



HAL
open science

Polynômes et optimisation convexe en commande robuste

Didier Henrion

► **To cite this version:**

Didier Henrion. Polynômes et optimisation convexe en commande robuste. Mathématiques [math]. Université Paul Sabatier - Toulouse III, 2007. tel-00246118

HAL Id: tel-00246118

<https://theses.hal.science/tel-00246118v1>

Submitted on 7 Feb 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE PAUL SABATIER DE TOULOUSE

HABILITATION A DIRIGER DES RECHERCHES

Préparée au
Laboratoire d'Analyse et d'Architecture des Systèmes
du CNRS à Toulouse

par

Didier Henrion

Docteur de l'Institut National des Sciences Appliquées
Docteur de l'Académie Tchèque des Sciences
Chargé de Recherche au CNRS

**Polynômes et optimisation convexe
en commande robuste**

Soutenue le 17 décembre 2007

Rapporteurs : M. Hisham Abou-Kandil, Prof. ENS de Cachan
M. Laurent El Ghaoui, Prof. Univ. de Californie à Berkeley
M. Paul Van Dooren, Prof. Univ. Catholique de Louvain
Examineurs : M. Jean-Baptiste Hiriart-Urruty, Prof. Univ. Toulouse
M. Jean-Bernard Lasserre, Dir. Rech. LAAS-CNRS Toulouse
M. Emmanuel Trélat, Prof. Univ. Orléans
Directrice de recherche : Me. Sophie Tarbouriech, Dir. Rech. LAAS-CNRS Toulouse

Table des matières

1	Introduction	7
2	Convexité, non-convexité et optimisation LMI	11
2.1	Convexité et LMI	12
2.2	Non-convexité et PMI	18
2.3	Convexité cachée	19
2.4	Relaxations LMI	20
2.5	Conclusions	25
3	Polynômes en commande robuste	27
3.1	Systèmes linéaires et polynômes	27
3.2	Stabilité des polynômes	29
3.3	Stabilisation de polynômes	33
3.4	Polynôme central	37
3.5	Géométrie des mineurs de Hurwitz	40
3.6	Critère de Hermite et relaxations LMI	42
3.7	Conclusions	45
4	Méthodes numériques et outils logiciels	47
4.1	Polynomial Toolbox	47
4.2	Solveurs LMI et BMI	48
4.3	GloptiPoly	49
4.4	HIFOO	50
5	Perspectives	53
5.1	Développements de logiciels	53
5.2	Conditionnement et stabilité numérique	55
5.3	LMI et géométrie algébrique pour l'automatique	56

Remerciements

Je dédie ce travail à ma femme Jana et à mon père Gérard, pour leur tolérance, leur patience et leur soutien.

Mes pensées vont également à ma fille Nella, à mon regretté collègue et ami Filipe Devy-Vareta, aux Professeurs Germain Garcia, Bernard Pradin et André Titli dont les enseignements de qualité m'ont aidé à choisir ma voie, et à mes amis de promotion Olivier Bachelier et Jamal Daafouz.

Je remercie également mes collègues Denis Arzelier, Jacques Bernussou, Jean-Baptiste Hiriart-Urruty, Jean-Bernard Lasserre, Dimitri Peaucelle, Isabelle Queinnec et Sophie Tarbouriech (à Toulouse), Sergej Čelikovský, Martin Hromčík, Zdeněk Hurák, Michal Kočvara, Vladimír Kučera, Michael Šebek et Miroslava Součková (à Prague), pour leur aide et leur soutien.

Chapitre 1

Introduction

Les principales avancées en commande des systèmes linéaires multivariables ont eu lieu dans les années 1960 et 1970, principalement suite aux travaux fondateurs de Kalman. En Europe de l'Ouest et en Amérique du Nord, ces résultats ont fait appel au formalisme de l'espace d'état, par opposition aux techniques algébriques ou polynomiales plus en vogue en Europe Centrale et Orientale. Pour un survol des points communs et des spécificités de ces deux approches, on pourra par exemple consulter [42].

Après une relative accalmie dans les années 1980, la recherche sur les systèmes linéaires a été relancée dans le contexte de la commande robuste [64, 68]. Outre un formalisme mathématique rigoureux, un dénominateur commun de ces travaux a été le souci d'applicabilité et d'implémentabilité sur ordinateur des techniques développées. En 1982, un numéro spécial de la revue Américaine IEEE Control Systems Magazine a été dédié à l'automatique assistée par ordinateur (CACSD pour computer-aided control system design), et un comité technique dédié a vu le jour [31]. A la même époque, on a pu assister au développement du logiciel de calcul scientifique Matlab sur la base d'outils d'analyse numérique permettant la résolution fiable de problèmes de valeurs propres et d'équations algébriques matricielles (Lya-punov, Riccati). Un exemple caractéristique est la synthèse de correcteurs optimaux H_∞ à l'aide d'équations algébriques de Riccati couplées [16], implementée dans la fonction `hinfsyn` de la Control System Toolbox de Matlab [49].

Parallèlement, à la fin des années 1980, des progrès théoriques considérables ont été effectués dans le domaine de la programmation mathématique. En particulier, suite aux travaux fondateurs de Nesterov et Nemirovski [50], les méthodes de points intérieurs se sont imposées pour la résolution de problèmes d'optimisation convexes mais non-linéaires. Concernant l'ap-

plication de ces techniques en automatique, les travaux de Boyd ont été précurseurs en la matière, tout d'abord dans un contexte de programmation linéaire [10], et ensuite dans le cadre de la programmation semi-définie et des inégalités matricielles linéaires, ou LMI [11]. Ainsi, l'approche LMI pour la synthèse de correcteurs optimaux H_∞ est également implémentée dans la fonction `hinfsyn` de la Control System Toolbox de Matlab, comme alternative à l'approche par équations algébriques de Riccati.

Dans les années 1990, la recherche en commande robuste des systèmes linéaires a combiné ainsi des techniques d'algèbre linéaire numérique et d'optimisation convexe. Pour des raisons essentiellement historiques et économiques, ces études se sont effectuées dans un cadre d'espace d'état, et la plupart des logiciels de CACSD n'utilisent pas le formalisme polynomial. Ainsi, la fonction `hinfsyn` convertit automatiquement les modèles polynomiaux dans l'espace d'état avant d'utiliser les algorithmes décrits ci-dessus. Pourtant, une théorie solide a été développée pour la commande H_∞ polynomiale, sur la base d'équations Diophantiennes et de factorisations spectrales polynomiales [43].

A la fin des années 1990, les travaux en commande robuste linéaire ont connu un certain essoufflement. En effet, les problèmes faciles résolus assez rapidement ont laissé la place à des problèmes fondamentalement beaucoup plus difficiles, et parfois même à des résultats négatifs. Par exemple, il a été démontré en 1994 dans la thèse de doctorat de Blondel [7] que le problème de la synthèse d'un correcteur linéaire unique stabilisant simultanément trois systèmes linéaires donnés, s'il est élémentaire à formuler, est un problème rationnellement indécidable. En d'autres termes, il n'est pas possible à l'heure actuelle de disposer d'un algorithme permettant de résoudre le problème à l'aide d'opérations fondamentales comme l'addition ou la multiplication. Un autre exemple de problème fondamental ouvert est le problème de retour de sortie statique, dont on ne connaît même la complexité [8], même si l'on sait qu'il est rationnellement décidable puisqu'équivalent à la recherche d'un point dans un ensemble semi-algébrique. Ainsi, il n'existe pas de fonction Matlab permettant de synthétiser un correcteur H_∞ optimal par retour de sortie de statique. La fonction `hinfsyn`, mentionnée à trois reprises ci-dessus, permet uniquement d'obtenir des correcteurs d'ordre égal à celui du système en boucle ouverte, une restriction qui peut s'avérer fondamentale lorsqu'on désire implémenter de manière simple et économique une loi de commande sur un correcteur embarqué.

Mes activités de recherche s'inscrivent dans la continuation des travaux fondamentaux en automatique des systèmes linéaires décrits brièvement ci-dessus, et proposent d'explorer quelques voies qui, à défaut d'être originales, semblent avoir été négligées jusqu'alors. Dans le cadre de mon projet d'intégration au CNRS déposé en 2000, j'ai ainsi tenté de combiner les méthodes polynomiales et l'optimisation convexe LMI dans le but de développer des outils numériques fiables et efficaces pour la résolution de problèmes fondamentaux en automatique des systèmes linéaires, et en particulier pour la commande robuste.

Ce mémoire est organisé en conséquence, avec tout d'abord un chapitre 2 consacré aux liens étroits existant entre convexité, non-convexité et optimisation LMI. Le chapitre 3 applique ces notions dans le cadre de l'approche polynomiale à la commande des systèmes linéaires. Sur la base des acquis théoriques mentionnés, diverses méthodes numériques et outils de CACSD sont ensuite décrits dans le chapitre 4, où l'on trouvera également une description du contexte historique dans lequel j'ai pu mener mes travaux de recherche. Le rapport s'achève avec une évocation dans le chapitre 5 des thèmes de recherche qui me concernent à l'heure actuelle.

Chapitre 2

Convexité, non-convexité et optimisation LMI

En 1993, Rockafellar affirmait : “The great watershed in optimization is not between linearity and nonlinearity, but convexity and nonconvexity” [61]. L’objectif de ce chapitre est de replacer cette phrase dans un contexte actuel, et relativement aux récentes avancées dans le domaine de l’optimisation convexe et de la géométrie algébrique.

Il est incontestable que la *convexité* est une propriété très souhaitable pour la résolution numérique d’un problème d’optimisation ou de décision. Les ouvrages de référence [5, 12] en sont de très belles illustrations. Par contre, la convexité n’est pas une propriété inhérente à un problème, c’est une propriété relative à une certaine *formulation* de ce problème. Ainsi beaucoup de problèmes que l’on peut supposer non-convexes admettent en fait une reformulation convexe, et dans le chapitre 3 nous en verrons les implications pour la commande des systèmes linéaires.

Dans le paragraphe 2.1 nous rappelons brièvement les notions fondamentales relatives aux *inégalités matricielles linéaires* (LMI), à la programmation semi-définie et aux propriétés de convexité associées. Par la suite, nous introduisons dans le paragraphe 2.2 les *inégalités matricielles polynomiales* (PMI) qui définissent généralement des domaines non-convexes. Dans les deux derniers paragraphes 2.3 et 2.4 nous montrons qu’en fait la séparation entre convexité et non-convexité n’est pas si nette, à la lumière de résultats récents en programmation semi-définie et géométrie algébrique.

2.1 Convexité et LMI

Nous considérons des problèmes d'optimisation conique, où le cône est convexe et symétrique. La théorie de la classification conique affirme que tout cône symétrique est le produit direct des cônes suivants : l'orthant positif, le cône quadratique (dit de Lorentz), le cône des matrices réelles positives semi-définies (PSD) que l'on dénommera cône PSD par simplicité, le cône des matrices complexes Hermitiennes PSD, le cône des matrices de quaternions PSD, et le cône exceptionnel de dimension 27 des matrices d'octonions PSD. Les deux premiers cônes sont inclus dans le cône semi-défini, et les trois derniers sont des sections de dimensions supérieures de ce même cône. Dans un contexte de programmation mathématique, le cône PSD semble donc être d'une généralité suffisante.

Le cône PSD est l'ensemble des matrices réelles symétriques à valeurs propres non-négatives. L'appartenance d'une matrice symétrique A à ce cône s'écrit

$$A \succeq 0.$$

L'appartenance à l'intérieur du cône, où toutes les valeurs propres de A sont strictement positives, s'écrit

$$A \succ 0.$$

En utilisant les formes quadratiques, il est facile de démontrer que le cône PSD est *convexe*.

Par abus de langage, l'intersection entre ce cône et un sous-espace affine est appelée inégalité matricielle linéaire, ou LMI (Linear Matrix Inequality)

$$A(x) = A_0 + \sum_{i=1}^n x_i A_i \succeq 0.$$

Le domaine LMI ainsi défini

$$\{x \in \mathbb{R}^n : A(x) \succeq 0\}$$

est un ensemble fermé convexe de \mathbb{R}^n .

Il existe un grand nombre d'ensembles convexes qui peuvent être représentés de cette manière. Une classification non-exhaustive est proposée dans [50] et [5]. Tout d'abord, il faut noter que les ensembles LMI sont semi-algébriques basiques [4], c'est-à-dire qu'ils s'expriment comme

$$\{x \in \mathbb{R}^n : f_j(x) \geq 0, j = 1, \dots, p\}$$

où les $f_j(x)$ sont des polynômes scalaires de la variable vectorielle x . Pour s'en convaincre, il suffit de construire comme dans [5, §4.10.1] les $f_j(x)$ comme mineurs principaux de $A(x)$, ou encore d'utiliser la règle des signes de Descartes [4, §2.2] comme décrit dans [29]. La question se pose alors de savoir quels ensembles semi-algébriques basiques convexes sont LMI. Une réponse constructive consisterait à déterminer les matrices A_i à partir des polynômes $f_j(x)$. Il semble que ce problème soit très difficile dans le cas général.

Une solution au problème de représentation LMI a récemment été proposée par Helton et Vinnikov [30] dans le cas particulier $n = 2$ et $p = 1$, c'est-à-dire pour les courbes algébriques planes. Etant donné un polynôme $f(x)$ irréductible (c'est-à-dire non-factorisable) tel que $f(0) = 1$ (normalisation sans perte de généralité), une condition nécessaire et suffisante d'existence d'une représentation LMI du domaine incluant l'origine

$$\{x \in \mathbb{R}^2 : f(x) \geq 0\} \ni 0$$

est que toute droite passant par l'origine coupe génériquement la courbe algébrique $f(x) = 0$ un nombre de fois égal au degré de $f(x)$, dénoté d . Dans la cas où la condition, dite de *rigidité convexe*, est satisfaite, alors il existe des matrices symétriques A_1 et A_2 de dimension d telles que

$$f(x) = \det A(x) = \det (I_d + x_1 A_1 + x_2 A_2).$$

Le polynôme $f(x)$ admet alors une représentation déterminantale symétrique monique, à savoir $A(0) = I_d$, la matrice identité. Le domaine LMI correspond alors à la composante connexe convexe incluant l'origine. La preuve de cette condition d'existence, qui fait appel à des résultats de géométrie complexe Riemannienne, permet en principe la construction numérique des matrices A_1 et A_2 à partir des coefficients du polynôme $f(x)$ [67]. Il existe également une procédure, due à Dixon [15], qui permet une construction par l'algèbre linéaire de ces matrices, à partir de la connaissance d'une certaine courbe de contact. La dérivation de cette courbe de contact semble cependant délicate dans le cas général.

A titre d'illustration, voici la représentation déterminantale symétrique monique d'une courbe elliptique, que l'on peut obtenir systématiquement à l'aide du Hessien (merci à Frédéric Han et Jean Vallès pour leur aide) :

$$f(x) = 1 - 2x_1 - x_1^2 - x_2^2 + 2x_1^3 = \det A(x) = \det \begin{bmatrix} 1 & x_1 & 0 \\ x_1 & 1 & x_2 \\ 0 & x_2 & 1 - 2x_1 \end{bmatrix}.$$

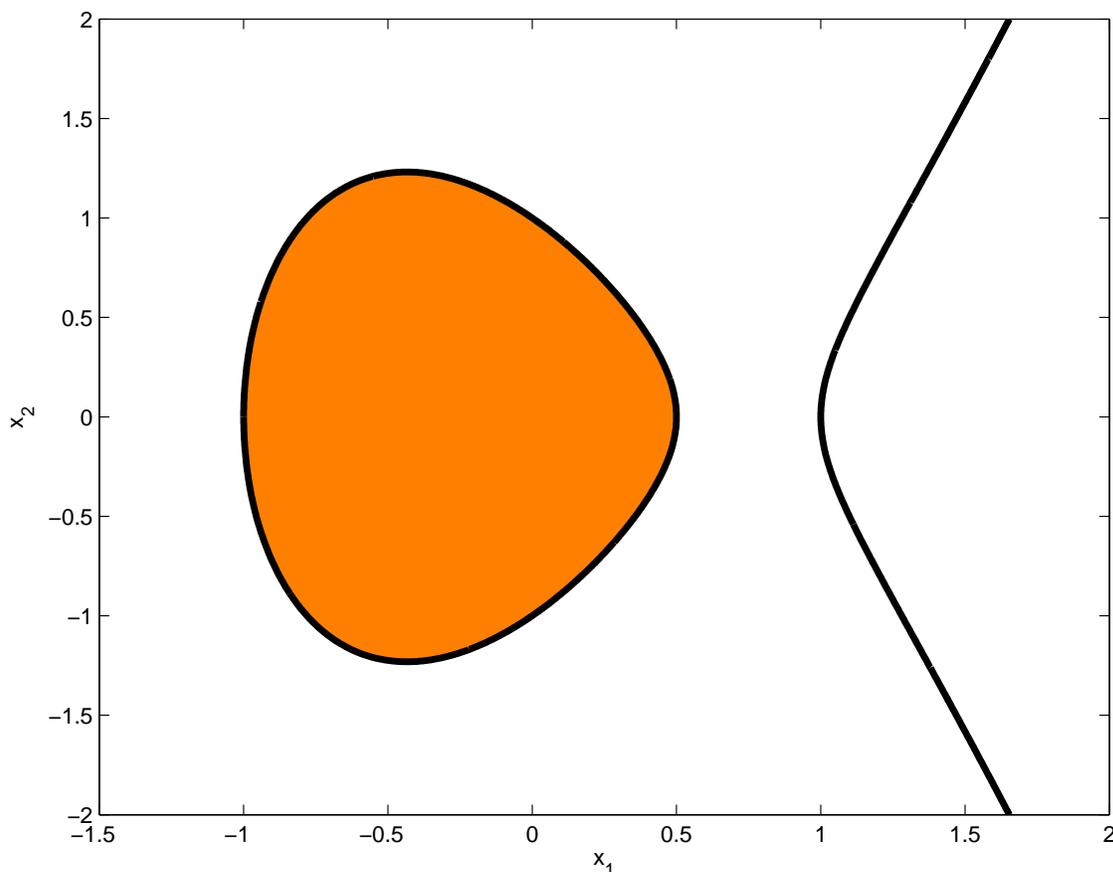


FIG. 2.1 – Domaine LMI associé à une composante connexe convexe délimitée par une courbe elliptique plane.

La courbe algébrique ainsi que le domaine LMI correspondant sont représentés sur la figure 2.1.

Dans le cas $n > 2$ il ne semble pas avoir de généralisation à la condition de Helton et Vinnikov. Egalement, lorsque $p > 1$, il n'existe pas à l'heure actuelle de méthode qui permette de déterminer si un ensemble convexe semi-algébrique admet une représentation LMI. Sur la figure 2.2 je représente un exemple classique d'ensemble LMI dans le cas $n = 3$ et $p = 2$, obtenu dans le cadre de relaxations LMI au problème d'optimisation combinatoire MAXCUT. Les équations sont

$$f_1(x) = 3 - x_1^2 - x_2^2 - x_3^2, \quad f_2(x) = 1 + 2x_1x_2x_3 - x_1^2 - x_2^2 - x_3^2$$

et

$$A(x) = \begin{bmatrix} 1 & x_1 & x_2 \\ x_1 & 1 & x_3 \\ x_2 & x_3 & 1 \end{bmatrix}.$$

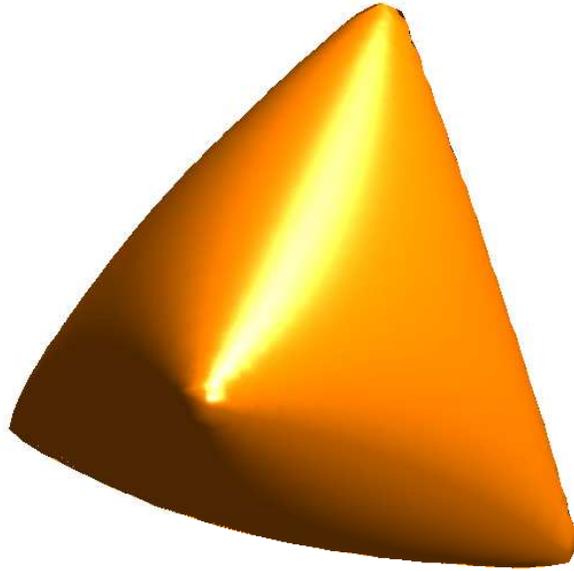


FIG. 2.2 – Exemple d'ensemble semi-algébrique basique convexe tri-dimensionnel.

Dans le cas $n = 2$ et $p = 1$, la condition de Helton et Vinnikov permet d'identifier dans le plan tous les ensembles semi-algébriques basiques qui n'admettent pas de représentation LMI. Un exemple simple est l'ensemble :

$$\{x \in \mathbb{R}^2 : 1 - x_1^4 - x_2^4 \geq 0\}$$

représenté sur la figure 2.3. En effet, une ligne droite quelconque passant par l'origine coupe la quartique $x_1^4 + x_2^4 = 1$ deux fois seulement.

Cet ensemble n'est donc pas LMI. Cependant, il est représentable comme la projection d'un ensemble LMI, et il peut s'écrire

$$\{x \in \mathbb{R}^n : \exists u \in \mathbb{R}^m : A(x, u) \succeq 0\}$$

où $A(x, u)$ est une matrice symétrique affine en x et u . Les variables u jouent donc le rôle de *variables additionnelles*, ou *liftings*. Dans l'exemple

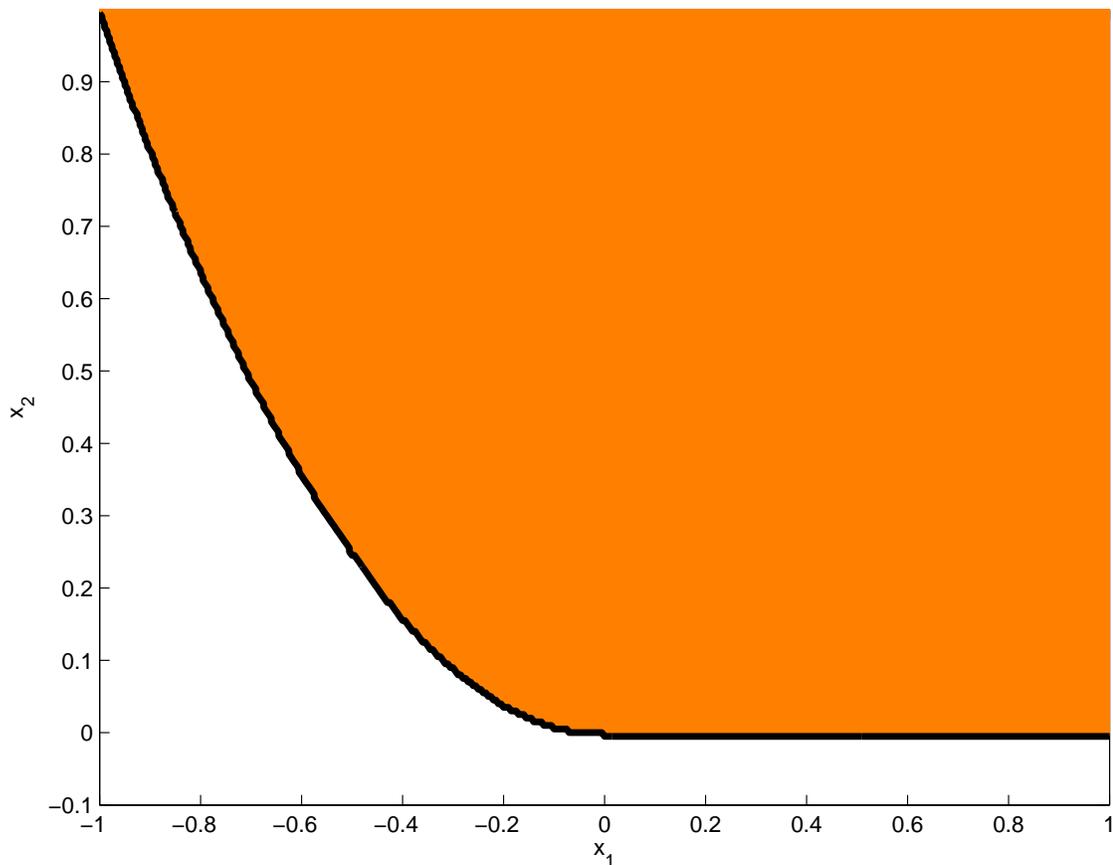


FIG. 2.4 – Exemple d'ensemble semi-algébrique non-basique facilement représentable comme la projection d'un ensemble LMI.

Cet ensemble n'est pas semi-algébrique basique, c'est-à-dire qu'il n'est pas possible de l'exprimer comme l'intersection d'ensembles décrits par des inégalités polynomiales. Cependant, en introduisant une variable supplémentaire ($m = 1$) on peut le représenter comme la projection de l'ensemble LMI

$$A(x, u) = \begin{bmatrix} x_1 - u & & & \\ & x_2 & u & \\ & & u & 1 \end{bmatrix}.$$

On consultera [50, 5] pour une discussion sur la distinction fondamentale entre représentation LMI (sans variables additionnelles) et projection LMI (avec variables additionnelles). De récents travaux de Lasserre semblent indiquer que tous les ensembles semi-algébriques convexes compacts peuvent être uniformément et asymptotiquement représentés par des projections LMI. Dans le paragraphe 2.4 je reviendrai sur la notion de variables additionnelles, et je montrerai qu'elles jouent également un rôle central pour

traiter les ensembles semi-algébriques non-convexes.

2.2 Non-convexité et PMI

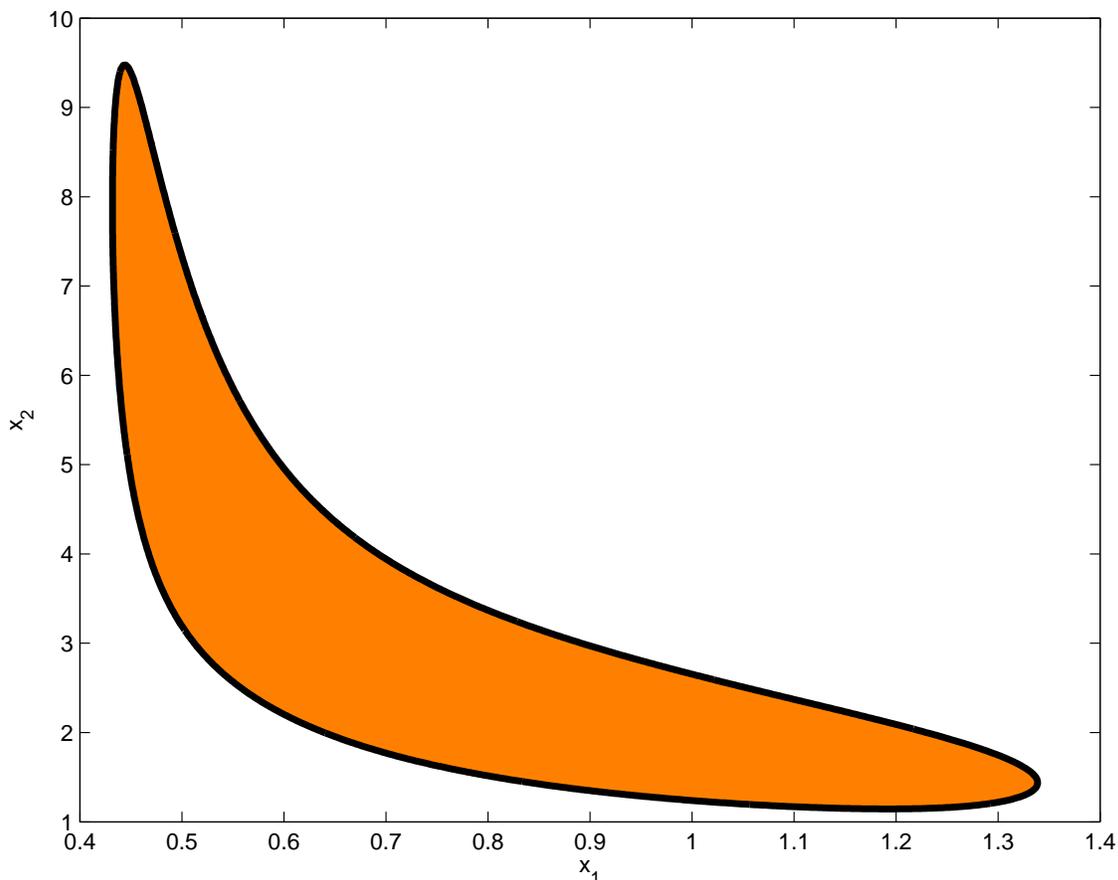


FIG. 2.5 – Exemple d'ensemble BMI non-convexe.

Dans le paragraphe précédent, nous avons vu que les domaines définis par des LMI sont convexes. Si nous autorisons maintenant des termes bilinéaires dans notre description

$$A(x) = A_0 + \sum_{i=1}^n x_i A_i + \sum_{i=1}^n \sum_{j=i+1}^n x_i x_j A_{ij} \succeq 0,$$

nous obtenons une *inégalité matricielle bilinéaire* (BMI), et nous quittons le monde de la convexité. Par exemple, sur la figure 2.5 nous représentons

l'ensemble BMI $\{x \in \mathbb{R}^2 : A(x) \succeq 0\}$ pour

$$A(x) = \begin{bmatrix} 10 - 9x_1 + 2x_2 & \star & \star \\ 1 - x_1 - x_2 & -5 - x_2 + 6x_1x_2 & \star \\ 2 + x_2 - 2x_1x_2 & 3x_1 + x_2 - 3x_1x_2 & x_1 \end{bmatrix}.$$

Ci-dessus les éléments symétriques ne sont pas répétés.

Il s'avère que la plupart des problèmes de commande produisent des BMI lorsqu'on utilise l'approche espace d'état [22]. Avec l'approche polynomiale, que nous allons décrire dans le chapitre 3, nous pouvons rencontrer des *inégalités matricielles polynomiales* (PMI) dont le degré peut être supérieur à deux, et que nous noterons

$$A(x) = \sum_{\alpha} x^{\alpha} A_{\alpha}.$$

Dans cette formule, $\alpha \in \mathbb{N}^n$ est un vecteur à n composantes entières non-négatives représentant les puissances des variables x_i pour $i = 1, \dots, n$.

2.3 Convexité cachée

La convexité d'un ensemble dans un espace Euclidien est une propriété géométrique intrinsèque. Comme mentionné auparavant, il semble que la convexité d'un problème d'optimisation soit une notion beaucoup moins intrinsèque. Par exemple, considérons le problème de la minimisation globale d'un polynôme monovarié réel

$$\min_{x_1 \in \mathbb{R}} f(x_1) = x_1/2 - x_1^2 - x_1^3/2 + x_1^4.$$

représenté sur la figure 2.6.

Ce polynôme est une fonction non-convexe, c'est-à-dire que son épigraphe $\{x \in \mathbb{R}^2 : x_2 \geq f(x_1)\}$ est un ensemble non-convexe. Pourtant, il est possible de démontrer que le problème de la minimisation de $f(x_1)$ est équivalent au problème LMI :

$$\begin{aligned} & \max_{y \in \mathbb{R}^5} y_0 \\ \text{s.t.} & \quad y_0 \geq y_1/2 - y_2 - y_3/2 + y_4 \\ & \quad \begin{bmatrix} 1 & \star & \star \\ y_1 & y_2 & \star \\ y_2 & y_3 & y_4 \end{bmatrix} \succeq 0. \end{aligned}$$

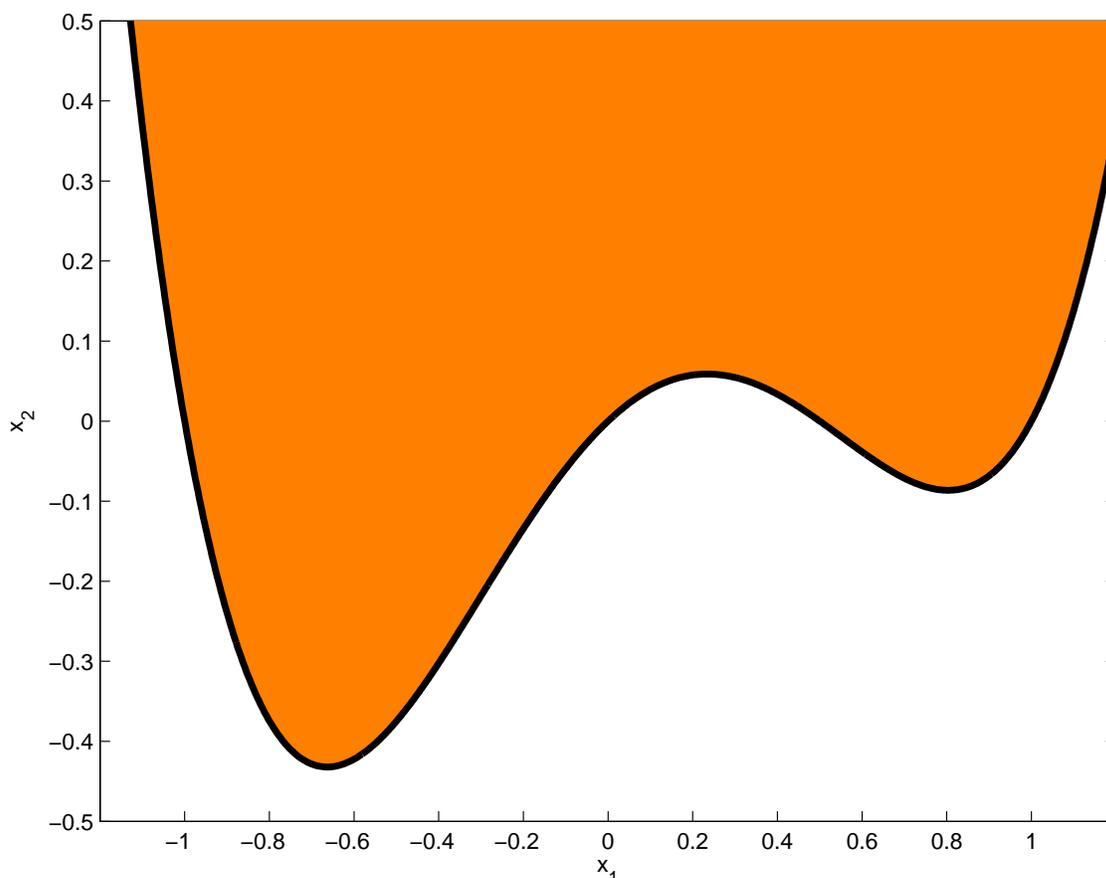


FIG. 2.6 – Polynôme non-convexe.

Géométriquement, en dénotant $x_1 = y_1$, $x_2 = y_0$, $u_1 = y_2$, $u_2 = y_3$, $u_3 = y_4$, la projection sur le plan (x_1, x_2) de l'ensemble LMI $A(x, u) \succeq 0$ décrit ci-dessus est l'enveloppe convexe de l'épigraphe du polynôme.

Remarquons cependant qu'il existe d'autres méthodes pour minimiser un polynôme monovarié. Par exemple on peut calculer les racines réelles de la dérivée du polynôme et comparer les valeurs prises par le polynôme en ces points.

Il existe de nombreux autres exemples de problèmes à formulation non-convexe qui peuvent être reformulés de manière convexe, voir par exemple [5, 33, 55].

2.4 Relaxations LMI

Nous avons vu jusqu'alors que certains ensembles semi-algébriques pouvaient se représenter à l'aide de LMI, ou alors de projections de LMI.

Dans le paragraphe 2.3 nous avons également observé que dans certains cas la convexité d'un problème pouvait être cachée. Dans ce paragraphe nous décrivons succinctement une méthode systématique de construction d'une hiérarchie de problèmes convexes LMI permettant de résoudre des problèmes non-convexes polynomiaux. Cette méthode a été originalement proposée par Lasserre [45]. Par la suite, elle a donné lieu à de nombreuses études, extensions, généralisations et applications, notamment en automatique [63].

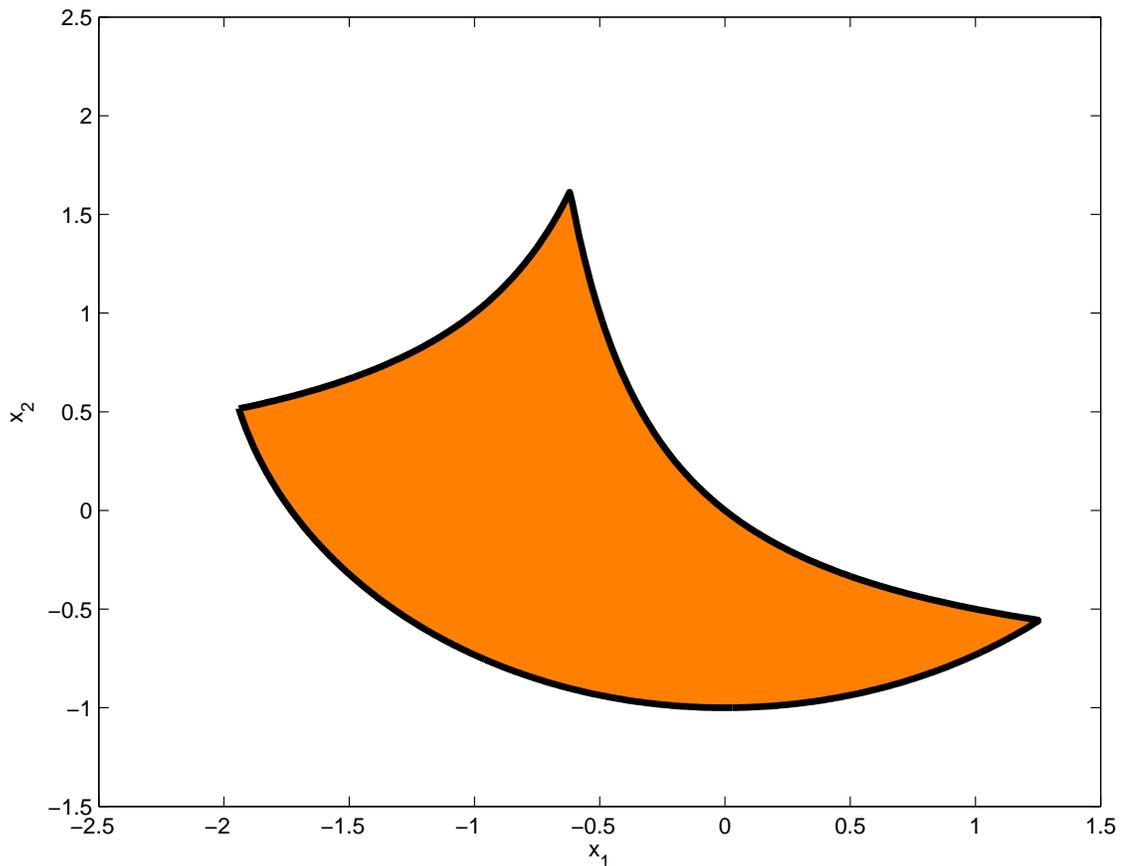


FIG. 2.7 – Ensemble admissible non-convexe.

La construction est relativement simple à expliquer à l'aide d'un exemple. Considérons le problème d'optimisation non-convexe

$$\begin{aligned}
 \max_{x \in \mathbb{R}^2} \quad & x_2 \\
 \text{t.q.} \quad & 3 + 2x_2 - x_1^2 - x_2^2 \geq 0 \\
 & -x_1 - x_2 - x_1x_2 \geq 0 \\
 & 1 + x_1x_2 \geq 0
 \end{aligned}$$

où la fonction objectif linéaire x_2 est maximisée sur un ensemble non-convexe délimité par des arcs circulaires et hyperboliques, voir la figure 2.7.

La première relaxation LMI de la hiérarchie mentionnée ci-dessus consiste à linéariser tous les monômes et à relâcher les relations non-linéaires qu'ils doivent satisfaire :

$$\begin{aligned} & \max_{y \in \mathbb{R}^5} y_{01} \\ \text{s.t.} & \begin{bmatrix} 1 & \star & \star \\ y_{10} & y_{20} & \star \\ y_{01} & y_{11} & y_{02} \end{bmatrix} \succeq 0 \\ & 3 + 2y_{01} - y_{20} - y_{02} \geq 0 \\ & -y_{10} - y_{01} - y_{11} \geq 0 \\ & 1 + y_{11} \geq 0. \end{aligned}$$

La matrice 3×3 positive semi-définie ci-dessus est dénommée matrice des moments d'ordre 1. S'il on impose que cette matrice soit de rang égal à 1, on se ramène au problème non-convexe original. En effet la contrainte de rang indique que tous les mineurs s'annulent simultanément, et donc par exemple

$$\det \begin{bmatrix} 1 & y_{10} \\ y_{10} & y_{20} \end{bmatrix} = y_{20} - y_{10}^2 = 0, \quad \det \begin{bmatrix} y_{20} & y_{11} \\ y_{11} & y_{02} \end{bmatrix} = y_{20}y_{02} - y_{11}^2 = 0$$

et en général toutes les relations non-linéaires seront satisfaites pour les moments, tout comme pour les monômes originaux. La relaxation LMI est donc obtenue en enlevant, c'est-à-dire en *relâchant*, la contrainte non-convexe de rang.

La résolution de cette première relaxation LMI donne la valeur optimale 2,0000 avec 5 chiffres significatifs. Sur la figure 2.8 nous représentons la projection de l'ensemble admissible de la relaxation LMI sur le plan (y_{10}, y_{01}) des monômes de premier degré linéarisés, également dénommés moments de premier ordre. Puisqu'il s'agit d'une relaxation, cet ensemble convexe contient l'ensemble admissible non-convexe original. On peut ainsi remarquer que l'optimum de la relaxation LMI est atteint en un point $x_1 = -1$, $x_2 = 2$ qui est inadmissible pour le problème non-convexe.

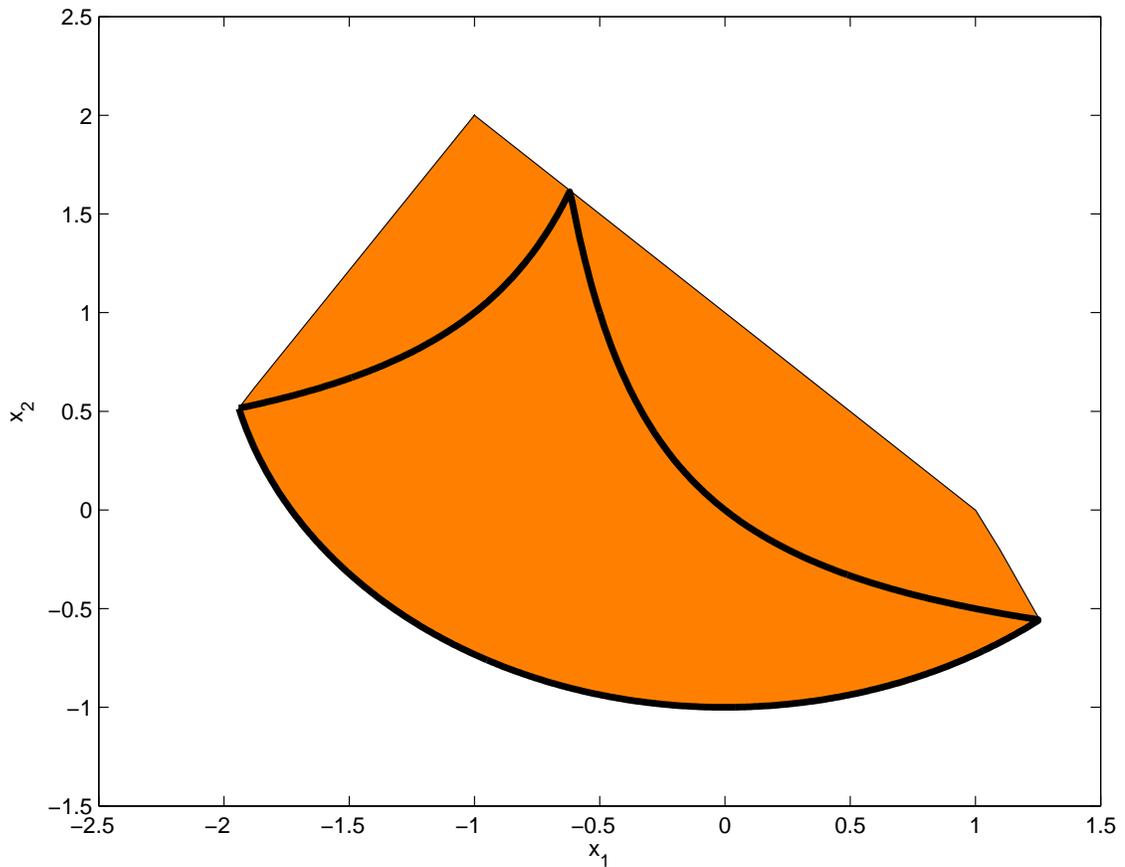


FIG. 2.8 – Projection de l'ensemble admissible de la première relaxation LMI.

La deuxième relaxation LMI de la hiérarchie s'écrit

$\max_{y \in \mathbb{R}^{14}} y_{01}$ t.q.

$$\begin{bmatrix} 1 & \star & \star & \star & \star & \star \\ y_{10} & y_{20} & \star & \star & \star & \star \\ y_{01} & y_{11} & y_{02} & \star & \star & \star \\ y_{20} & y_{30} & y_{21} & y_{40} & \star & \star \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & \star \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} 3 + 2y_{01} - y_{20} - y_{02} & \star & \star \\ 3y_{10} + 2y_{11} - y_{30} - y_{12} & 3y_{20} + 2y_{21} - y_{40} - y_{22} & \star \\ 3y_{01} + 2y_{02} - y_{21} - y_{03} & 3y_{11} + 2y_{12} - y_{31} - y_{13} & 3y_{02} + 2y_{03} - y_{22} - y_{04} \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} -y_{10} - y_{01} - y_{11} & \star & \star \\ -y_{20} - y_{11} - y_{21} & -y_{30} - y_{21} - y_{31} & \star \\ -y_{11} - y_{02} - y_{12} & -y_{21} - y_{12} - y_{22} & -y_{12} - y_{03} - y_{13} \end{bmatrix} \succeq 0$$

$$\begin{bmatrix} 1 + y_{11} & \star & \star \\ y_{10} + y_{21} & y_{20} + y_{31} & \star \\ y_{01} + y_{12} & y_{11} + y_{22} & y_{02} + y_{13} \end{bmatrix} \succeq 0.$$

Cette LMI est construite en introduisant des variables additionnelles (liftings) similaires à celles utilisées dans le paragraphe 2.1 pour modéliser des ensembles semi-algébriques convexes non-basiques. La matrice des moments est étendue à l'ordre 2 (monômes de degré 4), et sa structure est appliquée (localisée) aux contraintes également.

La résolution numérique de cette deuxième LMI donne la valeur optimale 1,6180 qui est en fait l'optimum global $(1 + \sqrt{5})/2$ aux erreurs d'arrondis près. De plus, les moments de premier ordre $y_{10}^* = -0,6180$, $y_{01}^* = 1,6180$ sont une solution globalement optimale au problème non-convexe original.

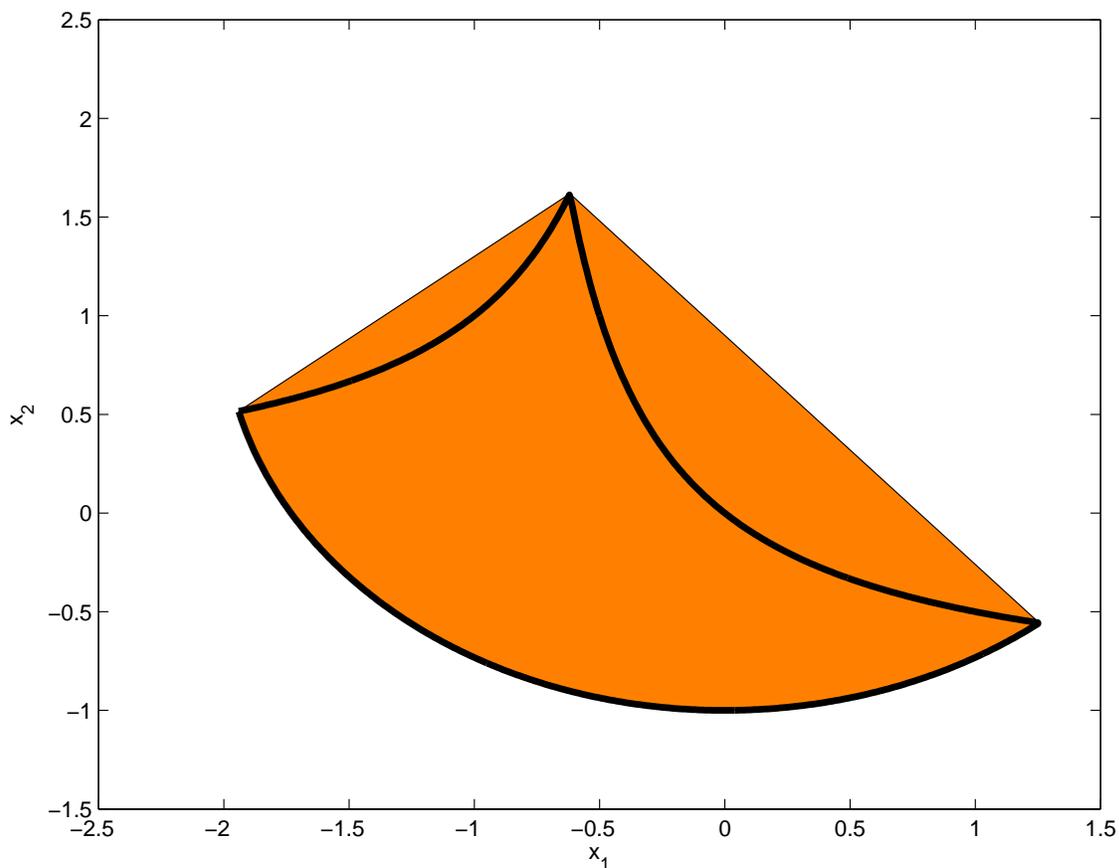


FIG. 2.9 – Projection de l'ensemble admissible de la deuxième relaxation LMI.

Sur la figure 2.9 nous représentons la projection de l'ensemble admissible de la deuxième relaxation LMI sur le plan (y_{10}, y_{01}) des moments de premier ordre. Par construction, cet ensemble est inclus dans l'ensemble admissible de la première relaxation LMI de la figure 2.8. De plus, cet ensemble est ici exactement l'enveloppe convexe de l'ensemble admissible

non-convexe original. L'optimum global est donc atteint puisque l'on maximise une fonction linéaire.

On imagine facilement comment étendre la construction aux monômes et moments de degrés plus élevés (troisième relaxation LMI, etc.), au prix d'introduire de plus en plus de variables additionnelles et de contraintes. Pour plus de détails techniques sur la construction de cette *hiérarchie de relaxations LMI*, et en particulier pour une interprétation probabiliste des problèmes d'optimisation polynomiale justifiant la terminologie de matrice des moments, voir [45]. Nous avons développé un logiciel permettant la construction de ces relaxations et leur résolution par les solveurs coniques, voir le paragraphe 4.3.

La construction décrite ci-dessus, proposée originalement uniquement dans le cas de l'optimisation polynomiale avec contraintes scalaires, a été étendue au cas des contraintes matricielles, c'est-à-dire aux PMI [29]. Les bases théoriques nécessaires à la dérivation de cette hiérarchie de relaxations LMI (dans le cas scalaire ou matriciel) et surtout à la preuve de sa convergence font appel à l'analyse fonctionnelle et à la géométrie algébrique réelle. Les idées fondamentales sont exposées dans [45], et mettent en évidence la dualité entre théorie des moments (conditions d'existence d'une mesure supportée sur un ensemble semi-algébrique) et la représentation de polynômes multivariés positifs (sur le même ensemble) comme sommes de carrés. Ce dernier aspect a également été étudié par Parrilo [52], et nous verrons dans le chapitre 3 quelques applications en commande.

2.5 Conclusions

Dans ce chapitre nous avons vu que les LMI pouvaient modéliser de nombreux objets semi-algébriques convexes, notamment à l'aide des projections et de la notion de variable additionnelle. La convexité d'un problème peut être cachée, et il s'avère que de nombreux problèmes non-convexes d'optimisation polynomiale peuvent être résolus facilement à l'aide de la programmation semi-définie. Ainsi, la frontière entre convexité et non-convexité semble à présent beaucoup moins claire que l'affirmait Rockafellar en 1993, voir l'introduction.

La difficulté de résolution d'un problème polynomial ne semble pas être reliée à la non-convexité de sa formulation. Il existe des problèmes à formulation non-convexe qui peuvent être résolus de manière très efficace car ils

admettent une reformulation convexe simple, à savoir comme un problème LMI avec peu de variables additionnelles.

Malheureusement, il semble très difficile de savoir à l'avance si un problème non-convexe admet une reformulation convexe simple. Cela semble possible uniquement pour des classes très restreintes de problèmes. En pratique, la solution consiste à construire une hiérarchie de relaxations convexes LMI du problème original obtenues en rajoutant progressivement des variables additionnelles et des contraintes les reliant. La profondeur qu'il est nécessaire d'atteindre dans cette hiérarchie, c'est-à-dire le nombre de variables et de contraintes additionnelles que l'on doit introduire, pourrait être une mesure de la difficulté de résolution d'un problème beaucoup plus informative que la distinction entre convexité et non-convexité.

Chapitre 3

Polynômes en commande robuste

Les *polynômes* sont des outils de modélisation particulièrement adaptés aux systèmes linéaires. Les approches polynomiale [41] et comportementale [54] utilisent abondamment les polynômes et matrices polynomiales, dont les propriétés mathématiques sont reliées aux propriétés physiques des systèmes modélisés. L'approche polynomiale peut être vue comme une alternative complémentaire à l'approche espace d'état.

Dans ce chapitre on rappelle tout d'abord rapidement (paragraphe 3.1) les notions fondamentales liées aux polynômes vus comme objets de modélisation pour les systèmes linéaires nominaux et incertains. Le paragraphe 3.2 est consacré aux outils *d'analyse de stabilité* pour les polynômes, qui sont actuellement beaucoup plus nombreux que les outils de *stabilisation* de polynômes étudiés dans le paragraphe 3.3. Une difficulté fondamentale semble être la non-convexité de la région de stabilité dans l'espace des coefficients des polynômes. Le chapitre se termine sur deux approches possibles pour étudier la stabilité et la stabilisation de polynômes nominaux et incertains, à savoir une approximation convexe de la région de stabilité autour d'un polynôme central (paragraphe 3.4) et l'utilisation de techniques de géométrie algébrique réelle (paragraphe 3.5 et 3.6). Comme nous le verrons, le dénominateur commun à ces approches est l'utilisation des LMI.

3.1 Systèmes linéaires et polynômes

Comme rappelé dans l'ouvrage de référence [35], le comportement entrée-sortie des systèmes linéaires scalaires (à une entrée et une sortie, ou SISO)

peut être modélisé à l'aide d'une fonction de transfert rationnelle :

$$P(s) = \frac{b(s)}{a(s)}$$

avec des *polynômes* dénominateur $a(s)$ et numérateur $b(s)$ à coefficients réels. L'indéterminée s est la variable complexe de Laplace dans le cas de systèmes continus, ou alors l'opérateur de décalage dans le cas de systèmes discrets. Par causalité, le degré de $b(s)$ est inférieur ou égal au degré de $a(s)$. Les racines de $a(s)$ sont les pôles, et les racines de $b(s)$ les zéros du système. On suppose que $P(s)$ est irréductible, c'est-à-dire qu'il n'y a pas de facteur commun entre $a(s)$ et $b(s)$, ou encore que les zéros et les pôles du système sont distincts.

Dans le cas de systèmes multivariables (à plusieurs entrées et sorties, ou MIMO), la fonction de transfert devient matricielle, et plusieurs écritures sont possibles car le produit matriciel n'est pas commutatif. Nous pouvons rencontrer des fractions de *matrices polynomiales* à gauche, à droite, ou avec dénominateur scalaire :

$$P(s) = A_L^{-1}(s)B_L(s) = B_R(s)A_R^{-1}(s) = \frac{B(s)}{a(s)}.$$

Notons que $a(s) = \det A_L(s) = \det A_R(s)$, et par abus de langage on parlera des racines, ou zéros, d'une matrice polynomiale, c'est-à-dire des racines de son déterminant.

Remarquons que les matrices polynomiales de premier degré $A(s) = sI_n - A$, appelées également faisceaux, correspondent aux systèmes linéaires décrits par des équations de premier ordre, c'est-à-dire à la modélisation dans l'espace d'état. Ainsi, tous les résultats disponibles pour les matrices polynomiales de degré quelconque pourront être spécialisés à l'espace d'état, que l'on considère donc comme un cas particulier [62].

Dans les années 1980 les progrès considérables en théorie des systèmes linéaires, résumés dans l'ouvrage [35], ont permis diverses extensions des résultats à des problèmes physiquement plus réalistes, et en particulier, à la prise en compte explicite de l'incertitude. Les travaux de Doyle, Helton, Kwakernaak, Safonov ou Zames, pour ne citer qu'eux, ont permis de fonder la *commande robuste*, qui est maintenant une branche solide et vive de l'automatique, comme en témoigne le riche programme de la dernière conférence IFAC Symposium on Robust Control Design organisée à Toulouse en juillet 2006. Une des avancées majeures de la commande robuste

a été l'optimisation H_∞ , qui est maintenant utilisée industriellement tout autant, voire plus, que la commande H_2 , cette dernière étant une extension de la commande linéaire quadratique. Ainsi, la poursuite de trajectoire du lanceur Ariane 5+ (charge utile 10 tonnes) en phase atmosphérique est assurée depuis 2002 par un correcteur H_∞ [2]. On pourra par exemple consulter [64, 68] comme ouvrages de référence sur la commande robuste, parmi beaucoup d'autres.

Les incertitudes affectant les systèmes sont de diverses natures, comme par exemple les dynamiques négligées, les non-linéarités douces, les paramètres physiques variant dans le temps. Pour simplifier, on considère deux grandes classes : les incertitudes structurées et les incertitudes non-structurées [68]. En général il est beaucoup plus difficile de gérer les incertitudes structurées que les incertitudes non-structurées. Plus on tente d'exploiter la connaissance et la structure du système, plus les méthodes d'analyse et de synthèse sont complexes et difficiles à mettre en oeuvre.

De manière élémentaire, on considère deux types d'incertitudes représentatives des deux grandes classes mentionnées ci-dessus : l'incertitude bornée en norme (non-structurée) et l'incertitude polytopique (structurée). Dans le cadre polynomial, on retrouvera donc respectivement des *ellipsoïdes* et des *polytopes* de polynômes [1, 3, 6].

3.2 Stabilité des polynômes

Etudier la stabilité asymptotique d'un système linéaire revient à étudier le lieu des racines de son polynôme caractéristique, c'est-à-dire du dénominateur scalaire $a(s)$. On parle alors de *polynôme stable*. Dans le cas de systèmes continus, la stabilité est assurée lorsque toutes les racines sont dans le demi-plan gauche ouvert. Dans le cas de systèmes discrets, la région de placement des racines est l'intérieur du disque unitaire. Nous utiliserons la notation \mathcal{D} pour la région de placement des racines dans le plan complexe, qui est un ensemble ouvert éventuellement non-borné.

Dès la fin du 19^{ème} siècle, Routh et Hurwitz, sous l'impulsion de Maxwell, formulèrent les conditions que doivent vérifier les coefficients d'un polynôme pour qu'il soit stable, dans le cas continu. En notant le polynôme $a(s) = a_0 + a_1s + a_2s^2 + \dots + a_ns^n$, les conditions de stabilité s'écrivent comme n inégalités polynomiales strictes en les coefficients a_i , que l'on notera $f_i(a) > 0$ pour $i = 1, \dots, n$. Les polynômes $f_i(a)$ sont les mineurs

principaux de degré i de la matrice de Hurwitz

$$\begin{bmatrix} a_{n-1} & a_{n-3} & & 0 & 0 \\ a_n & a_{n-2} & & \vdots & \vdots \\ 0 & a_{n-1} & \cdots & 0 & 0 \\ 0 & a_n & & a_0 & 0 \\ \vdots & \vdots & & a_1 & 0 \\ 0 & 0 & & a_2 & a_0 \end{bmatrix}$$

qui est carrée de dimension n . En d'autres termes, l'ensemble de stabilité \mathcal{S}_n est *l'intérieur d'un semi-algébrique basique* dans l'espace des coefficients \mathbb{R}^{n+1} . Des résultats similaires au critère de Routh-Hurwitz sont disponibles dans le cas discret, où la paternité des conditions de stabilité est généralement attribuée à Schur. On consultera par exemple le chapitre XV de [19] ou encore l'ouvrage [51] pour une bonne introduction à ces critères algébriques de stabilité.

Par la suite, Hermite dériva une formulation symétrique du critère de stabilité, à l'aide de la théorie des résultants et des formes quadratiques. Le polynôme $a(s)$ est stable si et seulement si une matrice $H(a) \in \mathbb{R}^{n \times n}$ *symétrique* et *quadratique* en a est définie positive. Les mineurs de cette matrice sont bien sûrs reliés aux polynômes $f_i(a)$ introduits ci-dessus, voir [51].

La géométrie de l'ensemble de stabilité \mathcal{S}_n est très complexe. Une de ses propriétés essentielles est la *non-convexité* pour $n \geq 3$. Sur la figure 3.1 on représente dans le cas des polynômes discrets l'intersection de \mathcal{S}_3 par le sous-espace affine bi-dimensionnel généré par les colonnes de la matrice

$$\frac{1}{200} \begin{bmatrix} -165 + 51a_1 + 20a_2 \\ 179 + 5a_1 + 18a_2 \\ -495 + 135a_1 + 60a_2 \end{bmatrix}$$

et qui est étudiée dans le Chapitre 10 de [1]. On remarquera la présence de longues branches hyperboliques non-convexes caractéristiques de la géométrie de \mathcal{S}_n .

Notons que la complexité de la géométrie de \mathcal{S}_n contraste avec la relative simplicité du test de l'appartenance d'un vecteur à cet ensemble. En effet, le critère de Routh-Hurwitz revient à tester le signe de fonctions polynomiales en a dont la construction est élémentaire. Le critère de Hermite

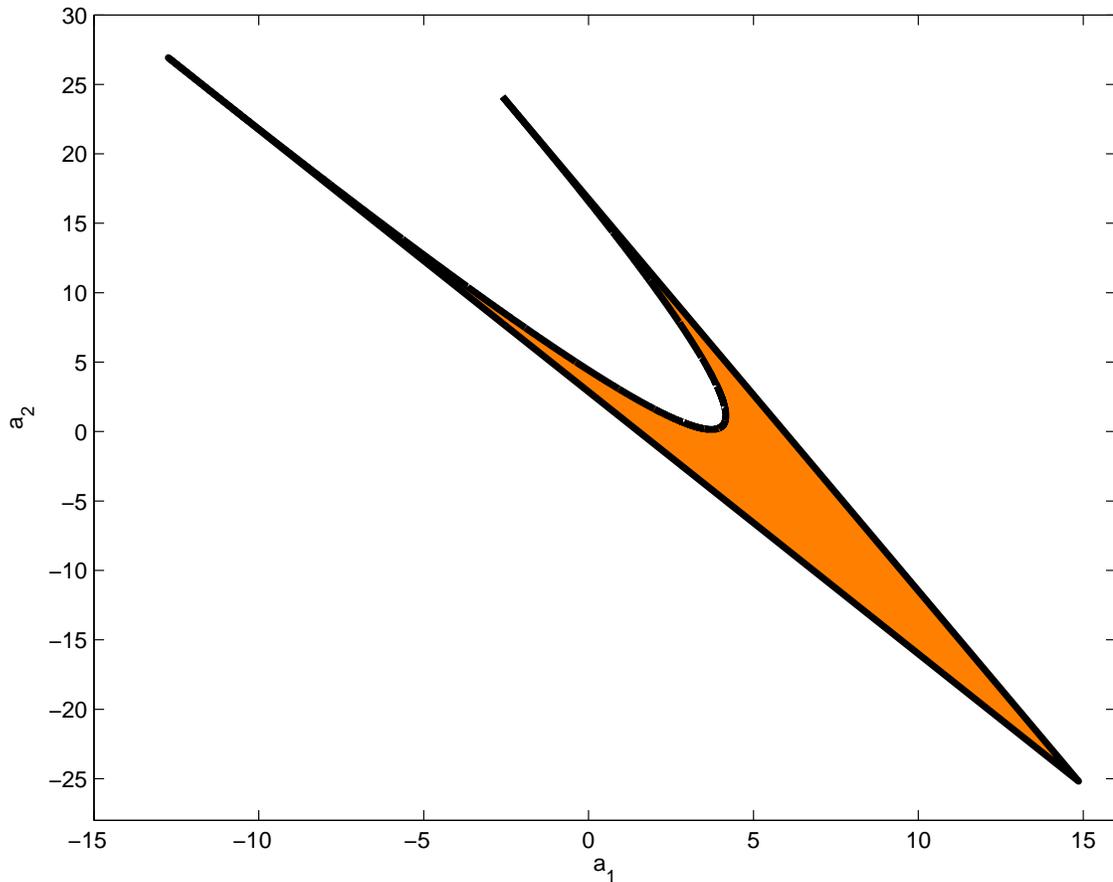


FIG. 3.1 – Coupe plane de l'ensemble non-convexe de stabilité des polynômes discrets de troisième degré.

est également très simple puisqu'il s'agit de vérifier le signe de tous les mineurs principaux d'une matrice symétrique quadratique en a . En d'autres termes, le problème de l'analyse de stabilité d'un polynôme peut être résolu à l'aide d'un nombre raisonnable (au plus cubique en fonction du degré n) d'opérations arithmétiques élémentaires sur les coefficients (addition, soustraction, multiplication, division et comparaison).

Dans le cas des polynômes incertains (ellipsoïdes et polytopes) évoqués dans le paragraphe 3.1, l'analyse de stabilité est plus délicate. Le résultat central qui a été à l'origine de nombreux travaux dans le domaine est le théorème de Kharitonov (1978), qui concerne la stabilité robuste d'un polytope de polynômes bien particulier, à savoir un hyper-rectangle de dimension n égale au degré du polynôme, dans le cas scalaire et continu. La stabilité robuste d'un tel polytope est équivalente à la stabilité de seulement 4 sommets bien identifiés parmi les 2^n sommets de l'hyper-rectangle.

Ainsi il n'est pas nécessaire d'appliquer le critère de Routh-Hurwitz à l'ensemble (infini) des polynômes dans le polytope, où encore à l'ensemble des sommets, qui est fini mais de cardinalité exponentielle en n . Il suffit de l'appliquer sur 4 polynômes, ce qui est une simplification et un gain calculatoire considérables. Malheureusement, il ne semble pas exister de résultat aussi fort que le théorème de Kharitonov lorsqu'on considère des polytopes plus généraux, ou la stabilité robuste de polynômes discrets. Une réduction est toutefois possible dans le cas de polytopes de polynômes scalaires, et il existe un théorème permettant de restreindre le test de stabilité aux arêtes exposées uniquement [1, 3, 6].

Dans les cas plus généraux, il ne semble pas possible d'assurer des réductions de complexité similaires. En particulier, la non-convexité de l'ensemble de stabilité \mathcal{S}_n a pour conséquence que la stabilité des deux sommets d'un segment n'implique pas la stabilité du segment tout entier. C'est vrai dans le cas des polynômes scalaires, voir la figure 3.1, mais également dans le cas des matrices (c'est-à-dire des polynômes matriciels linéaires). A titre d'exemple, la matrice

$$A = \begin{bmatrix} -6 & 12 \\ 2 & -6 \end{bmatrix}$$

et sa transposée A^* sont stables en temps continu, mais le milieu du segment

$$\frac{1}{2}(A + A^*) = \begin{bmatrix} -6 & 7 \\ 7 & -6 \end{bmatrix}$$

n'est pas stable.

Le cadre le plus général pour l'étude de stabilité robuste semble être la μ -analyse, proposée indépendamment par Doyle et Safonov dans les années 1980 [68]. Ces techniques ont donné lieu au développement de deux boîtes à outils Matlab spécialisées, maintenant intégrées à la Robust Control Toolbox.

Dans le cas d'un ellipsoïde de polynômes, l'analyse de stabilité robuste est équivalente au calcul de la norme H_∞ d'une certaine fonction de transfert rationnelle. Ce calcul peut s'effectuer efficacement (en arithmétique à virgule flottante) à l'aide de l'algèbre linéaire numérique (valeurs propres de matrices Hamiltoniennes) où à l'aide de l'algèbre polynomiale (séquences de Sturm) [9].

On pourra consulter les ouvrages [1, 3, 6] pour des études complètes des conditions de stabilité des polynômes incertains. En particulier le chapitre

introductif de [3] retrace d'une manière fort intéressante l'histoire des tests de stabilité robuste.

3.3 Stabilisation de polynômes

Supposons à présent que les propriétés du système $P(s)$ défini dans le paragraphe 3.1 ne soient pas satisfaisantes. Par exemple supposons que le polynôme caractéristique $a(s)$ soit instable. Considérons la configuration de rétroaction élémentaire où

$$K(s) = \frac{y(s)}{x(s)}$$

est la fonction de transfert d'un *correcteur*, ou contrôleur linéaire. La fonction de transfert résultante, dite en *boucle fermée*, s'écrit alors :

$$T(s) = \frac{K(s)P(s)}{1 + K(s)P(s)} = \frac{b(s)y(s)}{a(s)x(s) + b(s)y(s)}.$$

En l'absence de simplification pôles-zéros, la stabilité du système en boucle fermée sera assurée par la stabilité du polynôme dénominateur, appelé polynôme caractéristique :

$$a(s)x(s) + b(s)y(s) = d(s).$$

Le problème de synthèse, sous sa forme la plus simple, est donc celui de la *stabilisation* : étant donnés deux polynômes $a(s)$, $b(s)$ de degré n , trouver deux polynômes $x(s)$, $y(s)$ de degré m tels que le polynôme $d(s)$ soit stable.

On dénotera $x = [x_0 \ x_1 \ x_2 \ \dots]$ le vecteur colonne des coefficients de $x(s) = x_0 + x_1s + x_2s^2 + \dots$, et de même pour les autres polynômes. L'équation de stabilisation ci-dessus est linéaire en les inconnues (x , y et d), et elle est formulée dans l'anneau des polynômes. Par analogie avec l'anneau des entiers, on utilise la terminologie d'équation diophantienne.

Bien sûr on supposera que $a(s)$ est instable, sinon $y(s) = 0$ est une solution triviale au problème de stabilisation. On supposera également que $m < n - 1$, car dans le cas $m = n - 1$ l'équation diophantienne peut se reformuler, par identification des coefficients des puissances successives de

l'indéterminée s , comme une équation linéaire

$$[x_0 \ x_1 \ \cdots \ y_0 \ y_1 \ \cdots] \begin{bmatrix} a_0 & a_1 & \cdots & \\ & a_0 & a_1 & \cdots \\ & & & \ddots \\ b_0 & b_1 & \cdots & \\ & b_0 & b_1 & \cdots \\ & & & \ddots \end{bmatrix} = [d_0 \ d_1 \ \cdots]$$

dont la matrice carrée, dite de Sylvester, est non-singulière car $a(s)$ et $b(s)$ n'ont pas de racines communes. Dans le cas $m = n - 1$ on peut donc choisir n'importe quel polynôme stable $d(s)$ de degré $2n - 1$, et il existera une solution $x(s), y(s)$ à l'équation diophantienne, dite alors de placement de pôles. Seul le cas $m < n - 1$ est donc non-trivial pour la stabilisation des systèmes SISO.

Notons au passage que dans le cas de systèmes MIMO décrits par des fonctions de transferts matricielles, l'équation diophantienne devient

$$D(s) = A(s)X(s) + B(s)Y(s)$$

où maintenant $A(s)$, $X(s)$ et $D(s)$ sont des matrices polynomiales non-singulières. En particulier, si $X(s)$ et $Y(s)$ sont des matrices constantes, nous retrouvons le problème de stabilisation par retour de sortie statique. Par facilité, dans le cas MIMO on notera également par \mathcal{S}_p l'ensemble de stabilité des matrices polynomiales de degré p , même si la dimension de cet ensemble dépend maintenant également de la dimension des matrices, et non plus uniquement de leur degré.

Pour résumer, le problème de stabilisation d'un système d'ordre n par un correcteur d'ordre m revient donc à trouver un point dans l'intersection entre l'ensemble \mathcal{S}_p , où $p = n + m$, et l'espace affine correspondant à l'équation diophantienne. Etant donnée la complexité de la géométrie de \mathcal{S}_p , et notamment sa non-convexité lorsque $p \geq 3$, de nombreux travaux ont été consacrés à la construction d'*approximations convexes de l'ensemble de stabilité* \mathcal{S}_p . En particulier les approximations par l'intérieur donnent des conditions suffisantes de stabilisation, et les approximations par l'extérieur donnent des conditions suffisantes d'impossibilité de stabilisation.

Dans les années 1990, de nombreuses études ont été consacrées aux approximations convexes de l'ensemble de stabilité. Les ouvrages clés sont [1, 3, 6], déjà mentionnés dans le cadre de l'analyse de stabilité robuste.

Dans ce domaine, un des résultats remarquables est encore le théorème de Kharitonov, qui permet de s'assurer de l'inclusion dans \mathcal{S}_p d'un hyperrectangle de dimension p en testant uniquement la stabilité de 4 de ses sommets. On peut également trouver dans ces ouvrages de nombreuses approximations par l'intérieur de \mathcal{S}_p à l'aide de sphères ou d'ellipsoïdes, qui l'ont l'avantage d'avoir une description algébrique très simple, mais l'inconvénient d'être conservatrices et peu flexibles. Dans le paragraphe suivant, nous proposons une approximation convexe par l'intérieur décrite par la projection d'une LMI, en utilisant la notion de variables additionnelles introduite dans le chapitre 2.

Il faut souligner que la stabilisation est rarement la seule spécification formulée pour un système en boucle fermée. En général on exigera également un certain niveau de *performance*, et même typiquement plusieurs critères de performances contradictoires sur plusieurs fonctions de transfert. La synthèse d'un correcteur est alors un compromis entre les différentes spécifications, et tout l'art de l'ingénieur consiste à trouver la formulation des critères en adéquation avec le comportement physique optimal désiré pour le système. Les critères les plus classiques et les mieux maîtrisés sont de type fréquentiel, pour des raisons essentiellement historiques. Comme mentionné auparavant dans le paragraphe 3.1, les mesures de performance standards sont la norme H_2 , une généralisation du critère linéaire quadratique, et la norme H_∞ issue des travaux fondateurs en commande robuste. Pour simplifier, dans le cas SISO, la norme H_2 est liée à l'aire du diagramme de Bode, ce qui en fait une mesure moyenne dans le domaine fréquentiel, alors que la norme H_∞ correspond à la valeur la plus élevée dans ce même diagramme, ce qui en fait une mesure de type pire des cas. Un problème de synthèse multi-objectif typique peut être par exemple la minimisation de la norme H_2 d'une certaine fonction de transfert, sous la contrainte que la norme H_∞ d'une autre fonction de transfert soit inférieure à une certaine valeur, et sous la contrainte de stabilité robuste de la boucle fermée vis-à-vis d'incertitudes de type polytopique.

Avant de conclure ce paragraphe il est important de souligner que les critères de stabilité et de performance décrits ci-dessus ont des propriétés mathématiques assez peu favorables. Par conséquent, il est souvent difficile de formuler et de résoudre des problèmes de commande robuste multi-objectif à l'aide des outils d'optimisation non-linéaires classiques. Une propriété déjà évoqué est la non-convexité de l'ensemble de stabilité, rendant

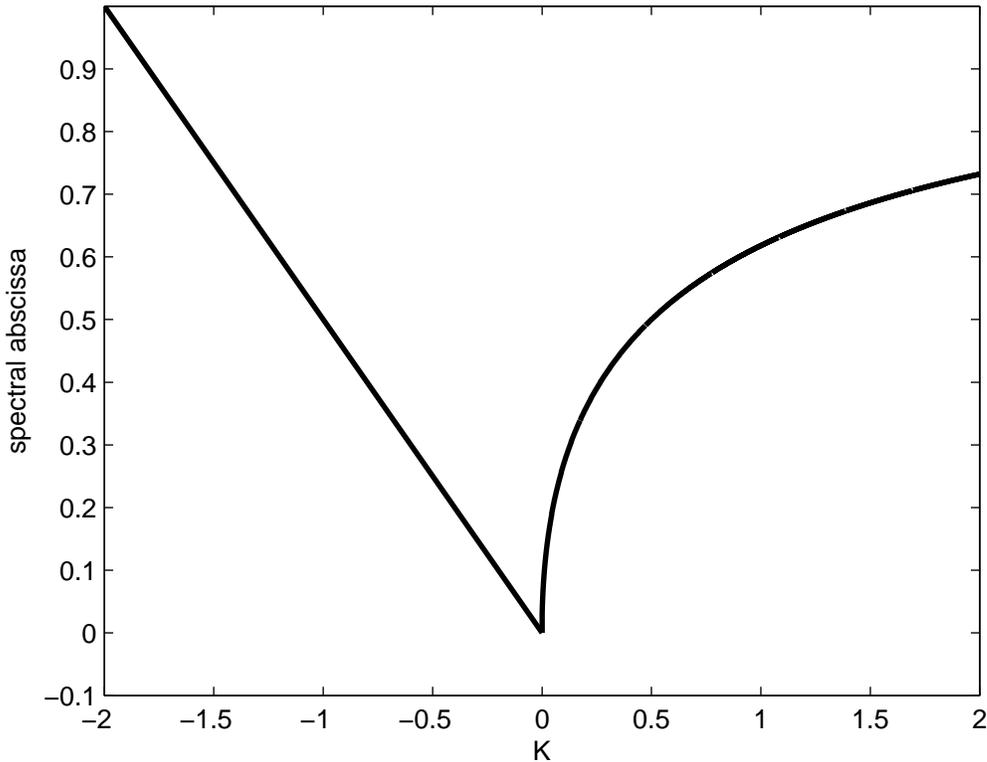


FIG. 3.2 – Abscisse spectrale non-lisse et non-Lipschitz à l’origine.

l’optimisation délicate. Une autre propriété peut-être moins appréciée est la *non-différentiabilité* des fonctions associées aux critères de performance. Considérons par exemple le problème de stabilisation d’un système linéaire $\dot{x} = Ax + Bu$, $y = Cx$ par un gain statique $u = Kx$. L’existence d’un gain stabilisant K est équivalente à la négativité de l’abscisse spectrale de la matrice $A + BKC$, c’est-à-dire de la partie réelle maximale de ses valeurs propres. Le problème de stabilisation revient donc à minimiser l’abscisse spectrale. Il se trouve que cette fonction est typiquement non-lisse (non-différentiable et parfois même non-Lipschitz) aux optima locaux, comme illustré sur la figure 3.2 dans le cas très simple de la matrice

$$\begin{bmatrix} 0 & 1 \\ K & -K \end{bmatrix}.$$

De manière similaire, la norme H_∞ n’est pas une fonction lisse des paramètres [14]. Il semble cependant que la norme H_2 soit lisse à l’intérieur de l’ensemble de stabilité \mathcal{S}_n , car elle peut s’exprimer comme une fonction

analytique rationnelle en les paramètres.

3.4 Polynôme central

Rappelons qu'un polynôme stable possède ses racines dans une région de placement \mathcal{D} , un ouvert du plan complexe, typiquement le demi-plan gauche ou le disque unitaire.

Etant donné un polynôme matriciel stable $C(s)$ carré de dimension q et de degré n , on remarquera qu'une condition suffisante de stabilité d'un autre polynôme matriciel $D(s)$ de mêmes dimension et degré est la condition de positivité réelle stricte (SPR) sur la matrice rationnelle $G(s) = C^{-1}(s)D(s)$:

$$G(s) + G^*(s) \succ 0, \forall s \in \partial\mathcal{D}$$

où $\partial\mathcal{D}$ est la courbe frontière $\sigma(s) = d_0 + d_1(s + s^*) + d_2s^*s = 0$ de l'ouvert $\mathcal{D} = \{s \in \mathbb{C} : \sigma(s) < 0\}$, avec d_0, d_1, d_2 scalaires réels donnés. Par exemple, $\sigma(s) = s + s^*$ lorsque \mathcal{D} est le demi-plan gauche, et $\sigma(s) = -1 + s^*s$ lorsque \mathcal{D} est le disque unitaire.

En effet, si $D(s) = C(s)$ la condition SPR est trivialement satisfaite puisque $G(s) = I$. Si $D(s)$ s'éloigne continûment de $C(s)$, dans l'espace des coefficients, alors la stabilité est perdue lorsqu'il existe un vecteur v et un zéro $s \in \partial\mathcal{D}$ tels que $D(s)v = 0$, d'où la violation de la condition SPR. D'une certaine manière, la fonction $f(D) = -\log \min_{s \in \partial\mathcal{D}} \det G(s)$ joue le rôle de barrière de l'ensemble de stabilité \mathcal{S}_p . En effet, la fonction tend vers $+\infty$ lorsque les coefficients de $D(s)$ s'approchent de la frontière de \mathcal{S}_p , et elle admet un minimum égal à zéro lorsqu'on est au centre $D(s) = C(s)$. Pour cette raison, nous donnons à $C(s)$ le nom de *polynôme central*.

Un aspect intéressant de la condition SPR est la possibilité de l'exprimer comme une LMI en D et une variable additionnelle P , en supposant que C soit donnée. Cette LMI peut s'obtenir en formulant la relation SPR comme une condition de positivité d'une matrice polynomiale monovariante, ce qui est équivalent au lemme de Kalman-Yakubovich-Popov, voir par exemple [20]. En définissant

$$F(P) = \Pi^* \begin{bmatrix} d_0P & d_1P \\ d_1P & d_2P \end{bmatrix} \Pi$$

avec

$$\Pi = \begin{bmatrix} I_q & 0 & 0 & 0 \\ & \ddots & & \\ 0 & 0 & I_q & 0 \\ \hline 0 & I_q & 0 & 0 \\ & & \ddots & \\ 0 & 0 & 0 & I_q \end{bmatrix}$$

nous obtenons alors la condition LMI SPR

$$C'D + D'C - F(P) \succeq 0,$$

voir [25].

La projection de la LMI SPR sur l'ensemble des paramètres D permet de décrire une approximation convexe par l'intérieur de l'ensemble non-convexe \mathcal{S}_p des matrices polynomiales stables de degré p . La géométrie de cette approximation convexe dépend uniquement du choix du polynôme central $C(s)$, voir [26] pour plus de détails.

A titre d'illustration, sur la figure 3.3 je représente dans le cas de polynômes scalaires ($q = 1$) discrets ($d_0 = -1$, $d_1 = 0$, $d_2 = 1$) de troisième degré ($n = 3$) la projection de la LMI SPR dans l'espace tri-dimensionnel des coefficients (en supposant le coefficient de plus haut degré normalisé à 1) pour un polynôme central fixé $C(s) = s^3$. La region LMI s'écrit $\{x \in \mathbb{R}^3 : \exists u \in \mathbb{R}^6 : F(x, u) \succeq 0\}$ avec

$$F(x, u) = \begin{bmatrix} u_0 & \star & \star & \star \\ u_1 & -u_0 + u_3 & \star & \star \\ u_2 & -u_1 + u_4 & -u_3 + u_5 & \star \\ x_0 & x_1 - u_2 & x_2 - u_4 & 2 - u_5 \end{bmatrix}$$

où le vecteur de variables additionnelles $u \in \mathbb{R}^6$ correspond à la matrice symétrique P de dimension 3 introduite ci-dessus. On remarque que l'approximation convexe touche les deux plans ainsi que le point selle du paraboloïde hyperbolique de l'enveloppe de l'ensemble non-convexe \mathcal{S}_3 .

La condition LMI SPR est intéressante à plusieurs égards. Premièrement, elle permet de formuler une condition suffisante LMI pour déterminer la stabilité d'un polytope de matrices polynomiales, un problème difficile de commande robuste. La condition devient

$$C'D_i + D_i'C - F(P_i) \succeq 0, \quad i = 1, 2, \dots, N$$

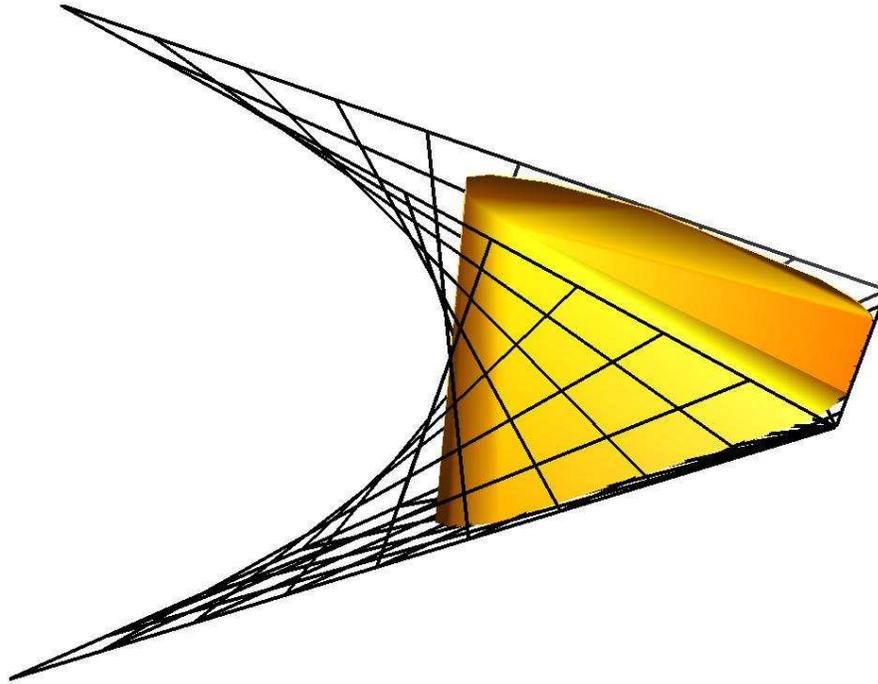


FIG. 3.3 – Approximation LMI SPR incluse dans l’enveloppe de l’ensemble non-convexe \mathcal{S}_3 .

où les $D_i(s)$ sont les sommets connus du polytope, et les inconnues sont C et les sommets P_i d’une fonction de Lyapunov polytopique (une classe particulière de fonction de Lyapunov dépendant des paramètres). Deuxièmement, la condition LMI SPR permet de formuler une condition suffisante LMI pour synthétiser des correcteurs d’ordre fixe ou structurés, un problème également difficile en général. A partir du moment où le polynôme central $C(s)$ est fixé, la condition LMI SPR devient linéaire en P et D , et donc en les paramètres du correcteur si ceux-ci entrent linéairement dans $D(s)$.

A l’aide de l’approche de Lyapunov pour la stabilité des équations différentielles et de la notion de fonction de stockage, il est possible d’étendre la condition LMI SPR à un contexte entrée-sortie et à des performances fréquentielles de type H_2 et H_∞ . La condition H_∞ peut d’ailleurs s’obtenir en considérant que le polynôme $D(s)$ est sujet à des incertitudes bornées en norme et en appliquant le théorème du faible gain. Ainsi, pour un polynôme

central $C(s)$ fixé, la condition

$$\begin{bmatrix} C'D + D'C - F(P) - \lambda C'C & N' \\ N & \lambda \gamma^2 I \end{bmatrix} \succeq 0$$

implique la performance $\|N(s)D^{-1}(s)\|_\infty < \gamma$. Il s'agit d'une condition LMI en N , D , λ (scalaire) et P .

De la même manière, il est possible d'appliquer la même condition pour la synthèse de correcteurs linéaires à paramètres variants (LPV). Une application de ces techniques à la commande de moteurs d'avions est d'ailleurs en développement dans le cadre d'un contrat de recherche avec l'industriel Snecma. La commande LPV semble particulièrement adaptée à la grande variabilité des diverses fonctions de transferts des moteurs dans l'enveloppe de vol [59, 60, 21].

3.5 Géométrie des mineurs de Hurwitz

Comme rappelé dans le chapitre 3, pour un polynôme $a(s) = a_0 + a_1s + \dots + a_ns^n$, l'ensemble de stabilité \mathcal{S}_n inclus dans l'espace \mathbb{R}^{n+1} des coefficients est l'intérieur d'un semi-algébrique basique. Une description explicite de cet ensemble peut être obtenue à l'aide des mineurs de la matrice de Hurwitz :

$$\mathcal{S}_n = \{a : f_i(a) > 0, i = 1, 2, \dots, n\}.$$

Chaque mineur $f_i(a)$ est un polynôme de degré i à $n + 1$ variables. On peut simplifier la description de l'ensemble en considérant des polynômes moniques, c'est-à-dire tels que $a_n = 1$, et surtout, dans le cas continu, en utilisant le critère de Liénard-Chipart, qui permet de formuler \mathcal{S}_n à l'aide d'un mineur sur deux seulement et en complétant avec des conditions de positivité sur les coefficients a_i [19, 51, 1]. Cependant, en général, la *complexité* de la formulation par mineurs est telle qu'il semble actuellement illusoire de pouvoir utiliser les logiciels de calcul scientifique de manière systématique pour résoudre par exemple le problème de stabilisation par correcteur d'ordre fixe pour des systèmes linéaires d'ordre moyen (disons supérieur à 7).

A titre d'exemple, considérons le problème de retour de sortie statique NN5 (7 états, 1 entrée, 2 sorties) de la collection COMPLIB compilée par Leibfritz [40]. On recherche deux gains scalaires x_1, x_2 stabilisants. Les

coefficients du polynôme caractéristique $a(s, x) = a_0(x) + a_1(x)s + \dots + a_6(x)s^6 + s^7$ (donnés ici avec 5 chiffres significatifs) sont :

$$\begin{aligned} a_0(x) &= 6,3000 + 2,2856x_1 \\ a_1(x) &= -448,72 + 49,523x_1 + 2,2856x_2 \\ a_2(x) &= 1,2196 + 7,2000 \cdot 10^{-2}x_1 + 49,523x_2 \\ a_3(x) &= 2249,5 - 2,6000x_1 + 7,2000 \cdot 10^{-2}x_2 \\ a_4(x) &= 458,43 - 2,6000x_2 \\ a_5(x) &= 96,515 \\ a_6(x) &= 10,171 \end{aligned}$$

et ils dépendent linéairement de x_1, x_2 (car il n'y a qu'une entrée).

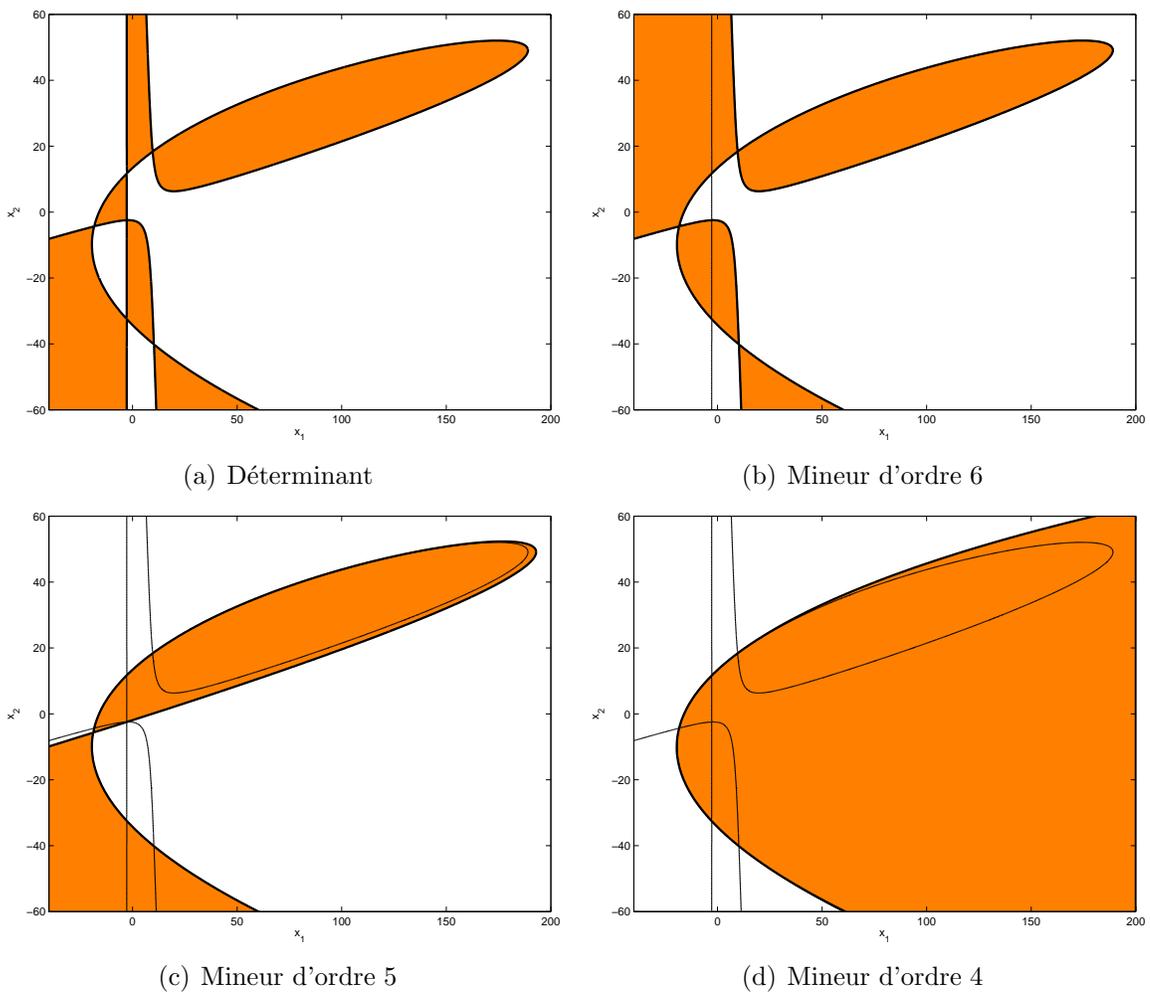


FIG. 3.4 – Problème NN5. Ensembles de positivité des mineurs (traits gras) et sextique déterminantale (traits fins).

Comme rappelé dans le paragraphe 3.2, le vecteur x est stabilisant si et

seulement si les mineurs principaux $f_1(x), \dots, f_7(x)$ de la matrice de Hurwitz sont strictement positifs. Ici $f_1(x) = a_6(x)$, $f_2(x) = a_5(x)a_6(x) - a_4(x)$, etc., et $f_7(x)$ est le déterminant de la matrice de Hurwitz, un polynôme de degré 6. Il s'agit d'une caractérisation algébrique du domaine de stabilité. Géométriquement, comme mentionné dans le paragraphe 3.2, ce même domaine est la coupe plane de \mathcal{S}_7 (l'ensemble non-convexe des polynômes stables de degré 7) par le sous-espace affine décrit ci-dessus.

Sur la figure 3.4(a) je représente dans le plan (x_1, x_2) la région de positivité du déterminant $f_7(x)$, qui est donc délimitée par une sextique. Sur la figure 3.4(b) je représente la région de positivité de $f_6(x)$, délimité par une quintique. Je représente également en traits fins superposés la sextique déterminantale $f_7(x) = 0$. Puisque $f_7(x) = a_0(x)f_6(x)$ est réductible (factorisable), on voit le rôle joué par le demi-plan $a_0(x) > 0$.

Sur la figure 3.4(c) je représente la région de positivité de $f_5(x)$, délimitée par une quartique. En particulier, on remarque que la sextique $f_7(x) = 0$ touche la quartique $f_5(x) = 0$ en certaines régions.

Sur la figure 3.4(d) je représente la région de positivité de la cubique $f_4(x)$, qui est également tangente avec $f_7(x)$ sur certains domaines.

Finalement, le domaine de positivité simultané de tous les mineurs, ou domaine de stabilité, est représenté sur la figure 3.5. On remarque que, pour cet exemple, 3 inégalités permettent d'isoler et donc de décrire le domaine de stabilité, à savoir $a_0(x) > 0$, $f_5(x) > 0$ et $f_6(x) > 0$. En général il semble difficile de simplifier systématiquement la description d'un tel ensemble semi-algébrique. Les différentes formes du critère de Liénard-Chipart semblent offrir les simplifications les plus générales [19].

Un autre aspect, mentionné dans les paragraphes 2.2 et 2.3, concerne la convexité cachée du domaine de stabilité. En général, ce domaine est non-convexe. Cependant, pour certains systèmes, ou certaines classes de systèmes, il peut devenir convexe. Pour l'exemple ci-dessus, cela semble être le cas, voir la figure 3.5, mais je ne suis pas capable de le démontrer. La question sous-jacente est celle de la convexité d'une coupe affine d'un ensemble semi-algébrique.

3.6 Critère de Hermite et relaxations LMI

Dans le paragraphe 3.2 nous avons mentionné le critère de Hermite comme formulation symétrique du critère de stabilité de Routh-Hurwitz :

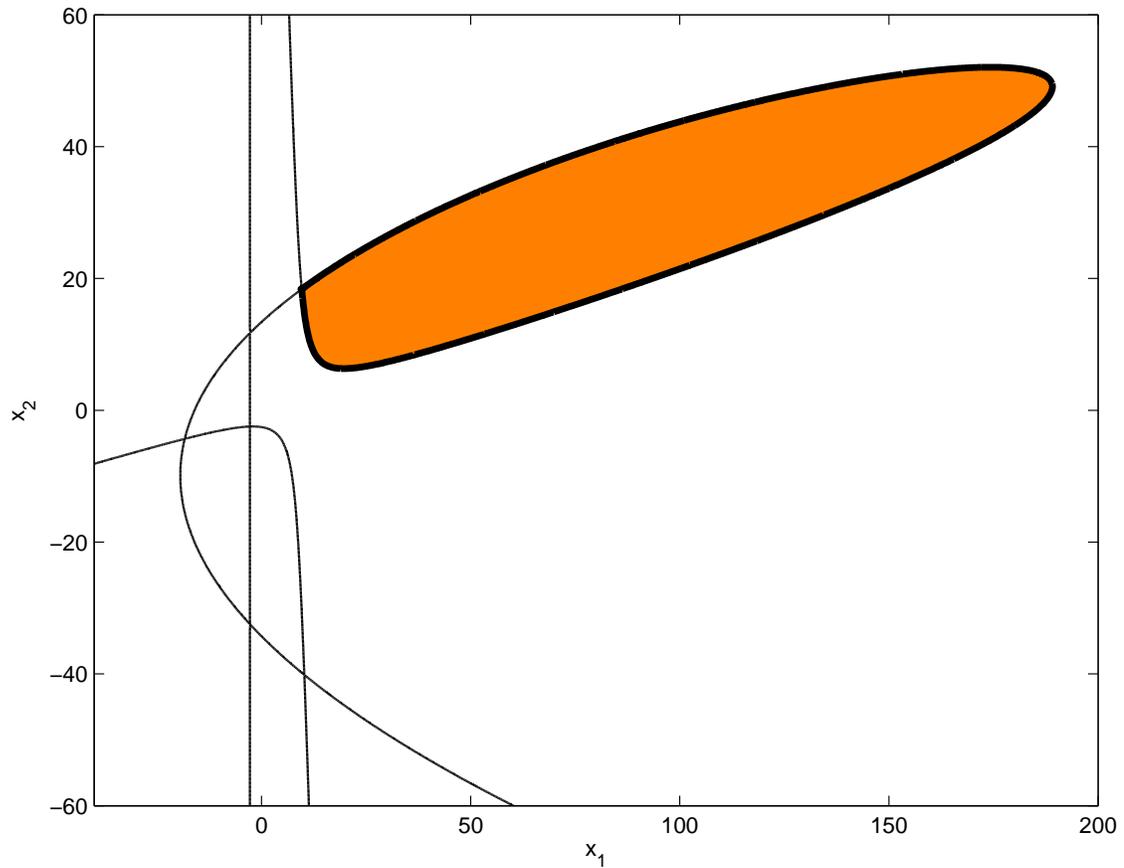


FIG. 3.5 – Problème NN5. Ensemble de stabilité.

le polynôme $a(s) = a_0 + a_1s + \dots + a_ns^n$ est stable si et seulement si

$$H(a) = \sum_i \sum_j a_i a_j H_{ij} \succ 0$$

où $H(a)$ est la matrice de Hermite quadratique en les coefficients a_i et les coefficients matriciels symétriques H_{ij} dépendent uniquement de la région de stabilité (et sont donc donnés).

Comme mentionné dans le paragraphe 3.3, le problème de synthèse de retour de sortie statique peut se formuler dans l'espace d'état comme la recherche d'une matrice K telle que la matrice $A + BKC$ soit stable, avec A , B , C matrices données. En considérant le polynôme caractéristique

$$a(s, K) = \det(sI - A - BKC) = \sum_{i=0}^n a_i(K)s^i,$$

où les coefficients $a_i(K)$ sont des polynômes multivariables en les entrées de K , et en lui appliquant le critère de Hermite, nous obtenons une inégalité

polynomiale matricielle PMI :

$$H(K) = \sum_i \sum_j a_i(K) a_j(K) H_{ij} \succ 0$$

directement en les coefficients de K . Pour résoudre ce problème numériquement, on peut ainsi appliquer par exemple la hiérarchie de relaxations LMI décrite dans le paragraphe 2.1.

L'approche classique de Lyapunov consiste à trouver non seulement la matrice K mais également une matrice symétrique P satisfaisant la BMI

$$(A + BKC)^* P + P(A + BKC) \prec 0, \quad P \succ 0.$$

Pour un système d'ordre 12 avec 1 entrée et 3 sorties, il n'y a que 3 scalaires à déterminer dans K mais 78 scalaires à déterminer dans P , ce qui complique considérablement la tâche.

Pour une discussion plus détaillée et des exemples illustratifs de cette approche Hermite-PMI par relaxations LMI, voir [29].

Pour terminer ce paragraphe, je désire mentionner à nouveau le problème de la convexité de l'ensemble de stabilité, déjà illustré sur un exemple d'ordre 7 dans le paragraphe précédent, voir la figure 3.5. Un exemple plus simple et plus explicite est le problème NN1 (3 états, 1 entrée, 2 sorties) de la collection COMPLIB, dont les coefficients du polynôme caractéristique sont :

$$\begin{aligned} a_0(x) &= x_2 \\ a_1(x) &= -13 - 5x_1 + x_2 \\ a_2(x) &= x_1 \\ a_3(x) &= 1, \end{aligned}$$

où les scalaires x_1 et x_2 sont les entrées de la matrice K définie ci-dessus. L'ensemble de stabilité est décrit par la région de positivité de la matrice de Hermite de ce polynôme

$$H(K) = \begin{bmatrix} -13x_2 - 5x_1x_2 + x_2^2 & x_2 & 0 \\ x_2 & x_1 & 0 \\ 0 & 0 & -13x_1 - x_2 - 5x_1^2 + x_1x_2 \end{bmatrix} \succ 0$$

et c'est une PMI quadratique en x . Comme mentionné dans le paragraphe 2.2, en général les PMI définissent des ensembles non-convexes. Ici cependant, l'ensemble de stabilité est une branche hyperbolique convexe, voir la figure 3.6.

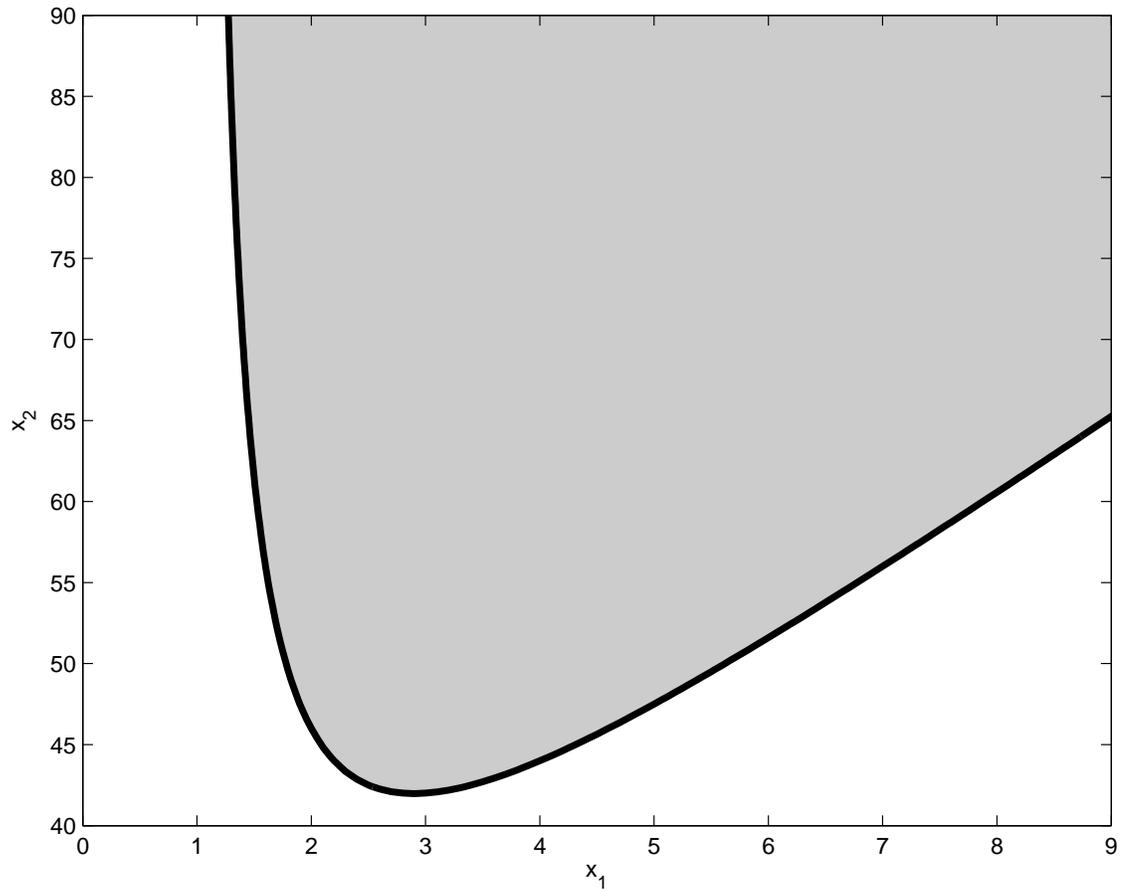


FIG. 3.6 – Problème NN1. Ensemble de stabilité.

Manuellement, on peut même obtenir une représentation LMI de la PMI ci-dessus :

$$\begin{bmatrix} -1 + x_1 & 1 \\ 1 & -1 - \frac{5}{18}x_1 + \frac{1}{18}x_2 \end{bmatrix} \succ 0.$$

Cependant, je ne sais pas comment généraliser cette construction aux systèmes d'ordre supérieur à 3.

3.7 Conclusions

La combinaison entre programmation semidéfinie et géométrie algébrique, ou plus simplement entre LMI et polynômes, permet une approche constructive à la résolution de problèmes fondamentaux en automatique linéaire et en commande robuste, comme par exemple le retour de sortie statique où la synthèse de correcteurs H_∞ d'ordre fixe. L'optimisation non-linéaire non-convexe non-lisse offre également une approche plus directe et pragmatique

à la résolution numérique de ces problèmes.

Bien que cela ne soit pas développé ici, la programmation semidéfinie et la géométrie algébrique fournissent également, une fois combinées, des outils efficaces pour résoudre de nombreux problèmes de commande non-linéaire, voir par exemple [58] ou encore [47].

Chapitre 4

Méthodes numériques et outils logiciels

Les principaux logiciels de calcul scientifique (Maple, Mathematica, Matlab) ont la capacité d'être enrichis par des modules extérieurs, paquetages (packages en anglais) ou boîtes à outils (toolboxes). Certains modules sont développés par des informaticiens professionnels et sont généralement proposés à la vente. D'autres sont conçus par des académiques et sont généralement disponibles librement (freeware), contribuant ainsi constructivement à l'avancement des recherches, et permettant également d'assurer une certaine visibilité à des travaux parfois difficiles à appréhender sans outils adéquats.

Dans ce chapitre je survole ma contribution à la conception et au développement de divers outils en relation avec mes thèmes de recherche. J'en profite également pour décrire l'historique de mes diverses collaborations dans ce cadre.

4.1 Polynomial Toolbox

A partir de 1996 j'ai eu l'occasion de collaborer avec Vladimír Kučera et Michael Šebek à l'Institut de Théorie de l'Information et d'Automatique de l'Académie Tchèque des Sciences à Prague. En 1996, Šebek était impliqué dans le développement d'une boîte à outils Matlab permettant de manipuler les polynômes et les équations polynomiales pour résoudre des problèmes de commande : la *Polynomial Toolbox* [56]. Il s'agissait d'un outil développé depuis 1989 en collaboration avec Huibert Kwakernaak de l'Université de Twente à Enschede aux Pays-Bas. Une version 1.4 relativement stable mais pas très facile d'utilisation était alors disponible dans le

domaine public. Ma contribution a consisté à développer une version 2.0 beaucoup plus simple d'utilisation et en particulier à intégrer diverses fonctions de résolution d'équations linéaires polynomiales diophantiennes et d'équations quadratiques de factorisation spectrale polynomiale. A terme, cela a donné lieu à divers travaux de nature plus ou moins théorique mêlant algèbre et analyse numérique, regroupés en 1999 dans mon manuscrit de thèse de doctorat tchèque [23]. Le développement de la version 2.0 de la Polynomial Toolbox s'est fait en collaboration avec Martin Hromčík, Soňa Pejchová et Rens Strijbos, et dans le cadre d'un projet d'excellence européen dénommé EUROPOLY regroupant certains experts en techniques polynomiales pour la commande. Pour diverses raisons, Kwakernaak et Šebek voulurent que la version 2.0 de la Polynomial Toolbox soit commercialisée. Une société dénommée PolyX Ltd fut créée en 1998 pour distribuer le logiciel. Plus tard j'ai également contribué à la version 2.5, incorporant dès 2000 diverses fonctions de commande robuste. En 2007, un projet de version 3.0 est toujours en cours d'élaboration.

4.2 Solveurs LMI et BMI

A la fin des années 1980 au LAAS-CNRS l'équipe de Jacques Bernussou (et ses étudiants José Geromel, Pedro Luis Dias Peres et Denis Arzelier) effectua des travaux pionniers sur l'optimisation convexe en commande robuste. On consultera par exemple le chapitre introductif de [11] pour une historique de ces travaux. Dans la lignée, les thèses de doctorat d'Olivier Bachelier, Jamal Daafouz, João Gomes da Silva Junior, Yann Labit, Dimitri Peaucelle et Christelle Pittet, pour ne citer que ces travaux parmi tant d'autres, utilisèrent le formalisme des inégalités matricielles linéaires (LMI) pour résoudre divers problèmes de commande (placement robuste de pôles, coût garanti, commande en présence de saturations, commande multi-objectif). Ce fut également le cadre de ma thèse au LAAS-CNRS [24]. Le groupe de recherche CSC (Commande des Systèmes Continus, créé en 1995 par Jacques Bernussou suite à la restructuration des activités en automatique) qui est devenu ensuite le groupe MAC (Méthodes et Algorithmes en Commande, dirigé par Jacques Bernussou de 1999 à 2002 puis par Sophie Tarbouriech de 2003 à 2006), a toujours été un utilisateur intensif de l'optimisation semi-définie et des LMI.

A la fin des années 1990, le principal outil dédié à la résolution de LMI

pour l'automatique était la LMI Control Toolbox de Matlab, conçue en 1993 par Pascal Gahinet initialement pour Scilab, suite à une visite d'Arkadi Nemirovski à l'INRIA Rocquencourt. Suite à de bons contacts initiés en 2000 avec Jos Sturm, le concepteur de SeDuMi, un solveur conique disponible depuis 1998 et réputé pour sa rapidité, nous avons entrepris avec Dimitri Peaucelle le développement d'une interface Matlab pour SeDuMi dédiée aux problèmes d'automatique. Le logiciel domaine public résultant, *SeDuMi Interface* [44], fut assez rapidement concurrencé et surpassé par une autre interface, développée dès 2001 par Johan Löfberg de l'Université de Linköping en Suède, et dénommée ironiquement YALMIP - Yet Another LMI Parser [48]. Parmi les nombreux avantages de YALMIP sur SeDuMi Interface, on notera la possibilité d'utiliser d'autres solveurs coniques que SeDuMi (comme par exemple CSDP, DSDP, PENSDP, SDPA, SDTP3) ainsi qu'une programmation orientée objet qui assure une interface utilisateur particulièrement simple et intuitive. A présent YALMIP est devenu la référence incontournable dans le domaine des interfaces Matlab pour les problèmes LMI. Sur la période 2003-2006, Dimitri Peaucelle a dirigé un projet financé par le LAAS-CNRS et portant sur le développement de routines Matlab utilisant YALMIP pour définir un certain nombre de problèmes LMI de commande robuste et d'évaluation de performances [53].

En 2003 ma collaboration avec Michal Kočvara de l'Académie Tchèque des Sciences a permis la mise au point, conjointement avec Michael Stingl de l'Université d'Erlangen à Nuremberg en Allemagne, du solveur PENBMI [39, 40] qui est le premier logiciel domaine public permettant de résoudre numériquement les problèmes non-convexes BMI (inégalités bilinéaires matricielles). PENBMI est une implémentation d'une méthode de pénalité et de Lagrangien augmenté, qui n'assure que la résolution locale de BMI (convergence vers un point satisfaisant les conditions d'optimalité du premier ordre). PENBMI est interfacé sous YALMIP.

4.3 GloptiPoly

En fin 2000, lors de mon intégration comme chargé de recherche au LAAS-CNRS, j'ai eu le privilège de pouvoir démarrer une collaboration avec Jean-Bernard Lasserre, qui venait alors de finaliser ses travaux sur les relaxations LMI pour l'optimisation polynomiale, et leur interprétations primale (théorie des moments) et duale (représentation de polynômes po-

sitifs comme sommes de carrés) [45]. Nous avons intégré ces résultats dans un logiciel Matlab de domaine public dénommé GloptiPoly (Global Optimization on Polynomials), que nous avons largement distribué dans les communautés d'automatique, d'optimisation et de mathématiques appliquées [27]. GloptiPoly permet de modéliser les relaxations LMI primales (sur les moments, brièvement décrites dans le paragraphe 2.4) d'un problème d'optimisation polynomiale, et d'appeler ensuite le solveur conique SeDuMi pour résoudre numériquement ces relaxations. Simultanément et indépendamment, le groupe de John Doyle au California Institute of Technology, et notamment son doctorant Pablo Parrilo [52], développa un logiciel concurrent focalisant sur l'interprétation duale (sommes de carrés de polynômes), dénommé SOSTOOLS [57].

La version 1 de GloptiPoly, distribuée en début 2002, permettait uniquement d'obtenir une hiérarchie de bornes sur l'optimum global. Arnold Neumaier, de l'Université de Vienne en Autriche, nous suggéra alors d'implémenter un algorithme d'extraction des solutions. La version 2 de GloptiPoly, distribuée en début 2003, contient un algorithme original d'extraction et de détection d'optimalité globale qui a permis de se démarquer nettement de SOSTOOLS [28]. GloptiPoly suscita l'intérêt d'un certain nombre de chercheurs, comme par exemple Fredrik Kahl de l'Université de Lund en Suède, un spécialiste de vision par ordinateur. Ma collaboration avec Kahl se concrétisa par le prix David Marr du meilleur article à la conférence ICCS de Pékin en 2005 [36]. En 2004, Johan Löfberg intégra les algorithmes de GloptiPoly 2 dans le module moments de YALMIP 3.

Une version 3 de GloptiPoly est actuellement en cours d'élaboration en collaboration avec Lasserre et Löfberg. Cette version permettra de résoudre le problème généralisé des moments (plusieurs mesures dont les moments à déterminer sont reliés linéairement), avec applications notamment en évaluation de performance des processus dynamiques stochastiques et en commande optimale non-linéaire avec contrainte sur l'état [47]. Cette orientation vers les problèmes des moments nous permet de nous démarquer encore plus des logiciels concurrents comme SOSTOOLS et YALMIP.

4.4 HIFOO

Au printemps 2003 lors d'un séminaire de l'Institut de Mathématiques Appliquées (IMA) de l'Université de Minnesota à Minneapolis, j'ai démarré

une collaboration avec Michael Overton de l'Université de New York, dans le but d'appliquer sur des problèmes d'automatique les algorithmes d'optimisation non-convexe non-lisse qu'il développait alors depuis plusieurs années avec James Burke de l'Université de Washington et Adrian Lewis de l'Université Simon Fraser de Vancouver, maintenant à l'Université de Cornell. Assez rapidement nous nous sommes focalisés sur la synthèse de correcteurs H_∞ d'ordre fixé pour les systèmes linéaires [14]. En début 2006 nos travaux se sont concrétisés par la première version d'un logiciel Matlab de domaine public dénommé HIFOO (H_∞ Fixed-Order Optimization) [13].

Diverses extensions sont prévues à HIFOO, dans le cadre de la thèse de doctorat de Marc Millstone à l'Université de New York : contraintes structurelles sur le correcteur, commande multi-objectif. HIFOO est déjà utilisé dans un contexte industriel, par exemple dans le cadre de l'ERT Enroulement de Dominique Knittel à l'Université de Strasbourg [37, 38].

Chapitre 5

Perspectives

Dans ce chapitre je décris brièvement les directions que j’essaye de donner actuellement à mes travaux de recherche.

5.1 Développements de logiciels

Le *développement de logiciels de domaine public* me semble être une nécessité pour permettre à la fois une meilleure compréhension des avancées théoriques et assurer une visibilité aux travaux de recherche. Ainsi depuis 2005 je suis responsable du comité technique CACSD (Computer-Aided Control System Design) de IEEE, fondé en 1982 [31]. La nouvelle version 3 de GloptiPoly ainsi que la nouvelle version 2 de HIFOO sont pour le moment mes priorités.

La version 3 de GloptiPoly, dont le développement a démarré en 2004, a été distribuée en 2006 de manière restreinte à certains collègues. Une version domaine public a été annoncée lors d’un séminaire de l’Institute of Mathematics and its Applications de Minneapolis en Janvier 2007, et elle est à présent disponible librement sur ma page internet. Il s’agit d’une refonte fondamentale de GloptiPoly 2 utilisant la programmation orientée objet. Afin de se démarquer nettement du logiciel concurrent SOSTOOLS (conçu par P. Parrilo et son équipe en 2002 et permettant de résoudre des problèmes de sommes de carrés polynomiaux [57]), l’accent a été porté sur la modélisation et la résolution de problèmes généralisés des moments [46], une vaste extension du problème classique des moments, avec applications en optimisation, probabilités, finance, commande, traitement du signal, chimie, cristallographie, tomographie, etc.

La version 2 de HIFOO est une extension de la version 1.5 dispo-

nible actuellement sur la page internet de M. Overton. Cette extension est développée dans le cadre du doctorat de Marc Millstone au Courant Institute of Mathematical Sciences de l'Université de New York. Les nouvelles fonctionnalités de la version 2 sont encore à discuter, mais d'ores et déjà les contraintes de type structurelles (creusité des matrices du correcteur, permettant d'imposer une structure compagne ou bloc-diagonale par exemple) et les spécifications multiples (commande multiobjectif, stabilisation simultanée) sont prises en compte.

A moyen terme j'aimerais parfaire mes connaissances en programmation mathématique et optimisation semidéfinie, dans le but de pouvoir comprendre les problèmes numériques inhérents à la résolution de relaxations LMIs pour l'*optimisation polynomiale*.

J'aimerais également mieux comprendre les interactions existant entre algèbre linéaire numérique et géométrie algébrique, dans le but d'obtenir des résultats originaux en *algèbre numérique polynomiale*, dans la lignée des travaux fondateurs de Stetter [65], mais applicables en automatique. Les travaux de doctorat de Juan-Carlos Zúñiga-Anaya sont une première tentative [69].

A titre d'exemple illustratif, le calcul des racines d'un polynôme univarié s'effectue sous Matlab à l'aide de l'instruction `roots`. A partir des coefficients du polynôme dans la base standard des monômes, cette fonction construit tout d'abord la matrice compagne associée. Les valeurs propres de cette matrice sont les racines recherchées du polynôme. Avec Matlab, le calcul des valeurs propres d'une matrice se fait à l'aide de la fonction `eig`, elle-même basée sur des routines LAPACK développées depuis plusieurs décennies. Le conditionnement des valeurs propres (sensibilité des valeurs propres à des perturbations affectant la matrice), et donc la qualité du résultat obtenu par `eig`, dépend du conditionnement de la matrice des vecteurs propres (rapport des valeurs singulières extrêmes). Ce conditionnement est généralement élevé (mauvais) dans le cas d'une matrice compagne. La matrice est alors équilibrée par une transformation de similarité diagonale à l'aide de la fonction `balance`. Cela a pour résultat d'améliorer le conditionnement des vecteurs propres sans changer les valeurs propres. La fonction `eig` est donc appliquée sur une matrice compagne équilibrée. Comme indiqué dans [66, Chapitre 55] cette méthode semble efficace pour des polynômes de degré faible (inférieur à 100) avec des racines simples. Pour un système linéaire dont la dynamique est régie par un polynôme, la

matrice compagne correspond à une réalisation en espace d'état. Il serait intéressant d'étudier l'interprétation de l'équilibrage de la matrice compagne en termes de trajectoires dans l'espace d'état. On pourrait également étudier cette technique de pré-conditionnement dans le cadre des équations linéaires Diophantiennes de placement de pôles, qui sont généralement mal conditionnées.

5.2 Conditionnement et stabilité numérique

Comme cela a été constaté dans le chapitre 2, la frontière entre convexité et non-convexité devient de plus en plus floue dans le cadre de la programmation semi-définie. Pour un problème d'optimisation, il me semble réducteur d'associer convexité à facilité de résolution. Certains problèmes non-convexes sont facilement solvables avec l'optimisation convexe, et inversement certains problèmes convexes sont difficiles à résoudre numériquement. La notion générale de *conditionnement*, déjà évoquée dans le paragraphe 5.1, semble également jouer un rôle central dans la mesure de la difficulté de résolution d'un problème, qu'il soit convexe ou non.

A ma connaissance, il n'existe pas à l'heure actuelle de solveur LMI numériquement stable, au sens où la solution calculée pour un problème bien conditionné correspond à la solution exacte du problème perturbé [32]. Il semble également que beaucoup de problèmes LMI (et en particulier les problèmes provenant de l'automatique) soient mal conditionnés, voire mal posés, et ceci quelle que soit la mesure approchée de conditionnement que l'on utilise [18]. S'il on combine instabilité numérique (de l'algorithme de résolution) et mauvais conditionnement (des problèmes LMI), on ne sera pas surpris d'apprendre que les ingénieurs automaticiens soient parfois sceptiques au sujet de l'applicabilité des méthodes d'analyse et de synthèse robustes basées sur les LMI.

Il serait intéressant de disposer d'une estimation simple et peu coûteuse du conditionnement d'un problème LMI, ou plus modestement d'une classe de problèmes LMI. Une telle estimation doit être calculable en un temps négligeable par rapport au temps nécessaire à la résolution du problème LMI. De tels estimés sont disponibles dans le cas du conditionnement de matrices, pour la résolution de systèmes linéaires d'équations [32].

5.3 LMI et géométrie algébrique pour l'automatique

Finalement, comme cela été mentionné déjà dans la conclusion du chapitre 3, il me semble que les progrès récents en géométrie algébrique réelle pourraient bénéficier aux automaticiens à la recherche d'outils fiables et puissants permettant de résoudre des problèmes difficiles, comme par exemple le retour de sortie statique, la commande multiobjectif, ou la synthèse systématique de lois de commande pour les systèmes non-linéaires, par exemple via la notion de platitude [17]. L'application de la géométrie algébrique en automatique, déjà entreprise par certains groupes, a été mentionnée par Rudolf Kalman lors de sa conférence plénière du congrès international IFAC de Prague en 2005 comme un des domaines de recherche les plus prometteurs. Quelques travaux préliminaires ont été effectués dans le cadre de la commande optimale polynomiale [47], mais l'applicabilité de l'approche et les liens existant avec les techniques de commande classique (par exemple commande LQ dans le cas linéaire) restent encore à démontrer.

Il est à noter que la nouvelle version 11 du logiciel Maple comporte de nouvelles routines beaucoup plus performantes pour le calcul de bases de Gröbner, un outil fondamental en géométrie algébrique. On peut ici tracer un parallèle avec l'algèbre linéaire : la factorisation QR puis SVD (et les factorisations orthogonales en général, qui sont numériquement stables) furent proposées dans les années 1960 comme alternatives à l'élimination Gaussienne (potentiellement instable), et ont révolutionné le calcul scientifique. Comme mentionné ci-dessous dans le paragraphe 5.1, une algèbre polynomiale numérique semble voir le jour, et dans quelques années elle devrait pouvoir s'appliquer pour résoudre des problèmes d'automatiques industriels.

Dans un autre registre, les liens intimes existant entre géométrie algébrique réelle et programmation semidéfinie, mis en évidence par les travaux de Lasserre [45] et Parrilo [52], semblent permettre, du moins en théorie, la résolution de problèmes d'automatique réputés difficiles jusqu'alors. Pour plus de détails, on pourra par exemple consulter le survol récent de Scherer [63]. Une bonne source d'inspiration, ainsi qu'une bonne introduction au domaine, est la documentation de l'outil de modélisation YALMIP développé par J. Löfberg [48], qui comporte des modules pour les sommes de carrés de polynômes et les problèmes des moments.

Dans un cadre plus mathématique, un problème ouvert proposé dans [34, Problem 2] concerne la description d'un ensemble semi-algébrique à l'aide d'un nombre minimum d'inégalités polynomiales et les liens existant avec les relaxations LMI des problèmes d'optimisation polynomiale mentionnées dans le paragraphe 2.4. On peut citer également la généralisation à plus de 2 variables de la condition de rigidité convexe de [30] mentionnée auparavant dans le paragraphe 2.1, qui permettrait de caractériser entièrement les ensembles convexes semi-algébriques admettant une représentation LMI. Un problème également ouvert concerne l'existence d'une représentation par projection d'une LMI (donc avec variables additionnelles, voir la discussion de la fin du paragraphe 2.1) de tout ensemble convexe semi-algébrique, c'est-à-dire à frontière polynomiale par morceaux. Dans le même ordre d'idée, A. S. Lewis aimerait savoir si le cône des matrices positives semi-définies de dimension 3 peut s'exprimer à l'aide d'une projection de cônes quadratiques de Lorentz. Finalement, pour faire le lien avec les aspects numériques mentionnés ci-dessus, il serait intéressant d'implémenter un algorithme permettant de détecter la rigidité convexe d'un ensemble semi-algébrique et de construire numériquement la représentation LMI associée.

Bibliographie

- [1] J. Ackermann et al. Robust control. Second edition, Springer, 2002. First edition published in 1993.
- [2] D. Arzelier, B. Clément, D. Peaucelle. Multi-objective $H_2 / H_\infty /$ Impulse-to-peak control of a space launch vehicle. European Journal of Control, Vol. 12, No. 1, pp. 57–70, 2006.
- [3] B. R. Barmish. New tools for robustness of linear systems. MacMillan, 1994.
- [4] S. Basu, R. Pollack, M.-F. Roy. Algorithms in real algebraic geometry. Springer, 2005.
- [5] A. Ben-Tal, A. Nemirovskii. Lectures on modern convex optimization. SIAM, 2001.
- [6] S. P. Bhattacharyya, H. Chapellat, L. H. Keel. Robust control : the parametric approach. Prentice Hall, 1995.
- [7] V. D. Blondel. Simultaneous stabilization of linear systems. Springer, 1994.
- [8] V. D. Blondel, J. N. Tsitsiklis. A survey of computational complexity results in systems and control. Automatica, Vol. 36, No. 9, pp. 1249–1274, 2000.
- [9] S. P. Boyd, V. Balakrishnan, P. Kabamba. A bisection method for computing the H_∞ norm of a transfer matrix and related problems. Mathematics of Control, Signals, and Systems, Vol. 2, No. 3, pp. 207–219, 1989
- [10] S. P. Boyd, C. H. Barratt. Linear controller design : limits of performance. Prentice Hall, 1991.
- [11] S. P. Boyd, L. El Ghaoui, E. Feron, V. Balakrishnan. Linear matrix inequalities in systems and control theory. SIAM, 1994.

- [12] S. P. Boyd, L. Vandenberghe. Convex optimization. Cambridge University Press, 2005.
- [13] J. V. Burke, D. Henrion, A. S. Lewis, M. L. Overton. HIFOO : a Matlab package for fixed-order controller design and H_∞ optimization. Proceedings of the IFAC Symposium on Robust Control Design, Toulouse, France, 2006.
- [14] J. V. Burke, D. Henrion, A. S. Lewis and M. L. Overton. Stabilization via nonsmooth, nonconvex optimization. IEEE Transactions on Automatic Control, Vol. 51, No. 11, pp. 1760–1769, 2006.
- [15] A. C. Dixon. Note on the reduction of a ternary quantic to a symmetrical determinant. Proceedings of the Cambridge Philosophical Society, Vol. 11, pp. 350–351, 1902.
- [16] J. C. Doyle, K. Glover, P. Khargonekar, B. Francis. State-space solutions to standard H_2 and H_∞ control problems. IEEE Transactions on Automatic Control, Vol. 34, No. 8, pp. 831–846, 1989.
- [17] M. Fliess, J. Lévine, P. Martin, P. Rouchon. Sur les systèmes non-linéaires différentiellement plats. Comptes Rendus de l'Académie des Sciences de Paris, Vol. I-315, pp. 619–624, 1992.
- [18] R. M. Freund, F. Ordoñez, K. C. Toh. Behavioral measures and their correlation with IPM iteration counts on semidefinite programming problems. Mathematical Programming, Vol. 109, No. 2-3, pp. 445–475, 2007.
- [19] F. R. Gantmacher. Theory of matrices. Vol. 2, AMS, 2000. Original version published in Russian in 1959.
- [20] Y. Genin, Y. Hachez, Y. Nesterov, R. Ştefan, P. Van Dooren, S. Xu. Positivity and linear matrix inequalities. European Journal of Control, Vol. 8, No. 3, pp. 275–298, 2002.
- [21] W. Gilbert, D. Henrion, J. Bernussou, D. Boyer. Polynomial LPV synthesis applied to turbofan engines. Proceedings of the IFAC Symposium on Automatic Control in Aerospace, Toulouse, France, 2007.
- [22] K. C. Goh, M. G. Safonov, G. P. Papavassilopoulos. Global optimization for the biaffine matrix inequality problem. Journal of Global Optimization, Vol. 7, pp. 365–380, 1995.

- [23] D. Henrion. Reliable algorithms for polynomial matrices. Ph. D. Thesis, Institute of Information Theory and Automation, Czech Academy of Sciences, Prague, Czech Republic, 1998.
- [24] D. Henrion. Stabilité des systèmes linéaires incertains à commande contrainte. Thèse de doctorat, Institut National des Sciences Appliquées de Toulouse, France, 1999.
- [25] D. Henrion, D. Arzelier, D. Peaucelle. Positive polynomial matrices and improved LMI robustness conditions. *Automatica*, Vol. 39, No. 8, pp. 1479-1485, 2003.
- [26] D. Henrion, M. Šebek, V. Kučera. Positive polynomials and robust stabilization with fixed-order controllers. *IEEE Transactions on Automatic Control*, Vol. 48, No. 7, pp. 1178-1186, 2003.
- [27] D. Henrion, J. B. Lasserre. GloptiPoly : global optimization over polynomials with Matlab and SeDuMi. *ACM Transactions on Mathematical Software*, Vol. 29, No. 2, pp. 165-194, 2003.
- [28] D. Henrion, J. B. Lasserre. Detecting global optimality and extracting solutions in GloptiPoly. In D. Henrion, A. Garulli (Editors). *Positive polynomials in control. Lecture Notes on Control and Information Sciences*, Vol. 312, Springer, 2005.
- [29] D. Henrion, J. B. Lasserre. Convergent relaxations of polynomial matrix inequalities and static output feedback. *IEEE Transactions on Automatic Control*, Vol. 51, No. 2, pp. 192-202, 2006.
- [30] J. W. Helton, V. Vinnikov. Linear matrix inequality representation of sets. *Communications on Pure and Applied Mathematics*. Vol. 60, No. 5, pp. 654-674, 2007.
- [31] C. J. Herget, A. J. Laub. Special issue on computer-aided design of control systems. *IEEE Control Systems Magazine*, Vol. 2, No. 4, 1982.
- [32] N. J. Higham. Accuracy and stability of numerical algorithms, Second edition, SIAM, 2002. First edition published in 1996.
- [33] J. B. Hiriart-Urruty, M. Torki. Permanently going back and forth between the quadratic world and the convexity world in optimization. *Applied Mathematics and Optimization*, Vol. 45, pp. 169-184, 2002.

- [34] J. B. Hiriart-Urruty. Potpourri of conjectures and open questions in nonlinear analysis and optimization. *SIAM Review*, Vol. 49, No. 2, pp. 255-273, 2007.
- [35] T. Kailath. *Linear systems*. Prentice Hall, 1980.
- [36] F. Kahl, D. Henrion. Globally optimal estimates for geometric reconstruction problems. *International Journal of Computer Vision*, Vol. 74, No. 1, pp. 3-15, 2007.
- [37] D. Knittel, M. Vedrines, D. Henrion, P. Pagilla. Robust H_∞ fixed order control strategies for large scale web winding systems. *Proceedings of the IEEE Conference on Control Applications*, Munich, Germany, 2006.
- [38] D. Knittel, M. Vedrines, D. Henrion, P. Pagilla. Fixed-order H_∞ decentralized control with model based feedforward for elastic web winding systems. *Proceedings of the IEEE Industry Applications Society Conference*, Tampa, USA, 2006.
- [39] M. Kočvara, M. Stingl. PENNON : a code for convex nonlinear and semidefinite programming. *Optimization Methods and Software*, Vol. 8, No. 3, pp. 317–333, 2003.
- [40] M. Kočvara, F. Leibfritz, M. Stingl, D. Henrion. A nonlinear SDP algorithm for static output feedback problems in COMPLIB. *Proceedings of the IFAC World Congress on Automatic Control*, Prague, Czech Republic, 2005.
- [41] V. Kučera. *Discrete linear control - the polynomial equation approach*. John Wiley and Sons, 1979.
- [42] V. Kučera. *Analysis and design of discrete linear control systems*. Prentice Hall, 1991.
- [43] H. Kwakernaak. Robust control and H_∞ optimization - tutorial paper. *Automatica*, Vol. 29, No. 2, pp. 255–273, 1993.
- [44] Y. Labit, D. Peaucelle, D. Henrion. SeDuMi Interface 1.02 : a tool for solving LMI problems with SeDuMi. *Proceedings of the IEEE Conference on Computer-Aided Control System Design*, Glasgow, UK, 2002.
- [45] J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, Vol. 11, No. 3, pp. 796–817, 2001.

- [46] J. B. Lasserre. A semidefinite programming approach to the generalized problem of moments. *Mathematical Programming*, Vol. 112, No. 1, pp. 65-92, 2008.
- [47] J. B. Lasserre, C. Prieur, D. Henrion. Nonlinear optimal control : numerical approximation via moments and LMI relaxations. *Proceedings of the joint IEEE Conference on Decision and Control and European Control Conference*, Sevilla, Spain, 2005.
- [48] J. Löfberg. YALMIP : a toolbox for modeling and optimization in Matlab. *Proceedings of the IEEE Conference on Computer Aided Control System Design*, Taipei, Taiwan, 2004.
- [49] The MathWorks, Inc. *Control System Toolbox for Matlab*. Version 6.2.1, 2005.
- [50] Y. Nesterov, A. Nemirovskii. *Interior point polynomial algorithms in convex programming*, SIAM, 1994.
- [51] P. C. Parks, V. Hahn. *Stability theory*. Prentice Hall, 1993. Original version published in German in 1981.
- [52] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, Vol. B-96, pp. 293–320, 2003.
- [53] D. Peaucelle, D. Arzelier. Robust multi-objective control toolbox. *Proceedings of the IEEE Symposium on Computer-Aided Control System Design*, Munich, Germany, 2006.
- [54] J. W. Polderman, J. C. Willems. *Introduction to mathematical systems theory : a behavioral approach*. Springer, 1998.
- [55] B. T. Polyak. Convexity of quadratic transformations and its use in control and optimization. *Journal of Optimization Theory and Applications*, Vol. 99, pp. 552–583, 1998.
- [56] PolyX, Ltd. *The Polynomial Toolbox for Matlab*. Version 2.5. 1998.
- [57] S. Prajna, A. Papachristodoulou, P.A. Parrilo. Introducing SOS-TOOLS : a general purpose sum of squares programming solver. *Proceedings of IEEE Conference on Decision and Control*, Las Vegas, USA, 2002.
- [58] S. Prajna, P. A. Parrilo, A. Rantzer. Nonlinear control synthesis by convex optimization. *IEEE Transactions on Automatic Control*, Vol. 49, No. 2, pp. 310–314, 2004

- [59] L. Reberga. Commande robuste multivariable des turboréacteurs. Thèse de doctorat, Institut National des Sciences Appliquées de Toulouse, France, 2005.
- [60] L. Reberga, D. Henrion, J. Bernussou, F. Vary. LPV modeling of a turbofan engine. Proceedings of the IFAC World Congress on Automatic Control, Prague, Czech Republic, 2005.
- [61] R. T. Rockafellar. Lagrange multipliers and optimality. *SIAM Review*, Vol. 35, pp. 183–238, 1993.
- [62] H. H. Rosenbrock. State-space and multivariable theory. Nelson, 1970.
- [63] C. W. Scherer. LMI relaxations in robust control. *European Journal of Control*, Vol. 12, No. 1, pp. 3-29, 2006.
- [64] S. Skogestad and I. Postlethwaite. Multivariable feedback control : analysis and design. Second edition, Wiley, 2005. First edition published in 1996.
- [65] H. J. Stetter. Numerical polynomial algebra. SIAM, 2004.
- [66] L. N. Trefethen, M. Embree. Spectra and pseudospectra : the behavior of nonnormal matrices and operators. Princeton University Press, 2005.
- [67] V. Vinnikov. Self-adjoint determinantal representations of real plane curves. *Mathematische Annalen*, Vol. 296, pp. 453–479, 1993.
- [68] K. Zhou, J. C. Doyle, K. Glover. Robust and optimal control. Prentice Hall, 1996.
- [69] J. C. Zúñiga-Anaya. Algorithmes numériques pour les matrices polynomiales avec applications en commande. Thèse de doctorat, Institut National des Sciences Appliquées de Toulouse, France, 2005.