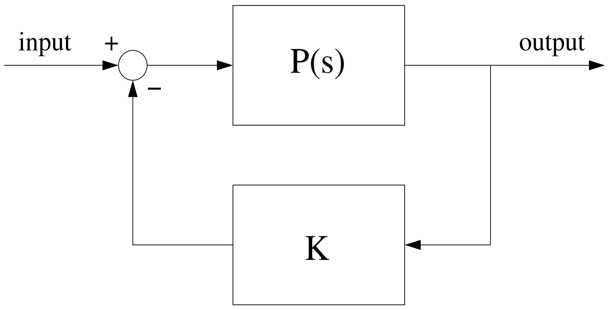

Figure 1: Plant $P(s)$ and static controller K.

polynomial matrix

$$X(s) = D(s) + KN(s) \tag{1}$$

is stable.

In this paper, we say that a polynomial matrix is stable when the zeros of its determinant belong to some region

$$\{s \in \mathbb{C} : \begin{bmatrix} 1 \\ \bar{s} \end{bmatrix}' \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} 1 \\ s \end{bmatrix} < 0\}$$

of the complex plane, where $\bar{s}$ denotes the conjugate of complex number s. Standard choices for symmetric matrix

$$H = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

are $\{a, b, c\} = \{0, 1, 0\}$ (open left-half plane, continuous-time systems) and $\{a, b, c\} = \{-1, 0, 1\}$ (open unit disc, discrete-time systems). Other choices are possible, provided matrix H has one strictly positive eigenvalue and one strictly negative eigenvalue.

Note that matrix polynomial equation (1) was extensively studied in the literature, where it is often referred to as a Diophantine polynomial equation [Kučera, 1979]. In [Kučera and Zagalak, 1991], conditions are given such that there exists a matrix K solving equation (1) when right hand-side matrix $X(s)$ is given. Similarly, a parametrization of all the solutions K to equation (1) is described in [Antsaklis and Gao, 1993] under the assumption that interpolated values of matrix $X(s)$ are available. Solutions of equation (1) are also described in the monograph [Kučera, 1991], in the special case that all the zeros of matrix $X(s)$ are located at the origin (discrete-time dead-beat control) or when matrix $X(s)$ is solution of a quadratic polynomial matrix equation (spectral factorization and linear quadratic control). In these references, the problem to be solved is referred to as the dynamics assignment problem because matrix $X(s)$ is available as input data and its invariant polynomials characterize the closed-loop system behavior. In our paper, the only information available regarding $X(s)$ is the fact that it must be stable.

In the case that matrix $X(s)$ is not given, the stabilization problem in this most general form can be solved with the help of the Youla-Kučera parametrization [Kučera, 1979], resulting then in a dynamical feedback law. Up to our knowledge, there is no way of controlling the order of the stabilizing controller with this parametrization. In particular, we cannot easily enforce the controller to be static. In the next section, we will describe the whole set of static controllers with a rank-constrained LMI rather than with the Youla-Kučera parametrization.

3 LMI with Rank Constraint

Now we show that solving the static output feedback problem in the polynomial setting introduced in section 2 amounts to solving an LMI with a non-convex rank constraint.

First we must recall the LMI formulation of the stability of a polynomial matrix, derived in [Henrion et al., 2001b]. Let $X(s) = X_0 + X_1 s + \cdots + X_d s^d$ be a non-singular polynomial matrix of size $n \times n$. Define

$$X = \left[\begin{array}{cccc} X_0 & X_1 & \cdots & X_d \end{array} \right]$$

as the constant matrix of size $n \times (d+1)n$ obtained by gathering matrix coefficients of successive powers of s. Let M denote a $(d+1)n \times n$ matrix whose columns span the null-space of X, i.e.

$$XM = 0.$$

Finally, define

$$R = \left[\begin{array}{ccccc} I_n & & & & 0 \\ & I_n & & & 0 \\ & & \ddots & & \vdots \\ & & & I_n & 0 \\ 0 & I_n & & & \\ 0 & & I_n & & \\ \vdots & & & \ddots & \\ 0 & & & & I_n \end{array} \right]$$

as a projection matrix of size $2dn \times (d+1)n$, let H be a 2×2 matrix defining a stability region as in section 2, and let $\otimes$ stand for the standard Kronecker product. The notation $P \succ 0$ (resp. $P \succeq 0$) means that matrix P is positive definite (resp. semidefinite).

Lemma 1 *Polynomial matrix $X(s)$ is stable if and only if there is a symmetric matrix P of size $dn \times dn$ solving the LMI feasibility problem*

$$M'R'(H \otimes P)RM \prec 0, \quad P \succ 0. \tag{2}$$

Proof: The complete proof is rather involved and not reproduced here for conciseness. It can be found in [Henrion et al., 2001b]. The main lines of the proof can be sketched as follows. First of all, the problem of checking stability of a polynomial matrix is formulated as a quadratic optimization problem. Second, an equivalent LMI optimization problem with a non-convex rank constraint is derived using linear fractional representations. Third, it is shown that the rank constraint is irrelevant. Fourth and last, the LMI formulation (2) is obtained via standard semi-definite programming duality arguments. □

With the help of the Sylvester matrix

$$S = \left[\begin{array}{cccc} D_0 & D_1 & \cdots & D_d \\ N_0 & N_1 & \cdots & N_d \end{array} \right]$$

of size $(n+m) \times (d+1)n$ associated with polynomial matrices $D(s) = D_0 + D_1 + \cdots + D_d s^d$ and $N(s) = N_0 + N_1 + \cdots + N_d s^d$, polynomial matrix equation (1) can be written as a linear system of equations

$$X = \left[\begin{array}{cc} I_n & K \end{array} \right] S. \tag{3}$$

With these notations, we can state the following

Theorem 1 *The static output feedback problem is solved if and only if there is a symmetric matrix*

$$Z = \left[\begin{array}{cc} Z_{11} & Z'_{21} \\ Z_{21} & Z_{22} \end{array} \right] \succeq 0$$

(partitioned such that Z_{11} has size $n \times n$ and Z_{22} has size $m \times m$) and a symmetric matrix P of size $dn \times dn$ solving the rank-constrained LMI feasibility problem

$$\begin{array}{l} F(Z,P) = R'(H \otimes P)R - S'ZS \prec 0 \\ P \succ 0, \quad Z_{11} \succ 0, \quad \mathrm{rank}\, Z = n. \end{array} \tag{4}$$

A static output feedback matrix is then $K = Z_{11}^{-1} Z'_{21}$.

Proof: If the above problem is feasible, then matrix Z can be factorized as

$$Z = \left[\begin{array}{cc} I_n & K \end{array} \right]' Z_{11} \left[\begin{array}{cc} I_n & K \end{array} \right]. \tag{5}$$

Recalling equation (3), the first inequality in LMI (4) reads $X'Z_{11}X \succ R'(H \otimes P)R$, which implies LMI (2) after projection onto the null-space of X. Invoking lemma 1, closed-loop denominator polynomial matrix $X(s)$ is stable and the static output feedback problem is solved. Conversely, if the static output feedback problem is solved, from lemma 1 there exists a symmetric positive definite P such that $M'R'(H \otimes P)RM \prec 0$. The columns of M span the null-space of UX for any non-singular U, so it follows from Finsler's theorem (see e.g. theorem 2.3.10 in [Skelton et al., 1998]) that there exists a matrix $Z_{11} = U'U \succ 0$ such that $X'Z_{11}X \succ R'(H \otimes P)R$. Equivalently, there exists a matrix Z as defined in (5) such that LMI (4) holds. $\square$

We can derive an equivalent formulation of theorem 1:

Corollary 1 *The static output feedback problem is solved if and only if there is a matrix*

$$Q = \left[\begin{array}{cc} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{array} \right]$$

(partitioned such that Q_{11} has size $n \times n$ and Q_{22} has size $dn \times m$) and a symmetric matrix P of size $dn \times dn$ solving the rank-constrained LMI feasibility problem

$$\begin{array}{l} G(Q,P) = R'(H \otimes P)R + QS + (QS)' \prec 0 \\ P \succ 0, \quad \mathrm{rank}\, Q = \mathrm{rank}\, Q_{11} = n. \end{array} \tag{6}$$

A static output feedback matrix is then $K = Q_{11}^{-1} Q_{12}$.

Proof: Recalling the proof of theorem 1, equivalence of LMI (2) and LMI (6) readily follows from theorem 2.3.12 in [Skelton et al., 1998]. From the parametrization of all candidate matrices Q given in [Skelton et al., 1998], upper-left block Q_{11} can always be chosen non-singular. $\square$

4 Static State Feedback

In this section we show that in the special case of static state feedback, solving rank-constrained LMI problem (6) amounts to solving a convex LMI problem.

Let $D(s) = A - sI_n$ and $N(s) = B$ so that the static output feedback problem boils down to the static state feedback problem, i.e. finding K such that constant matrix $A + KB$ is stable.

Lemma 2 *Let $[\alpha \quad \beta]$ be a left eigenvector corresponding to the negative eigenvalue of matrix H. Then, in the special case of static state feedback, it is necessary and sufficient to set*

$$Q = \begin{bmatrix} \beta Q_{11} & \beta Q_{12} \\ -\alpha Q_{11} & -\alpha Q_{12} \end{bmatrix}$$

in LMI problem (6), to remove the rank constraint and the constraint that P is positive definite.

Proof: As proved in [Henrion and Meinsma, 2001], stability of matrix $A + KB$ is equivalent to the LMI

$$\begin{bmatrix} I_n \\ A + KB \end{bmatrix}' (H \otimes P) \begin{bmatrix} I_n \\ A + KB \end{bmatrix} \prec 0, \quad P \succ 0.$$

From theorem 2.3.12 in [Skelton et al., 1998], the above LMI is equivalent to the two projected LMIs

$$\begin{bmatrix} W \\ AW \end{bmatrix}' (H \otimes P) \begin{bmatrix} W \\ AW \end{bmatrix} \prec 0, \quad \begin{bmatrix} \alpha I_n \\ \beta I_n \end{bmatrix}' (H \otimes P) \begin{bmatrix} \alpha I_n \\ \beta I_n \end{bmatrix} \prec 0$$

where the columns of $n \times (n - m)$ matrix W span the null-space of matrix B. Noting that

$$\begin{bmatrix} \beta I_n & -\alpha I_n \end{bmatrix} \begin{bmatrix} \alpha I_n \\ \beta I_n \end{bmatrix} = 0, \quad \begin{bmatrix} A & -I_n \\ B & 0 \end{bmatrix} \begin{bmatrix} W \\ AW \end{bmatrix} = 0,$$

and invoking once more theorem 2.3.12 in [Skelton et al., 1998], these two LMIs are in turn equivalent to the existence of two matrices Q_{11} and Q_{12} such that

$$H \otimes P + \begin{bmatrix} \beta I_n \\ -\alpha I_n \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \end{bmatrix} \begin{bmatrix} A & -I_n \\ B & 0 \end{bmatrix} + \left(\begin{bmatrix} \beta I_n \\ -\alpha I_n \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \end{bmatrix} \begin{bmatrix} A & -I_n \\ B & 0 \end{bmatrix} \right)' \prec 0.$$

Note that the constraint $P \succ 0$ is implicitely contained in the above LMI. The fact that Q_{11} can always be chosen non-singular follows from the parametrization of all candidate matrices Q_{11} and Q_{12} given in [Skelton et al., 1998]. $\square$

5 Algorithm

There exists efficient (polynomial time) algorithms to solve standard LMI problems, see [The Mathworks, 1998]. Unfortunately, the rank constraint in LMI problems (4) or (6) is non-convex and no efficient algorithm has been proposed so far to address it. This section will focus on a heuristic for handling the rank-constraint, i.e. an efficient numerical procedure that may find the solution at a low computational cost, but without guarantee to do so. In other words, it may fail sometimes, the trade-off being between the amount of computation involved and the overall failure rate. The heuristic is based on a idea proposed in [El Ghaoui et al., 1997], an extension to LMIs of a linearization technique based on complementarity.

Before exposing the algorithm, let us define the symmetric matrix

$$T = \begin{bmatrix} T_{11} & T'_{21} \\ T_{21} & T_{22} \end{bmatrix}$$

partitioned conformably with matrix Z.

Lemma 3 *Rank-constrained LMI problem (4) is feasible if and only if zero is the global optimum of the optimization problem*

$$\begin{array}{rl} \min & \operatorname{trace} TZ \\ \text{s.t.} & F(Z,P) \prec 0, \quad P \succ 0 \\ & Z \succeq 0, \quad Z_{11} \succ 0 \\ & T \succeq 0, \quad T_{22} \succ 0. \end{array} \tag{7}$$

Proof: Since both matrices T and Z are positive semi-definite, trace $TZ = 0$ implies $TZ = 0$ thus $T_{21}Z_{11} + T_{22}Z_{21} = 0$ and $T_{21}Z'_{21} + T_{22}Z_{22} = 0$. Both matrices Z_{11} and T_{22} are non-singular thus $T_{22}^{-1}T_{21} = -Z_{21}Z_{11}^{-1}$ and $Z_{22} = -T_{22}^{-1}T_{21}Z'_{21} = Z_{21}Z_{11}^{-1}Z'_{21}$. Therefore rank Z_{11} = rank $Z = n$ and rank-constrained LMI (4) follows. To show the converse implication, select a matrix Z_{11} feasible for problem (4), build Z as in (5) and let

$$T = \begin{bmatrix} K \\ -I_m \end{bmatrix} \begin{bmatrix} K \\ -I_m \end{bmatrix}'.$$

Now observe that trace $TZ = 0$ and both matrices Z and T satisfy LMI constraints (7).□

We can prove a similar result for rank-constrained LMI problem (6) if in problem (7) we replace $F(Z,P) \prec 0$ with $G(Q,P) \prec 0$, we add the LMI constraint $Q_{11} + Q'_{11} \succ 0$ and either the LMI constraint

$$V(Z,Q) = \begin{bmatrix} Z & Q' \\ Q & I_{(d+1)n} \end{bmatrix} \succeq 0$$

(where sub-matrix Z_{22} has size $m \times m$) or the LMI constraint

$$W(Z,Q) = \begin{bmatrix} Z & Q \\ Q' & I_{n+m} \end{bmatrix} \succeq 0$$

(where sub-matrix Z_{22} has size $dn \times dn$). Indeed, as soon as $Z \succeq Q'Q$ or $Z \succeq QQ'$ and Q_{11} is non-singular, rank $Z = n$ necessarily implies rank $Q =$ rank $Q_{11} = n$.

For conciseness, we will use the following acronyms to refer to the LMI feasibility sets in the rest of the paper:

- LMI **F**: $F(Z,P) \prec 0$
- LMI **GV**: $G(Q,P) \prec 0$, $V(Z,Q) \succeq 0$
- LMI **GW**: $G(Q,P) \prec 0$, $W(Z,Q) \succeq 0$

with the additional LMI constraints $P \succ 0$, $Z \succeq 0$, $Z_{11} \succ 0$, $T \succeq 0$, $T_{22} \succ 0$ and $Q_{11} + Q'_{11} \succ 0$.

In the optimization problem of lemma 3, the non-convexity is concentrated in the criterion only. The idea is then to convexify the criterion for example using the first order linearization

$$\begin{array}{rcll} \{T_{k+1}, Z_{k+1}\} & = & \arg \ \min & \mathrm{trace}(TZ_k + T_k Z) \\ & & \mathrm{s.t.} & \text{LMI } \mathbf{F},\ \mathbf{GV} \text{ or } \mathbf{GW}. \end{array} \tag{8}$$

The above problem is a convex LMI optimization problem that can be addressed efficiently. If we repeatedly solve this problem for increasing values of index k, we obtain a sequence

$$t_k = \mathrm{trace}(T_{k+1}Z_k + T_k Z_{k+1})$$

that is bounded below by zero and decreases, thus the resulting algorithm always converges [El Ghaoui et al., 1997]. However, due to the non-convexity of the problem, the sequence t_k does not necessarily converge to zero. The main difficult is then to find a suitable stopping criterion. This point is emphasized in the next section.

6 Numerical Experiments

6.1 Parameters

We wrote a macro for solving the static output feedback problem with the algorithm described in section 5. The macro will be part of the next release of the Polynomial Toolbox for MATLAB, see [Polyx, 2000]. The LMI problems are solved with the LMI Toolbox for MATLAB [The Mathworks, 1998]. There are several input parameters to the macro:

- the relative accuracy ϵ used by the LMI solver
- the maximum allowed number of iterations $k_{\max}$ of the algorithm: when $k > k_{\max}$ then we consider that the algorithm has failed;

- the minimum allowed relative variation of the criterion μ: when $t_{k+1} - t_k < \mu t_k$ then we consider that the algorithm has failed;
- the relative accuracy ν for rank evaluation: let $\sigma_1 \geq \sigma_2 \geq \cdots$ denote the singular values of matrix Z, if $\sigma_{n+1} \leq \nu \sigma_n$ then we consider that a matrix Z of rank n has been found that solves the LMI.

Practical experiments indicated that the choice $\epsilon = 10^{-8}$, $k_{\max} = 15$, $\mu = 10^{-4}$ and $\nu = 10^{-6}$ is suitable for most of the problems we had to cope with.

6.2 Scaling

In order to improve the overall numerical behavior of the cone complementarity linearization algorithm, we normalized the rows of Sylvester matrix S. We pre-multiplied S with the diagonal scaling matrix diag $\{\delta_1, \ldots, \delta_{n+m}\}$ where δ_i^{-1} is the Euclidean norm of the i-th row in S. The resulting static output feedback feedback is then the scaled matrix diag $\{\delta_1^{-1}, \ldots, \delta_n^{-1}\} K \mathrm{diag}\, \{\delta_{n+1}, \ldots, \delta_{n+m}\}$.

6.3 Complexity

The number of variables involved at each iteration of the algorithm depends on the LMI formulation of the problem. Following the nomenclature introduced in section 5, the number of decision variables in each LMI is as follows:

- LMI **F**: $N_F = n^2 d^2/2 + n^2 + m^2 + 2nm + nd/2 + n + m$
- LMI **GV**: $N_{GV} = n^2 d^2/2 + n^2 d + nmd + 2n^2 + m^2 + 3mn + nd/2 + n + m$
- LMI **GW**: $N_{GW} = 3n^2 d^2/2 + 3n^2 d + nmd + 2n^2 + nm + 3nd/2 + n$

It holds $N_F < N_{GV} \leq N_{GW}$. Therefore, in order to minimize the computational time, our implementation of the algorithm first attempts to solve the rank-constrained problem with LMI **F**, which is less time-consuming. In the case of failure, we try to solve LMI **GV**. If the algorithm still fails, then we eventually try to solve LMI **GW**.

6.4 Intensive Tests

We carried out intensive numerical experiments to evaluate the performance of the algorithm. We generated collections of randomly chosen matrices $D(s)$ (degree d, size $n \times n$) and $N(s)$ (degree d, size $m \times n$) such that we are sure that the open-loop matrix $D(s)$ is unstable and that there always exists a static output feedback matrix K such that the closed-loop matrix $D(s) + KN(s)$ is stable.

Matrices $D(s)$ and $N(s)$ were generated as follows:

1. choose K, Y_i and N_i for $i = 0, 1, \ldots, d$ as random real matrices with normally distributed entries (mean zero and variance one);
2. let $Y(s) = Y_0 + Y_1 + \cdots + Y_d s^d$;
3. extract a stable spectral factor $X(s) = X_0 + X_1 s + \cdots + X_d s^d$ of para-Hermitian polynomial matrix $Y^\star(s)Y(s) = X^\star(s)X(s)$ where the star denotes transpose conjugation (this is performed with the macro `spf` of the Polynomial Toolbox for MATLAB);
4. let $D_i = X_i - KN_i$ for $i = 0, 1, \ldots, d$;
5. if polynomial matrix $D(s) = D_0 + D_1 s + \cdots + D_d s^d$ is stable, go to the first step.

To improve conditioning of matrices P, $F(Z, P)$, $G(Q, P)$, Z_{11} or T_{22} we systematically enforced the constraint $\succeq I$ instead of the constraint $\succ 0$. Note that this is done without loss of generality because these LMIs are homogeneous in the decision variables.

For different values of d, n and m, we report in table 1 (resp. table 2) the rate of success and failure of our algorithm for 100 random continuous-time (resp. discrete-time) polynomial matrices $D(s)$ and $N(s)$.

d	n	m	Success rate	LMI **F**	LMI **GV**	LMI **GW**
2	1	1	**96**	80	9	7
2	1	2	**96**	69	13	14
2	2	1	**90**	74	12	4
2	3	2	**74**	56	11	7
2	2	3	**86**	51	12	23
3	1	1	**78**	61	8	9
3	1	2	**72**	59	6	7
3	2	1	**79**	68	9	2
3	2	3	**40**	37	3	0
3	3	2	**62**	55	5	2

Table 1: Stabilized continuous-time plants.

The tables must be interpreted as follows: for example when $d = n = m = 1$ in the continuous-time case, 96 random plants over 100 were stabilized with a static output feedback law: among them, 80 plants were stabilized with the help of the rank-constrained first LMI **F**, 9 with the second LMI **GV** (meaning that the LMI **F** failed) and 7 plants were stabilized with the third LMI **GW** (meaning that both LMIs **F** and **GV** failed). The remaining 4 plants were also stabilizable by static output feedback by construction, but the algorithm did not converge to a stabilizing matrix.

One can check that the success rates are rather high, which means that frequently the algorithm was able to find a static output feedback. Discrete-time results are significantly

d	n	m	Success rate	LMI **F**	LMI **GV**	LMI **GW**
2	1	1	**99**	87	2	10
2	1	2	**99**	75	16	8
2	2	1	**96**	82	8	6
2	2	3	**98**	79	10	9
2	3	2	**88**	63	10	15
3	1	1	**96**	71	12	13
3	1	2	**92**	62	11	19
3	2	1	**90**	70	12	8
3	2	3	**83**	46	17	20
3	3	2	**66**	33	16	17

Table 2: Stabilized discrete-time plants.

better than continuous-time results, especially in the case of medium-order systems (only 40 continuous-time plants out of 100 were stabilized in the case $d = 3$, $n = 2$, $m = 3$). We did not find any satisfying interpretation of this phenomenon.

Relatively frequently, the algorithm converged with the first LMI **F**, but sometimes it was necessary to resort to the second LMI **GV** or even to the third LMI **GW**. Of course this is done at the price of an additional computational burden.

When the algorithm did not converge, the main reason was no significant variation of the criterion. In very rare instances it did not converge because the allowed number of iterations was exceeded.

Finally, let us point out that no definitive conclusion can be drawn from these intensive testings. One must be careful when interpreting these statistics because there is no proof that the rate success does not depend on the way random polynomial matrices $D(s)$ and $N(s)$ are built.

6.5 First Example

Consider the polynomial matrices

$$D(s) = \begin{bmatrix} -s - 2s^2 & -1 - 4s - 3s^2 & 1 - 4s - 3s^3 \\ -2 - 2s - 3s^2 + 5s^3 & -1 - 5s - 3s^2 + 3s^3 & -3s - s^2 - 3s^3 \\ -2 - s - 3s^2 + 2s^3 & -1 - 2s - s^3 & -2s - s^2 - s^3 \end{bmatrix}$$

and

$$N(s) = \begin{bmatrix} -2 - s - s^2 + 2s^3 & 0 & -1 + s - s^2 \\ -s^2 + s^3 & -2s - s^2 & 1 - 2s + s^2 - s^3 \end{bmatrix}.$$

We seek a constant matrix K such that closed-loop polynomial matrix $X(s) = D(s) + KN(s)$ is stable in the discrete-time sense, i.e. all the zeros of $X(s)$ must lie within the unit disk. Note that open-loop polynomial matrix $D(s)$ is unstable.

After 3 iterations and an overall computational time of about 8 seconds on a Sun Ultra 5 workstation, the cone complementarity linearization algorithm converged to the matrices

$$Z = \left[\begin{array}{rrr|rr} 1573.5 & -768.70 & 2.9417 & -51.322 & -1048.1 \\ -768.70 & 669.00 & -16.314 & 39.683 & 431.25 \\ 2.9417 & -16.314 & 38.324 & 94.080 & 81.151 \\ \hline -51.322 & 39.683 & 94.080 & 242.24 & 229.82 \\ -1048.1 & 431.25 & 81.151 & 229.82 & 886.58 \end{array}\right]$$

and

$$P = \left[\begin{array}{rrrrrrrrr} 59.577 & 30.017 & 10.012 & 25.125 & 11.643 & 12.426 & 5.2627 & 6.8021 & 5.9512 \\ 30.017 & 49.458 & 22.774 & 36.284 & 11.632 & 24.347 & -21.933 & -3.8990 & 15.639 \\ 10.012 & 22.774 & 19.289 & 27.129 & 9.3707 & 14.986 & -3.6309 & 6.4747 & 9.3221 \\ 25.125 & 36.284 & 27.129 & 109.26 & 46.878 & 29.435 & 16.759 & 17.965 & 21.909 \\ 11.643 & 11.632 & 9.3707 & 46.878 & 46.669 & 24.975 & 31.585 & 11.239 & 20.328 \\ 12.426 & 24.347 & 14.986 & 29.435 & 24.975 & 31.568 & 13.244 & 7.0848 & 19.254 \\ 5.2627 & -21.933 & -3.6309 & 16.759 & 31.585 & 13.244 & 128.72 & 50.046 & 8.4803 \\ 6.8021 & -3.8990 & 6.4747 & 17.965 & 11.239 & 7.0848 & 50.046 & 42.308 & 8.6569 \\ 5.9512 & 15.639 & 9.3221 & 21.909 & 20.328 & 19.254 & 8.4803 & 8.6569 & 25.033 \end{array}\right]$$

satisfying LMI **F** (75 decision variables). The ratio between the 4th and 3rd singular values of Z is equal to $1.4727 \cdot 10^{-11}$, which is less than the relative accuracy $\nu = 10^{-6}$, hence matrix Z has almost rank 3. The resulting static output feedback matrix, after scaling, is given by

$$K = \left[\begin{array}{rr} 0.10377 & -1.5182 \\ 0.50552 & -0.46184 \\ 3.8385 & 3.0792 \end{array}\right].$$

In figure 2 one can check that the open-loop polynomial matrix $D(s)$ is unstable, whereas closed-loop polynomial matrix $X(s) = D(s) + KN(s)$ is stable.

6.6 Second Example

We consider the static output feedback example of [Anderson et al., 1975], a continuous-time plant described in a state-space setting

$$\begin{aligned} \dot{x} &= \left[\begin{array}{ccc} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 13 & 0 \end{array}\right] x + \left[\begin{array}{c} 0 \\ 0 \\ 1 \end{array}\right] u \\ y &= \left[\begin{array}{rrr} 0 & 5 & -1 \\ -1 & -1 & 0 \end{array}\right] x. \end{aligned}$$

whose transfer function is given by

$$P(s) = \frac{N(s)}{D(s)} = \frac{\left[\begin{array}{c} s^2 - 5s \\ s + 1 \end{array}\right]}{-s^3 + 15s}.$$

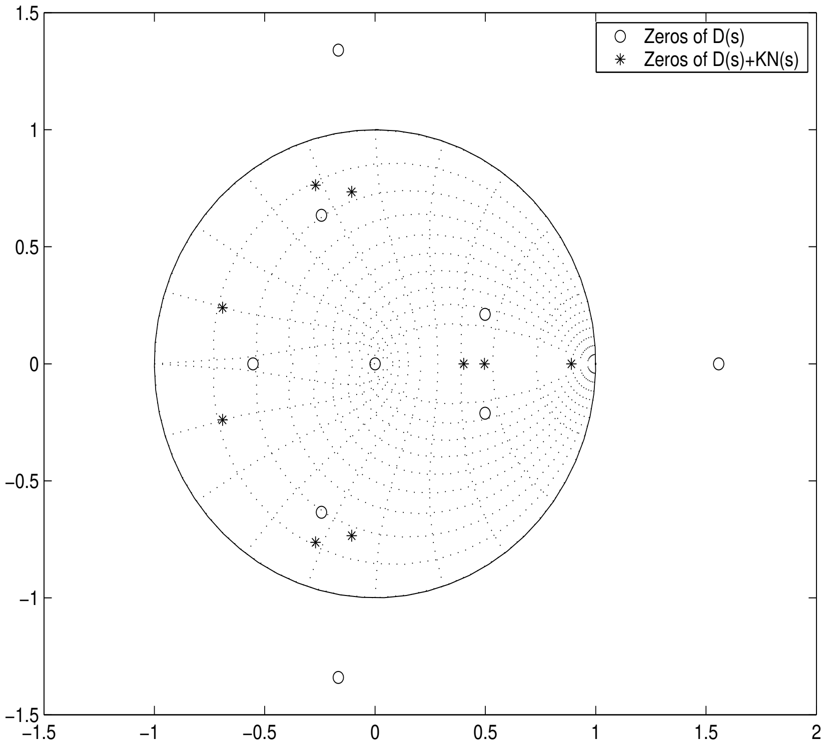

Figure 2: Root-locus of open-loop and closed-loop systems.

Our algorithm fails to find a rank-constrained solution to LMIs **F** and **GV**, but converges after 4 iterations to a solution

$$Q = \left[\begin{array}{c|cc} 2847.6 & -2882.6 & -15100 \\ \hline 2361.3 & -2390.4 & -12521 \\ 1534.8 & -1553.7 & -8138.5 \\ 6.5335 & -6.6092 & -34.620 \end{array}\right] \qquad P = \left[\begin{array}{ccc} 13903 & 4145.8 & 242.94 \\ 4145.8 & 2930.7 & 192.57 \\ 242.94 & 192.57 & 119.02 \end{array}\right]$$

that is feasible for LMI **GW**. The ratio between the second and first singular values of Q is equal to $2.0794 \cdot 10^{-7}$, which is less than the relative accuracy $\nu = 10^{-6}$, hence matrix Q is almost rank-one. The resulting static output feedback matrix, after scaling, is given by

$$K = \left[\begin{array}{cc} -2.5885 & -48.888 \end{array}\right].$$

Indeed, one can check that the closed-loop denominator polynomial

$$X(s) = D(s) + KN(s) = -s^3 - 2.5885s^2 - 22.946s - 48.888$$

has all its zeros in the left half-plane.

6.7 Third Example

Consider example 1 in [Cao et al., 1998], where the scalar continuous-time plant

$$P(s) = \frac{N(s)}{D(s)} = \frac{s+\beta}{s^2-1}$$

must be stabilized with a constant scalar gain K. It is easy to show that the plant is stabilizable if and only if $\beta > 0$. We will illustrate the behavior of our algorithm for different values of parameter β:

- When $\beta \geq 1$, the algorithm converges very fast to a rank-constrained solution to the first LMI **F** (9 decision variables). For example, when $\beta = 1$ we get $K = 2.0000$.
- When $\beta = 0.9$, no rank-constrained solution is found for the first LMI **F** (no significant relative variation of the criterion is detected after 7 iterations), but the algorithm converges after 5 iterations to a rank-constrained solution to the second LMI **GV** (15 decision variables). We get $K = 1.47356$.
- When $\beta = 0.5$, the algorithm fails with both LMIs **F** (no significant relative variation of the criterion detected after 8 iterations) and **GV** (no variation of the criterion after 5 iterations), but a rank-constrained solution is found for LMI **GW** (21 decision variables) after 2 iterations. We get $K = 3.96592$.
- When $\beta = 0.0005$, the algorithm still converges after 4 iterations to a rank-constrained solution to LMI **GW**, resulting in $K = 2325.8$.
- When $\beta < 0.0005$, the algorithm does not converge.

6.8 Fourth Example

Finally, we will show that sometimes our algorithm may fail, due to its heuristic nature. Some simple ad hoc tuning may then help convergence of the algorithm.

Consider the classical example of the nominal model of a VTOL helicopter (example 3 in [Geromel et al., 1998]), whose state-space representation is given by

$$\begin{aligned}
\dot{x} &= \begin{bmatrix} -0.0366 & 0.0271 & 0.0188 & -0.4555 \\ 0.0482 & -1.0100 & 0.0024 & -4.0208 \\ 0.1002 & 0.3681 & -0.7070 & 1.4200 \\ 0 & 0 & 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0.4422 & 0.1761 \\ 3.5450 & -7.5920 \\ -5.5200 & 4.4900 \\ 0 & 0 \end{bmatrix} u \\
y &= \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} x.
\end{aligned}$$

We consider a left polynomial matrix description $P'(s) = (N(s)D^{-1}(s))'$ of the system, given by polynomial

$$D(s) = -0.15 - 0.13s + 1.4s^2 - 3.7s^3 - 2.1s^4$$

and matrix

$$N(s) = \begin{bmatrix} -1.5 - 37s - 5.6s^2 - 7.5s^3 \\ 1.7 + 16s + 12s^2 + 16s^3 \end{bmatrix}$$

obtained with macro `ss2lmf` of the Polynomial Toolbox. With these matrices, the algorithm solves the rank-constrained LMI **F** in 3 iterations and returns the static output feedback matrix

$$K = \begin{bmatrix} -0.20335 & -0.65452 \end{bmatrix}$$

such that closed-loop denominator polynomial $D(s) + KN(s)$ is stable.

Now let us try to work on the transposed system, whose right polynomial matrix fraction description $P(s) = N_2(s)D_2^{-1}(s)$ is given by matrices

$$D_2(s) = \begin{bmatrix} -1.4 - 3.9s + 0.18s^2 & 0.17 + 2.8s + 5.9s^2 \\ -3.2 - 5s & 0.33 + 6.2s + 7.3s^2 \end{bmatrix}$$

and

$$N_2(s) = \begin{bmatrix} 23 + 0.64s & -2.1 - 34s \end{bmatrix}$$

obtained with macro `ss2rmf` of the Polynomial Toolbox. With the scaling of Sylvester matrix S described in paragraph 6.2 the cone complementarity linearization algorithm fails to find a rank-constrained solution to LMIs **F**, **GV** and **GW**. But, when the algorithm is run on the scaled matrices $D_2(s)/3, N_2(s)$ without additional scaling of Sylvester matrix S, a rank-deficient solution to LMI **GV** is found by the algorithm and a stabilizing static output feedback matrix

$$K_2 = \begin{bmatrix} -0.61004 \\ -1.9636 \end{bmatrix}$$

is returned such that closed-loop denominator matrix $D_2(s) + K_2N_2(s)$ is stable.

From this simple numerical example, we can conclude that the choice of the polynomial matrix description and the scaling may be crucial as far as convergence of the algorithm is concerned.

7 Conclusion

Providing a systematic, reliable and efficient algorithm to solve the static output feedback problem is far beyond the objective of this paper. Our main contribution is rather another, novel design method that is still of a heuristic nature but that may prove an interesting alternative to existing algorithms surveyed in [Syrmos et al., 1997] and [de Oliveira and Geromel, 1997]. Indeed, numerical experiments reported here tend to prove that the LMI-based cone complementarity algorithm proposed in [El Ghaoui et al., 1997] can solve fairly efficiently the polynomial matrix version of the static output feedback problem.

However, it is our feeling that numerical properties of the algorithm must be studied in further detail. It is well-known that the cone complementarity algorithm is prone to the zigzagging phenomena, as any other first-order approximation to a non-linear non-convex programming problem. Scaling of the decision variables can help to overcome this problem. Another line of attack could consist of using second-order information, as proposed in [Fares et al., 2001]. Convergence also somehow depends on the choice of the initial feasible point at the first step of the iterative algorithm, but in a way that seems tricky to characterize.

In section 4 we showed that the nice convexity properties of the static state feedback problem can be recovered from our rank-constrained LMI formulation of the static output feedback. Using the same arguments we guess that it should be possible for example to show that the problem of designing a controller of order equal to that of the plant is also a convex problem. It should be instructive to find out other classes of control problems leading to the same nice conclusion.

Finally, the rank-constrained LMI formulation of polynomial matrix problems can be applied to address several other central problems in control, namely low-order dynamic output feedback controller design, stabilization of polynomial matrix polytopes (as an extension of the analysis condition proposed in [Henrion et al., 2001]), or simultaneous stabilization of polynomial matrices (as an extension of the scalar results proposed in [Henrion et al., 1999]).

References

[Anderson et al., 1975] B. D. O. Anderson, N. K. Bose and E. I. Jury "Output Feedback Stabilization and Related Problems - Solution via Decision Methods", *IEEE Transactions on Automatic Control*, Vol. 20, No. 1, pp. 53–66, 1975.

[Antsaklis and Gao, 1993] P. J. Antsaklis and Z. Gao "Polynomial and Rational Matrix Interpolation: Theory and Control Applications", *International Journal of Control*, Vol. 58, No. 2, pp. 349–404, 1993.

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

[Cao et al., 1998] Y. Y. Cao, J. Lam and Y. X. Sun "Static Output Feedback Stabilization: An ILMI Approach", *Automatica*, Vol. 34, No. 12, pp. 1641–1645, 1998.

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

[El Ghaoui et al., 1997] 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.

[Fares et al., 2001] B. Fares, P. Apkarian, D. Noll "An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory", *International Journal of Control*, Vol. 74, No. 4, pp. 348–360, 2001.

[Geromel et al., 1998] J. C. Geromel, C. C. de Souza and R. E. Skelton "Static Output Feedback Controllers: Stability and Convexity", *IEEE Transactions on Automatic Control*, Vol. 43, No. 1, pp. 120–125, 1998.

[Henrion et al., 1999] 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.

[Henrion et al., 2001] D. Henrion, D. Arzelier, D. Peaucelle and M. Šebek "An LMI Condition for Robust Stability of Polynomial Matrix Polytopes", *Automatica*, Vol. 37, No. 3, pp. 461–468, 2001.

[Henrion and Meinsma, 2001] D. Henrion and G. Meinsma "Rank-one LMIs and Lyapunov's Inequality", *IEEE Transactions on Automatic Control*, Vol. 46, No. 8, 2001.

[Henrion et al., 2001b] D. Henrion, O. Bachelier and M. Šebek "D-stability of Polynomial Matrices", *International Journal of Control*, Vol. 74, No. 8. pp. 845–856, 2001.

[Kučera, 1979] V. Kučera "Discrete Linear Control: The Polynomial Approach", *John Wiley and Sons*, Chichester, 1979.

[Kučera, 1991] V. Kučera "Analysis and Design of Discrete Linear Control Systems", *Prentice Hall*, London, 1991.

[Kučera and Zagalak, 1991] V. Kučera and P. Zagalak "Constant Solutions of Polynomial Equations", *International Journal of Control*, Vol. 53, No. 2, pp. 495–502, 1991.

[The Mathworks, 1998] The MathWorks, Inc. "LMI Control Toolbox for Matlab", Release 1.0.5, 1998.

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

[Polyx, 2000] Polyx, Inc., "Polynomial Toolbox for MATLAB", see the web page `www.polyx.cz`, 2000.

[Skelton et al., 1998] R. E. Skelton, T. Iwasaki and K. Grigoriadis "A Unified Algebraic Approach to Linear Control Design", *Taylor and Francis*, London, 1998.

[Syrmos et al., 1997] V. L. Syrmos, C. T. Abdallah, P. Dorato and K. Grigoriadis "Static Output Feedback - A Survey", *Automatica*, Vol. 33, No. 2, pp. 125–137, 1997.