Optimization on linear matrix inequalities for polynomial systems control

Didier Henrion[1,2]

Draft lecture notes of June 10, 2013

[1]LAAS-CNRS, Univ. Toulouse, France. henrion@laas.fr
[2]Fac. Elec. Engr., Czech Tech. Univ. Prague, Czech Rep. henrion@fel.cvut.cz

Abstract

Many problems of systems control theory boil down to solving polynomial equations, polynomial inequalities or polyomial differential equations. Recent advances in convex optimization and real algebraic geometry can be combined to generate approximate solutions in floating point arithmetic.

In the first part of the course we describe semidefinite programming (SDP) as an extension of linear programming (LP) to the cone of positive semidefinite matrices. We investigate the geometry of spectrahedra, convex sets defined by linear matrix inequalities (LMIs) or affine sections of the SDP cone. We also introduce spectrahedral shadows, or lifted LMIs, obtained by projecting affine sections of the SDP cones. Then we review existing numerical algorithms for solving SDP problems.

In the second part of the course we describe several recent applications of SDP. First, we explain how to solve polynomial optimization problems, where a real multivariate polynomial must be optimized over a (possibly nonconvex) basic semialgebraic set. Second, we extend these techniques to ordinary differential equations (ODEs) with polynomial dynamics, and the problem of trajectory optimization (analysis of stability or performance of solutions of ODEs). Third, we conclude this part with applications to optimal control (design of a trajectory optimal w.r.t. a given functional).

For some of these decision and optimization problems, it is hoped that the numerical solutions computed by SDP can be refined a posteriori and certified rigorously with appropriate techniques.

Disclaimer

These lecture notes were written for a tutorial course given during the conference "Journées Nationales de Calcul Formel" held at Centre International de Rencontres Mathématiques, Luminy, Marseille, France in May 2013. They are aimed at giving an elementary and introductory account to recent applications of semidefinite programming to the numerical solution of decision problems involving polynomials in systems and control theory. The main technical results are gathered in a hopefully concise, notationally simple way, but for the sake of conciseness and readability, they are not proved in the document. The reader interested in mathematical rigorous comprehensive technical proofs is referred to the papers and books listed in the "Notes and references" section of each chapter. Comments, feedback, suggestions for improvement of these lectures notes are much welcome.

Acknowledgement

I am grateful to the organizers of the conference, and especially to Guillaume Chèze and Mohab Safey El Din, for giving me the opportunity to prepare and present this material. Many thanks to Mohamed Rasheed Hilmy Abdalmoaty, Mathieu Claeys, Simone Naldi, Luis Rodrigues and Mohab Safey El Din for their suggestions and remarks on the text.

Contents

Chapter 1

Motivating examples

In this introductory section we describe elementary problems of systems control theory that can be formulated as decision and optimization problems over polynomial equations and differential equations.

1.1 Structured eigenvalue assignment

We consider the problem of designing a pulsed power generator in an electrical network. The engineering specification of the design is that a suitable resonance condition is satisfied by the circuit so that the energy initially stored in a number of stage capacitors is transferred in finite time to a single load capacitor which can then store the total energy and deliver the pulse.

Mathematically speaking, the problem can be formulated as the following structured matrix eigenvalue assignment problem. Let $n \in \mathbb{N}$ and define the matrix

$$B = \begin{pmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & & & \vdots \\ 0 & & \ddots & & 0 \\ \vdots & & & 2 & -1 \\ 0 & \cdots & 0 & -1 & \frac{n+1}{n} \end{pmatrix}.$$

Given positive rational numbers $a_k \in \mathbb{Q}$, $k = 1, \ldots, n$, consider the eigenvalue assignment problem

$$\det(sI_n - B^{-1} \operatorname{diag} x) = s^n + b_1(x)s^{n-1} + \cdots + b_{n-1}(x)s + b_n(x) = \prod_{k=1}^{n}(s + a_k)$$

where $x \in \mathbb{R}^n$ is a vector of unknowns and $\operatorname{diag} x$ stands for the n-by-n matrix with entries of x along the diagonal. In systems and control terminology, this is a structured pole placement problem, and vector x can be interpreted as a parametrization of a linear controller to be designed. By identifying like powers of indeterminate s in the above

relation, it can be formulated as a polynomial system of equations $p_k(x) = 0$, $k = 1, \ldots, n$ where

$$\begin{aligned} p_1(x) &= b_1(x) - a_1 - \cdots - a_n \\ p_2(x) &= b_2(x) - a_1a_2 - a_1a_3 - \cdots - a_{n-1}a_n \\ &\vdots \\ p_n(x) &= b_n(x) - a_1a_2 \cdots a_n. \end{aligned}$$

In the context of electrical generator design, a physically relevant choice of eigenvalues is

$$a_k = \frac{1}{(2k)^2 - 1}, \quad k = 1, \ldots, n.$$

For example, if $n = 2$, we obtain the following system

$$\begin{aligned} \tfrac{3}{4}x_1 + x_2 - \tfrac{2}{5} &= 0 \\ \tfrac{1}{2}x_1x_2 - \tfrac{1}{45} &= 0. \end{aligned}$$

More generally, we obtain a system with n unknowns and n polynomial equations of respective degrees $1, \ldots, n$ which has typically much less than $n!$ real solutions. Geometrically, the feasibility set

$$X = \{x \in \mathbb{R}^n \ : \ p_k(x) = 0, \quad k = 1, \ldots, n\}$$

is a zero-dimensional real algebraic set of small cardinality. When $n = 8$, say, we would like to find a point in X.

1.2 Control law validation

In aerospace engineering, the validation of control laws is a critical step before industrialization. Generally it is carried out by expensive time-simulations. A very simple, but representative example, is the validation of a control law for a one-degree-of-freedom model of a launcher attitude control system in orbital phase. The closed-loop system must follow a given piecewise linear angular velocity profile. It is modeled as a double integrator

$$I\ddot{\theta}(t) = u(t)$$

where I is a given constant inertia, $\theta(t)$ is the angle and $u(t)$ is the torque control. We denote

$$x(t) = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \end{bmatrix}$$

and we assume that both angle $x_1(t)$ and angular velocity $x_2(t)$ are measured, and that the torque control is given by

$$u(x(t)) = \mathrm{sat}(F' \mathrm{dz}(x_r(t) - x(t)))$$

where $x_r(t)$ is the given reference signal, $F \in \mathbb{R}^2$ is a given state feedback, the prime denotes transposition, sat is a saturation function such that $\mathrm{sat}(y) = y$ if $|y| \leq L$ and $\mathrm{sat}(y) = L\,\mathrm{sign}(y)$ otherwise, dz is a dead-zone function such that $\mathrm{dz}(x) = 0$ if $|x_i| \leq D_i$

for some $i = 1, 2$ and $\mathrm{dz}(x) = 1$ otherwise. Thresholds $L > 0$, $D_1 > 0$ and $D_2 > 0$ are given.

We would like to verify whether the system state $x(t)$ reaches a given subset $X_T = \{x \in \mathbb{R}^2 \ : \ x^T x \leq \varepsilon\}$ of the deadzone region after a fixed time T, and for all possible initial conditions $x(0)$ chosen in a given subset X_0 of the state-space, and for zero reference signals.

1.3 Bolza's optimal control problem

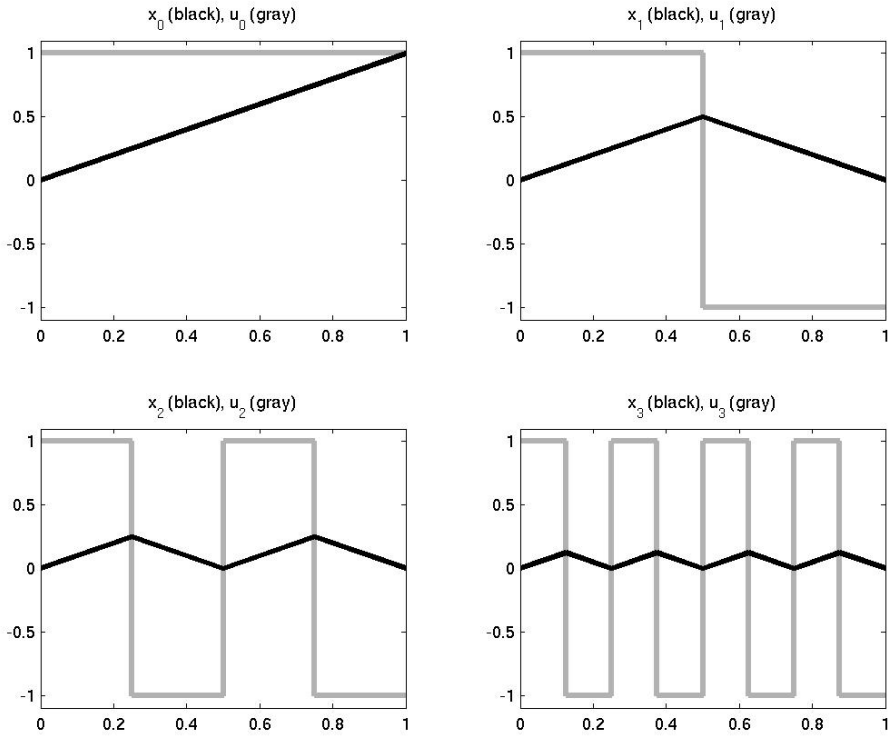

Figure 1.1: Sequences of state trajectories and control inputs for Bolza's example.

Our last example is a classical academic problem of calculus of variations illustrating that an optimal control problem with smooth data (infinitely differentiable Lagrangian and dynamics, no state or input constraints) can have a highly oscillatory optimal solution.

Consider the optimal control problem

$$\begin{array}{rll} p^* = & \inf & \int_0^1 (x^4(t) + (u^2(t) - 1)^2)\, dt \\ & \text{s.t.} & \dot{x}(t) = u(t), \quad t \in [0, 1], \\ & & x(0) = 0, \quad x(1) = 0 \end{array}$$

where the infimum is w.r.t. a Lebesgue integrable real-valued control $u \in \mathscr{L}^1([0, 1]; \mathbb{R})$. Intuitively, the state trajectory $x(t)$ should remain zero, and the velocity $\dot{x} = u$ should be

equal to $+1$ or -1, so that the nonnegative Lagrangian $l(t, x(t), u(t)) := x^4(t)+(u^2(t)-1)^2$ remains zero, and hence the objective function is zero, the best we can hope. We can build a sequence of bang-bang controls $u_k(t)$ such that for each $k = 0, 1, 2, \ldots$ the corresponding state trajectory $x_k(t)$ has a sawtooth shape, see Figure 1.1. With such a sequence the objective function tends to $\lim_{k\to\infty} \int_0^1 l(t, x_k(t), u_k(t))dt = \int_0^1 x_k^4(t)dt = 0$ and hence $p^* = 0$. This infimum is however not attained with a control law $u(t)$ belonging to the space of Lebesgue integrable functions.

We would like to develop a numerical method that can deal with such oscillation phenomena and would allow us to construct explicitly an optimal control law.

1.4 Course outline

The objective of this document is to describe a systematic approach to the numerical solution of these nonlinear nonconvex decision problems. Our strategy will be as follows:

1. the problem is relaxed and linearized to an LP on measures, interpreted as the dual to an LP on continuous functions;
2. since the decision problems have polynomial data, the measure LP is formulated as a moment LP;
3. a hierarchy of finite-dimensional LMI relaxations is used to solve the moment LP numerically, with guarantees of asymptotic, and sometimes finite convergence.

Since we do not assume that the reader is familiar with SDP and the geometry of LMIs, the document starts with an introductory Chapter 2 on finite-dimensional conic programming. In Chapter 3, our approach is applied to nonconvex finite-dimensional polynomial optimization. Finally, we conclude with Chapter 4 on nonconvex infinite-dimensional optimization on solutions of polynomial differential equations, and a last Chapter 5 on extensions to polynomial optimal control.

Chapter 2

Conic optimization

2.1 Convex cones

In this section we describe linear programming over convex cones in finite dimensional Euclidean spaces.

Definition 2.1 (Convex set) *A set K is convex if $x_1, x_2 \in K$ implies $\lambda x_1 + (1-\lambda)x_2 \in K$ for all $\lambda \in [0,1]$.*

Geometrically speaking, a set is convex whenever the line segment linking two of its points belongs to the set.

Definition 2.2 (Convex hull) *Given a set K, its convex hull, denoted by* conv K*, is the smallest closed convex set which contains K.*

The convex hull can be expressed as

$$\operatorname{conv} K := \left\{ \sum_{k=1}^{N} \lambda_k x_k \ : \ N \in \mathbb{N},\ x_k \in K,\ \lambda_k \geq 0,\ \sum_{k=1}^{N} \lambda_k = 1 \right\}.$$

The convex hull of finitely many points $\operatorname{conv}\{x_1, \ldots, x_N\}$ is a polytope with vertices at these points. A theorem by Carathéodory states that given a set $K \subset \mathbb{R}^n$, every point of conv K can be expressed as $\sum_{k=1}^{n+1} \lambda_k x_k$ for some choice of $x_k \in K$, $\lambda_k \geq 0$, $\sum_{k=1}^{n+1} \lambda_k = 1$.

Definition 2.3 (Cone) *A set K is a cone if $\lambda \geq 0, x \in K$ implies $\lambda x \in K$.*

It follows that a convex cone is a set which is invariant under addition and multiplication by non-negative scalars.

Let us denote the scalar product of two vectors x, y of $\mathbb{R}^n$ as follows:

$$\langle x, y \rangle := x'y = \sum_{k=1}^{n} x_k y_k$$

where the prime, applied to a vector or a matrix, denotes the transpose. More generally, we use the prime to denote the dual vector space:

Definition 2.4 (Dual space) *The dual of a vector space V is the space V' of all linear functionals on V.*

When applied to a cone, the prime denotes the dual cone:

Definition 2.5 (Dual cone) *The dual of a cone K is the cone*

$$K' := \{y \in \mathbb{R}^n \,:\, \langle x, y\rangle \geq 0,\ \forall x \in K\}.$$

Geometrically, the dual cone of K is the set of all nonnegative linear functions on K. Notice that the dual K' is always a closed convex cone and that K'' is the closure of the conic hull, i.e. the smallest convex cone that contains K. In particular, if K is a closed convex cone, then $K'' = K$. A cone K such that $K' = K$ is called self-dual.

A cone K is pointed if $K \cap (-K) = \{0\}$ and solid if the interior of K is not empty. A cone which is convex, closed, pointed and solid is called a proper cone. The dual cone of a proper cone is also a proper cone. A proper cone K induces a partial order (a binary relation that is reflexive, antisymmetric and transitive) on the vector space: $x_1 \geq x_2$ if and only if $x_1 - x_2 \in K$.

Definition 2.6 (Linear cone) *The linear cone, or positive orthant, is the set*

$$\{x \in \mathbb{R}^n \,:\, x_k \geq 0,\ k = 1, \ldots, n\}.$$

Definition 2.7 (Quadratic cone) *The quadratic cone, or Lorentz cone, is the set*

$$\{x \in \mathbb{R}^n \,:\, x_1 \geq \sqrt{x_2^2 + \cdots + x_n^2}\}.$$

Let $\mathbb{S}^n$ denote the Euclidean space of n-by-n symmetric matrices of $\mathbb{R}^{n\times n}$, equipped with the inner product

$$\langle X, Y\rangle := \operatorname{trace} X'Y = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}y_{ij}$$

defined for two matrices X, Y with respective entries x_{ij}, y_{ij}, $i, j = 1, \ldots, n$.

Definition 2.8 (Gram matrix) *Given a real quadratic form $f : \mathbb{R}^n \to \mathbb{R}$, the (unique) matrix $X \in \mathbb{S}^n$ such that $f(y) = y'Xy$ is called the Gram matrix of f.*

Definition 2.9 (Positive semidefinite matrix) *A matrix is positive semidefinite when it is the Gram matrix of a nonnegative quadratic form.*

In other words, a matrix $X \in \mathbb{S}^n$ is positive semidefinite, denoted by $X \geq 0$, if and only if $y'Xy \geq 0$, $\forall y \in \mathbb{R}^n$ or equivalently, if and only if the minimum eigenvalue of X is non-negative. This last statement makes sense since symmetric matrices have only real eigenvalues.

Definition 2.10 (Semidefinite cone) *The semidefinite cone, or cone of positive semidefinite matrices, is the set*

$$\{X \in \mathbb{S}^n \,:\, X \geq 0\}.$$

Proposition 2.1 (Self-dual cones) *The linear, quadratic and semidefinite cones are self-dual convex cones.*

Note finally that if $K = \mathbb{R}^n$ is interpreted as a cone, then its dual $K' = \{0\}$ is the zero cone, which contains only the zero vector of $\mathbb{R}^n$.

2.2 Primal and dual conic problems

Conic programming is linear programming in a convex cone K: we want to minimize a linear function over the intersection of K with an affine subspace:

$$\begin{array}{rcll} p^* & = & \inf & c'x \\ & & \text{s.t.} & Ax = b \\ & & & x \in K \end{array} \tag{2.1}$$

where the infimum is w.r.t. a vector $x \in \mathbb{R}^n$ to be found, and the given problem data consist of a matrix $A \in \mathbb{R}^{m \times n}$, a vector $b \in \mathbb{R}^m$ and a vector $c \in \mathbb{R}^n$. Note that the feasibility set $\{x \in \mathbb{R}^n \,:\, Ax = b,\, x \in K\}$ is not necessarily closed, so that in general we speak of an infimum, not of a minimum.

If $K = \mathbb{R}^n$, the whole Euclidean space, or free cone, then problem (2.1) amounts to solving a linear system of equations. If K is the linear cone, then solving problem (2.1) is called linear programming (LP). If K is the quadratic cone, then this is called (convex) quadratic programming (QP). If K is the semidefinite cone, then this is called (linear) **semidefinite programming** (SDP).

In standard mathematical programming terminology, problem (2.1) is called the primal problem, and p^* denotes its infimum. The primal conic problem has a dual conic problem:

$$\begin{array}{rcll} d^* & = & \sup & b'y \\ & & \text{s.t.} & z = c - A'y \\ & & & z \in K'. \end{array} \tag{2.2}$$

Note that from Proposition 2.1, if K is the direct product of linear, quadratic and semidefinite cones, then $K' = K$. If K contains a free cone, then the corresponding components in K' are zero: we can enforce equality constraints on some entries in vector z in dual problem (2.2), and they correspond to unrestricted entries in vector x in primal problem (2.1).

Example 2.1 *If K is the direct product of a 2-dimensional free cone with a 2-dimensional LP cone and a 2-dimensional SDP cone, then in primal problem (2.1) the constraint*

$x \in K \subset \mathbb{R}^7$ can be expressed entrywise as:

$$\begin{array}{l} x_1 \text{ free}\,, x_2 \text{ free}, \\ x_3 \geq 0,\ x_4 \geq 0, \\ \begin{pmatrix} x_5 & x_6 \\ x_6 & x_7 \end{pmatrix} \geq 0 \end{array}$$

and in dual problem (2.2) the constraint $z \in K' \subset \mathbb{R}^7$ can be expressed entrywise as:

$$\begin{array}{l} z_1 = 0,\ z_2 = 0, \\ z_3 \geq 0,\ z_4 \geq 0, \\ \begin{pmatrix} z_5 & z_6 \\ z_6 & z_7 \end{pmatrix} \geq 0. \end{array}$$

If K consists of only one semidefinite cone, primal problem (2.1) can be written as follows:

$$\begin{array}{rcll} p^* & = & \inf & \langle C, X \rangle \\ & & \text{s.t.} & \mathcal{A}X = b \\ & & & X \geq 0 \end{array} \tag{2.3}$$

where the given problem data consist now of a linear operator $\mathcal{A} : \mathbb{S}^n \to \mathbb{R}^m$, a vector $b \in \mathbb{R}^m$ and a matrix $C \in \mathbb{S}^n$. The action of operator $\mathcal{A}$ is described entrywise as $\langle A_k, X \rangle = b_k$, for given matrices $A_k \in \mathbb{S}^n$, $k = 1, \ldots, m$. The adjoint or dual operator $\mathcal{A}' : (\mathbb{R}^m)' = \mathbb{R}^m \to (\mathbb{S}^n)' = \mathbb{S}^n$ is the unique linear map such that $\langle \mathcal{A}'y, X \rangle = \langle y, \mathcal{A}X \rangle$ for all $X \in \mathbb{S}^n$ and $y \in \mathbb{R}^m$. More concretely, $\mathcal{A}'y = \sum_{k=1}^m A_k y_k$.

Primal SDP problem (2.3) has a dual SDP problem:

$$\begin{array}{rcll} d^* & = & \sup & \langle b, y \rangle \\ & & \text{s.t.} & C - \mathcal{A}'y \geq 0 \end{array} \tag{2.4}$$

where the supremum is w.r.t. a vector $y \in \mathbb{R}^m$.

Example 2.2 *The primal SDP problem*

$$\begin{array}{rcll} p^* & = & \inf & x_{11} + x_{22} + x_{33} \\ & & \text{s.t.} & -2x_{21} = 1 \\ & & & -2x_{31} = 1 \\ & & & -2x_{32} = 1 \\ & & & \begin{pmatrix} x_{11} & x_{21} & x_{31} \\ x_{21} & x_{22} & x_{32} \\ x_{31} & x_{32} & x_{33} \end{pmatrix} \geq 0 \end{array}$$

has a dual SDP problem

$$\begin{array}{rcll} d^* & = & \sup & y_1 + y_2 + y_3 \\ & & \text{s.t.} & \begin{pmatrix} 1 & y_1 & y_2 \\ y_1 & 1 & y_3 \\ y_2 & y_3 & 1 \end{pmatrix} \geq 0. \end{array}$$

Both problems share the data

$$A_1 = -\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},\ A_2 = -\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix},\ A_3 = -\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

and

$$b = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},\ C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

on the 3-dimensional SDP cone $K = K'$.

2.3 Spectrahedra and LMIs

The convex feasibility sets of problems (2.1) and (2.2) are intersections of a convex cone with an affine subspace. We would like to understand the geometry of these sets. In particular, we would like to know whether a given convex set can be modeled like this.

The most general case relevant for our purposes is when K is the direct product of semidefinite cones. Indeed, note first that every linear cone is the direct product of one-dimensional quadratic cones, or equivalently, of one-dimensional semidefinite cones. Second, note that a quadratic cone is a particular affine section of the semidefinite cone:

$$\{x \in \mathbb{R}^n \,:\, x_1 \geq \sqrt{x_2^2 + \cdots + x_n^2}\} \;=\; \{x \in \mathbb{R}^n \,:\, \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_2 & x_1 & & 0 \\ \vdots & & \ddots & \vdots \\ x_n & 0 & \cdots & x_1 \end{pmatrix} \geq 0\}.$$

It follows that every set that can be represented as an affine section of direct products of the linear and quadratic cone can be represented as an affine section of direct products of the semidefinite cone. Finally, note that a direct product of semidefinite cones can be expressed as an affine section of a single semidefinite cone, e.g.

$$\{x \in \mathbb{R}^4 \,:\, x_1 \geq 0,\ \begin{pmatrix} x_2 & x_3 \\ x_3 & x_4 \end{pmatrix} \geq 0\} \;=\; \{x \in \mathbb{R}^4 \,:\, \begin{pmatrix} x_1 & 0 & 0 \\ 0 & x_2 & x_3 \\ 0 & x_3 & x_4 \end{pmatrix} \geq 0\}.$$

For this reason, in most of the remainder of this document, we consider a single semidefinite cone constraint.

Definition 2.11 (LMI) *A linear matrix inequality (LMI) is a constraint*

$$F_0 + \sum_{k=1}^{n} x_k F_k \geq 0$$

on a vector $x \in \mathbb{R}^n$, where matrices $F_k \in \mathbb{S}^m$, $k = 0, 1, \ldots, n$ are given.

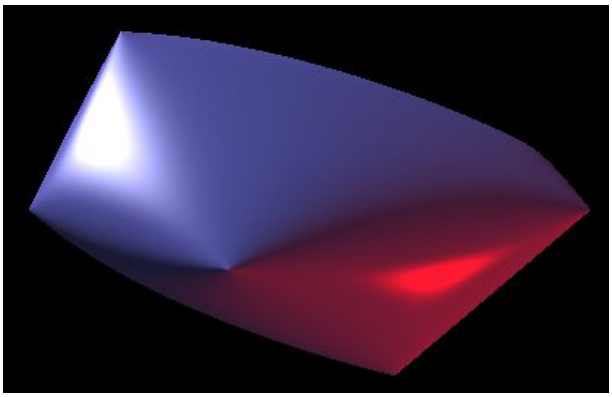

Figure 2.1: A spectrahedron.

Note that an LMI constraint is generally nonlinear, but it is always convex. To prove convexity, rewrite the LMI constraint as

$$y'\left(F_0 + \sum_{k=1}^{n} x_k F_k\right) y = (y'F_0 y) + \sum_{k=1}^{n} (y'F_k y) x_k \geq 0$$

which models infinitely many linear constraints on $x \in \mathbb{R}^n$, parametrized by $y \in \mathbb{R}^m$.

Definition 2.12 (Spectrahedron, or LMI set) *A spectrahedron is a set described by an LMI:*

$$\{x \in \mathbb{R}^n \ : \ F_0 + \sum_{k=1}^{n} x_k F_k \geq 0\}$$

where matrices $F_k \in \mathbb{S}^m$, $k = 0, 1, \ldots, n$ are given.

In other words, spectrahedra are affine sections of the semidefinite cone, or equivalently, LMI sets. Note that in the case where matrices F_k, $k = 0, 1, \ldots, n$ all commute (e.g. if they are all diagonal), the LMI reduces to m affine inequalities, and the spectrahedron reduces to a polyhedron.

On Figure 2.1 we represent a spectrahedron in the case $n = 3$ and $m = 5$. We observe that its boundary is almost everywhere smooth and curved outwards (by convexity), but it also includes vertices and edges.

Let $\mathbb{R}[x]$ denote the ring of polynomials of the indeterminate $x \in \mathbb{R}^n$ with real coefficients. Given a polynomial $f \in \mathbb{R}[x]$, we define its set of zeros, or level set, as $\{x \in \mathbb{R}^n : f(x) = 0\}$. We define its open superlevel set as $\{x \in \mathbb{R}^n \ : \ f(x) > 0\}$, and its closed superlevel set as $\{x \in \mathbb{R}^n \ : \ f(x) \geq 0\}$. Note that these sets are defined in $\mathbb{R}^n$, not in $\mathbb{C}^n$, since in this document we are mainly concerned with optimization.

Definition 2.13 (Algebraic set) *An algebraic set is an intersection of finitely many polynomial level sets.*

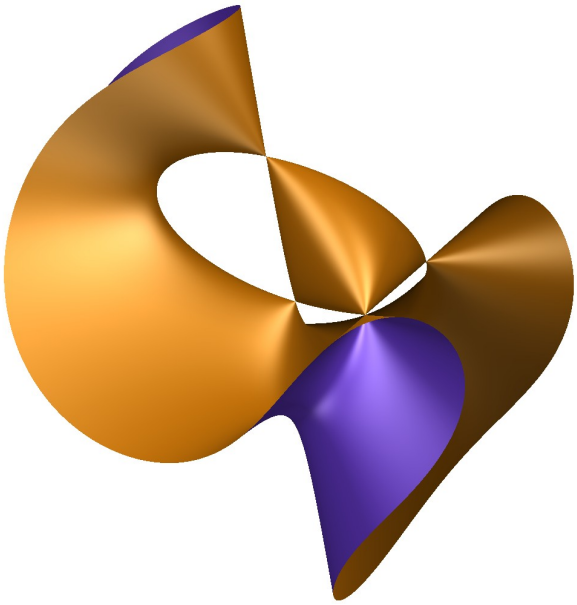

Figure 2.2: The Cayley cubic surface and its spectrahedron.

Definition 2.14 (Semialgebraic set) *A semialgebraic set is a union of finitely many intersections of finitely many open polynomial superlevel sets.*

Definition 2.15 (Closed basic semialgebraic set) *A closed basic semialgebraic set is an intersection of finitely many closed polynomial superlevel sets.*

Now, let us denote by

$$F(x) := F_0 + \sum_{k=1}^{n} x_k F_k$$

the affine symmetric matrix describing a spectrahedron, and build its characteristic polynomial

$$t \mapsto \det\left(tI_m + F(x)\right) = \sum_{k=0}^{m} f_{m-k}(x) t^k$$

which is monic, i.e. $f_0(x) = 1$. Coefficients $f_k \in \mathbb{R}[x]$, $k = 1, \ldots, m$ are multivariate polynomials called the defining polynomials of the spectrahedron. They are elementary symmetric functions of the eigenvalues of $F(x)$.

Proposition 2.2 (Spectrahedra are closed basic semialgebraic sets) *A spectrahedron can be expressed as follows:*

$$\{x \in \mathbb{R}^n \,:\, F_0 + \sum_{k=1}^{n} x_k F_k \geq 0\} \;=\; \{x \in \mathbb{R}^n \,:\, f_k(x) \geq 0,\ k = 1, \ldots, m\}.$$

Example 2.3 (The pillow) *As an elementary example, consider the pillow spectrahedron*

$$X := \{x \in \mathbb{R}^3 \ : \ F(x) := \begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & 1 & x_3 \\ x_2 & x_3 & 1 \end{pmatrix} \geq 0\}$$

and its defining polynomials

$$\begin{array}{rcl} f_1(x) & = & \text{trace}\, F(x) = 3, \\ f_2(x) & = & 3 - x_1^2 - x_2^2 - x_3^2, \\ f_3(x) & = & \det F(x) = 1 + 2x_1x_2x_3 - x_1^2 - x_2^2 - x_3^2, \end{array}$$

On Figure 2.2 we represent the Cayley cubic surface

$$\{x \in \mathbb{R}^3 \ : \ f_3(x) = 0\}$$

which is the algebraic closure of the boundary of the pillow spectrahedron (the inner convex region)

$$X = \{x \in \mathbb{R}^3 \ : \ f_2(x) \geq 0, \ f_3(x) \geq 0\}.$$

In other words, the polynomial which vanishes along the boundary of X also vanishes outside of X, along the Cayley cubic surface.

2.4 Spectrahedral shadows and lifted LMIs

We have seen that a spectrahedron is a closed basic semialgebraic set. Moreover, it is a convex set. All spectrahedra are convex closed basic semialgebraic, so one may then wonder conversely whether all convex closed basic semialgebraic sets are spectrahedra. The answer is negative, even though we do not explain why in this document.

Proposition 2.3 (The TV screen is not a spectrahedron) *The planar convex closed basic semialgebraic set*

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

is not a spectrahedron.

Consequently, in order to represent convex closed basic semialgebraic sets, we have to go beyond affine sections of the semidefinite cone. This motivates the following definitions.

Definition 2.16 (Lifted LMI, liftings) *A lifted LMI is a constraint*

$$F_0 + \sum_{k=1}^{n} x_k F_k + \sum_{l=1}^{p} u_l G_l \geq 0$$

on a vector $x \in \mathbb{R}^n$, which implies additional variables $u \in \mathbb{R}^p$ called liftings, and where matrices $F_k \in \mathbb{S}^m$, $k = 0, 1, \ldots, n$ and $G_l \in \mathbb{S}^m$, $l = 1, \ldots, p$ are given.

Definition 2.17 (Spectrahedral shadow, or lifted LMI set) *A spectrahedral shadow is the affine projection of a spectrahedron:*

$$\{x \in \mathbb{R}^n \ : \ F_0 + \sum_{k=1}^{n} x_k F_k + \sum_{l=1}^{p} u_l G_l \geq 0, \ u \in \mathbb{R}^p\}$$

where matrices $F_k \in \mathbb{S}^m$, $k = 0, 1, \ldots, n$ and $G_l \in \mathbb{S}^m$, $l = 1, \ldots, p$ are given.

Spectrahedral shadows are also called semidefinite representable sets.

Example 2.4 (The TV-screen is a spectrahedral shadow) *The planar convex closed basic semialgebraic set*

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

can be expressed as the spectrahedral shadow

$$\left\{x \in \mathbb{R}^2 \ : \ \begin{pmatrix} 1-u_1 & u_2 \\ u_2 & 1+u_1 \end{pmatrix} \geq 0, \quad \begin{pmatrix} 1 & x_1 \\ x_1 & u_1 \end{pmatrix} \geq 0, \quad \begin{pmatrix} 1 & x_2 \\ x_2 & u_2 \end{pmatrix} \geq 0, \quad u \in \mathbb{R}^2 \right\}.$$

Proposition 2.4 (Planar semialgebraic sets are spectrahedral shadows) *Every planar convex closed semialgebraic set is a spectrahedral shadow.*

Conjecture 2.1 (Are convex semialgebraic sets spectrahedral shadows ?) *Every convex closed semialgebraic set is a spectrahedral shadow.*

2.5 SDP duality

In this section we sketch some algorithms for linear semidefinite programming (SDP). For notational simplicity we consider only the case of a single semidefinite cone, with given data $\mathcal{A} : \mathbb{S}^n \to \mathbb{R}^m$, $b \in \mathbb{R}^m$ and $C \in \mathbb{S}^n$. We want to solve the primal SDP problem (2.3) and its dual SDP problem (2.4) reproduced here for the reader's convenience:

$$\begin{array}{rclll} p^* & = & \inf & \langle C, X \rangle & \\ & & \text{s.t.} & \mathcal{A}X = b & \\ & & & X \geq 0 & \end{array} \qquad \begin{array}{rclll} d^* & = & \sup & \langle b, y \rangle & \\ & & \text{s.t.} & Z = C - \mathcal{A}'y & \\ & & & Z \geq 0. & \end{array}$$

Define the feasibility sets

$$P := \{X \in \mathbb{S}^n \ : \ \mathcal{A}X = b, \ X \geq 0\}, \quad D := \{y \in \mathbb{R}^m \ : \ C - \mathcal{A}'y \geq 0\}.$$

Most of the algorithms for solving SDP problems make use of the following elementary duality properties.

Proposition 2.5 (Weak duality) *If P and D are nonempty, it holds $p^* \geq d^*$.*

Indeed, if P and D are nonempty, there exist $X \in P$ and $y \in D$. Letting $Z := C - \mathcal{A}'y$ it holds

$$p^* - d^* = \langle C, X\rangle - \langle b, y\rangle = \langle X, Z\rangle \geq 0$$

since $X \geq 0$ and $Z \geq 0$.

Proposition 2.6 (Strong duality) *If P has nonempty interior and D is nonempty, then the supremum d^* is attained and $p^* = d^*$. Similarly, if D has nonempty interior and P is nonempty, then the infimum p^* is attained and $p^* = d^*$.*

Note in passing that whenever $X \geq 0$, $Z \geq 0$, the scalar condition $p^* - d^* = \langle X, Z\rangle = 0$ is equivalent to the nonsymmetric matrix condition $XZ = 0$ or to the symmetric matrix condition $XZ + ZX = 0$.

2.6 Numerical SDP solvers

Here we briefly describe numerical methods, implemented in floating-point arithmetic, to solve SDP problems. The most successful algorithms are primal-dual interior-point methods.

A triple (X, y, Z) solves the primal-dual SDP problems (2.3)-(2.4) if and only if

$$\begin{array}{cc} \mathcal{A}X = b,\ X \geq 0 & \text{(primal feasibility)} \\ \mathcal{A}'y + Z = C,\ Z \geq 0 & \text{(dual feasibility)} \\ XZ = 0 & \text{(complementarity).} \end{array}$$

The key idea behind primal-dual interior-point methods is then to consider the nonlinear system of equations

$$\begin{array}{rcl} \mathcal{A}X & = & b \\ \mathcal{A}'y + Z & = & C \\ XZ & = & \mu I_n \end{array} \tag{2.5}$$

parametrized in the scalar $\mu > 0$. These are necessary and sufficiently optimality conditions for the strictly convex problem

$$\begin{array}{ll} \inf & \langle C, X\rangle + \mu f(X) \\ \text{s.t.} & \mathcal{A}X = b \end{array}$$

where $f(X) := -\log\det X$ is a barrier function for the semidefinite cone (strictly convex and finite in its interior, and infinite elsewhere). For this reason, scalar μ is called the barrier parameter.

If P and D have nonempty interior, it can be shown that for a given $\mu > 0$, system (2.5) has a unique solution such that $X > 0$ and $Z > 0$, and hence the set $\{(X(\mu), y(\mu), Z(\mu)) : \mu > 0\}$ defines a smooth curve parametrized by μ, called the central path. The interior-point algorithm consists then in applying Newton's method to minimize a weighted sum of the linear objective function and the barrier function, by following the central path, letting $\mu \to 0$. Given initial feasible solutions, this generates a sequence of feasible solutions such that the duality gap $\langle C, X\rangle - \langle b, y\rangle$ is less that a given threshold $\epsilon > 0$ after $O(\sqrt{n}\log\epsilon^{-1})$ iterations. Each Newton iteration requires:

- $O(n^2m)$ operations to evaluate the barrier function;
- $O(n^3m)$ operations to evaluate and store its gradient;
- $O(n^2m^2)$ operations to evaluate and store its Hessian;
- $O(m^3)$ operations to solve the Newton linear system of equations.

A symmetric matrix of size n has $O(n^2)$ independent entries, so in general we may assume that $m = O(n^2)$ and hence the dominating term in this rough complexity analysis comes from the evaluation and storage of the Hessian of the barrier function. Data sparsity and block structure must be exploited as much as possible in these steps. It follows that a global worst-case asymptotic complexity estimate for solving a dense SDP problem is $O(n^{6.5} \log \epsilon^{-1})$. In practice the observed computational burden is much smaller, but it strongly depends on the specific implementation and on the problem structure.

Newton's method needs an initial feasible point, and if no such point is available, an auxilliary SDP problem must be solved first. An elegant approach to bypass the search of an initial point consists of embedding the primal-dual problem in a larger problem which is its own dual and for which a trivial feasible starting point is known: this is the so-called homogeneous self-dual embedding. A drawback of this approach is that iterates are primal and dual feasible for the original SDP problems only when the barrier parameter vanishes.

The most successful semidefinite programming solvers are implementations of primal-dual interior-point algorithms:

- `SeDuMi`, `SDPT3`, `MOSEK`: homogenous self-dual embedding;
- `CSDP`, `SDPA`: path-following predictor-corrector;
- `DSDP`: path-following with dual-scaling;

but there are also other implementations based on different algorithms:

- `LMILAB`: projective method;
- `PENSDP`: penalty and augmented Lagrangian.

There exist parallel implementations of `CSDP` and `SDPA`.

Most of these solvers are available under `Matlab`, and they are interfaced through the parsers `YALMIP` and `cvx`. Some elementary SDP solver is available under `Scilab` and `Sage`, and `cvxopt` is a `Python` interface with some SDP features. The solver `CSDP` can be embedded in C language, the solver `SDPA` is also available with `Python` interface, and `PENSDP` is available as a standalone solver or called in `Fortran` or C language.

2.7 Rigorous SDP solvers

The numerical methods described in the previous sections are implementable in floating-point arthimetic, but very little is known about backward stability of these algorithms. More annoyingly, it is difficult to estimate or bound the conditioning of a SDP problem, which implies that none of these numerical solvers can provide a priori guarantees about the quality of their output, even for a restricted problem class.

To address this issue, various strategies can be followed:

- multiprecision arithmetic;
- interval arithmetic;
- symbolic computation.

Higher precision or arbitrary precision arithmetic allows to deal with better floating-point approximations of real numbers, at the price of an increased computational burden. Currently, the solver `SDPA` is available in quad-double precision, double-double precision and arbitrary precision arithmetic.

Interval arithmetic can be used to obtain rigorous bounds on the output of numerical SDP solvers. A `Matlab` implementation of a verified SDP solver is `VSDP`. It relies on the `Intlab` toolbox for interval computations.

Symbolic computation can be used to solve SDP problems exactly, by solving (e.g. with Gröbner basis techniques) the quadratic system of equations arising from optimality conditions. Alternatively, feasible points in spectrahedra can be obtained by techniques for finding real solutions of systems of polynomial equations and inequalities.

To justify further the need for these techniques, note first that there are SDP problems with integer data with no solution among the rationals:

Example 2.5 (Irrational optimal solution) *The problem*

$$\begin{array}{ll} \sup & y \\ \text{s.t.} & \begin{pmatrix} 1 & y \\ y & 2 \end{pmatrix} \geq 0 \end{array}$$

has solution $y^* = \sqrt{2}$.

Example 2.6 (Irrational spectrahedron)

$$\{y \in \mathbb{R} : \begin{pmatrix} 1 & y \\ y & 2 \end{pmatrix} \geq 0, \begin{pmatrix} 2y & 2 \\ 2 & y \end{pmatrix} \geq 0\} = \{\sqrt{2}\}.$$

In general, exact solutions of SDP problems must be found in algebraic extensions of the ground field of the input data. Recall that when both primal and dual problems have

nonempty interiors, solutions (X, y) are characterized by the optimality conditions (2.5) with $\mu = 0$, i.e.

$$\begin{array}{rcl} \langle A_k, X\rangle & = & b_k, \quad k = 1, \ldots, m \\ X(C - \sum_{k=1}^m y_k A_k) & = & 0_n. \end{array} \tag{2.6}$$

This is a system of $m + n(n+1)/2$ real linear and quadratic equations in $m + n(n+1)/2$ real variables. If we have a basis for the nullspace of the operator $\mathcal{A}$, we can remove the first m equality constraints and derive a system of $n(n+1)/2$ quadratic equations in $n(n+1)/2$ variables.

Example 2.7 (Irrational optimal solution, again) *Optimality conditions for the problem of Example 2.5 are as follows:*

$$\begin{array}{rcl} -x_{21} & = & 1 \\ x_{11} + y x_{21} & = & 0 \\ x_{21} + y x_{22} & = & 0 \\ y x_{21} + 2 x_{22} & = & 0 \end{array} \iff \begin{array}{rcl} x_{11} & = & \pm\sqrt{2} \\ x_{21} & = & -1 \\ x_{22} & = & \pm\frac{\sqrt{2}}{2} \\ y & = & \pm\sqrt{2} \end{array}$$

from which it follows that the primal-dual optimal solution is

$$X^* = \begin{pmatrix} \sqrt{2} & -1 \\ -1 & \frac{\sqrt{2}}{2} \end{pmatrix} \quad y^* = \sqrt{2}.$$

In the classical Turing machine model of computation, an integer number N is encoded in binary notation, so that its bit size is $\log_2 N + 1$. The following spectrahedron with integer coefficients has points with exponential bit size:

Example 2.8 (Exponential spectrahedron) *Any point in the spectrahedron*

$$\{y \in \mathbb{R}^m \,:\, \begin{pmatrix} 1 & 2 \\ 2 & y_1 \end{pmatrix} \geq 0, \begin{pmatrix} 1 & y_1 \\ y_1 & y_2 \end{pmatrix} \geq 0, \cdots, \begin{pmatrix} 1 & y_{m-1} \\ y_{m-1} & y_m \end{pmatrix} \geq 0\}$$

satisfies $y_m \geq 2^{2^m}$.

Example 2.9 (Algebraic solution) *Consider the problem*

$$\begin{array}{ll} \sup & y_1 + y_2 + y_3 \\ \text{s.t.} & \begin{pmatrix} 1+y_3 & y_1+y_2 & y_2 & y_2+y_3 \\ y_1+y_2 & 1-y_1 & y_2-y_3 & y_2 \\ y_2 & y_2-y_3 & 1+y_2 & y_1+y_3 \\ y_2+y_3 & y_2 & y_1+y_3 & 1-y_3 \end{pmatrix} \geq 0. \end{array}$$

Optimality conditions (2.6) yield 13 equations in 13 unknowns. Using Gröbner basis techniques, it is found that these equations have 26 complex solutions. The optimal first variable y_1^* *is the root of a degree 26 univariate polynomial with integer coefficients. This polynomial factors into a degree 16 term*

$$403538653715069011 y_1^{16} - 248077486494886030 4 y_1^{15} + \cdots + 149571632340416$$

and a degree 10 term

$$2018y_1^{10} - 12156y_1^9 + 17811y_1^8 + \cdots - 163$$

both irreducible in $\mathbb{Q}[y_1]$*. The optimal solution* y_1^* *is therefore an algebraic number of degree 16 over* $\mathbb{Q}$*, and it can be checked that it is also the case for the other 12 optimal coordinates* $y_2^*, y_3^*, x_{11}^*, x_{21}^*, \ldots, x_{44}^*$*.*

The above examples indicate that it can be quite costly to solve an SDP problem exactly. The algebraic degree of an SDP problem is the degree of the algebraic extension of the problem data coefficient field over which the solutions should be found. Even for small n and m, this number can be very large.

2.8 Notes and references

References on convex analysis are [51] and [22]. See [6] for an elementary introduction to convex optimization, and [3] for a more advanced treatment aimed at applied mathematicians and engineers. Systems control applications of linear matrix inequalities are described in [5]. Good historical surveys on SDP are [58] and [57]. Classifications of sets and functions that can be represented by affine sections and projections of LP, QP and SDP cones can be found in [38], [3] and [35]. Elementary concepts of algebraic geometry (algebraic sets, semialgebraic sets) are surveyed in [8], and connections between SDP, convex geometry and algebraic geometry are explored in [4]. Proposition 2.2 is proved in [49, Theorem 20]. Example 2.3 comes from the SDP relaxation of a 3-dimensional MAXCUT problem, a classical problem of combinatorial optimization, see [33] and also [40, Example 2] for the link with the Cayley cubic. The example of Proposition 2.3 was studied in [16]. The proof of Proposition 2.4 can be found in [56]. Conjecture 2.1, a follow-up of a question posed in [35, Section 4.3.1], can be found in [17]. A basic account of semidefinite programming duality (Propositions 2.6 and 2.5), as well as Examples 2.6 and 2.8 can be found in [33, Section 2]. Techniques of real algebraic geometry for finding rational points in convex semialgebraic sets are described in [55]. Example 2.9 is taken from [40, Example 4], which describes an approach to quantifying the complexity of solving exactly an SDP problem. Hyperlinks to SDP solvers can be found easily, and the online documentation of the interface `YALMIP` contains many pointers to webpages and software packages.

Chapter 3

Finite-dimensional polynomial optimization

3.1 Measures and moments

Let X be a compact subset of the Euclidean space $\mathbb{R}^n$. Let $\mathscr{B}(X)$ denotes the Borel σ-algebra, defined as the smallest collection of subsets of X which contains all open sets.

Definition 3.1 (Signed measure) *A signed measure is a function $\mu : \mathscr{B}(X) \to \mathbb{R} \cup \{\infty\}$ such that $\mu(\emptyset) = 0$ and $\mu(\cup_{k\in\mathbb{N}} X_k) = \sum_{k\in\mathbb{N}} \mu(X_k)$ for any pairwise disjoint $X_k \in \mathscr{B}(X)$.*

Definition 3.2 (Positive measure) *A positive measure is a signed measure which takes only nonnegative values.*

Positive measures on the Borel σ-algebra are often called Borel measures, and positive measures which take finite values on compact sets are often called Radon measures.

Definition 3.3 (Support) *Given a measure μ its support* $\operatorname{spt}\mu$ *is the closed set of all points x such that $\mu(A) \neq 0$ for every neighborhood A of x. We say that μ is supported on a set A whenever* $\operatorname{spt}\mu \subset A$.

Definition 3.4 (Probability measure) *A probability measure μ on X is a positive measure such that $\mu(X) = 1$.*

Let us denote by $\mathscr{M}_+(X)$ the cone of positive measures supported on X, and by $\mathscr{P}(X)$ the set of probability measures supported on X. Geometrically, $\mathscr{P}(X)$ is an affine section of $\mathscr{M}_+(X)$.

Example 3.1 (Lebesgue measure) *The Lebesgue measure on $\mathbb{R}^n$, also called uniform measure, denoted λ, is a positive measure returning the volume of a set A. For instance, when $n = 1$ and $a \leq b$, $\lambda([a, b]) = b - a$.*

Example 3.2 (Dirac measure) *The Dirac measure at $x = \xi$, denoted $\_\xi(dx)$ or $\delta_{x=\xi}$, is a probability measure such that $\delta_\xi(A) = 1$ if $\xi \in A$, and $\delta_\xi(A) = 0$ if $\xi \notin A$.*

For a given compact set $X \subset \mathbb{R}^n$, let $\mathscr{M}(X)$ denote the Banach space of signed measures supported on X, so that a measure $\mu \in \mathscr{M}(X)$ can be interpreted as a function that takes any subset of X and returns a real number. Alternatively, elements of $\mathscr{M}(X)$ can be interpreted as linear functionals acting on the Banach space of continuous functions $\mathscr{C}(X)$, that is, as elements of the dual space $\mathscr{C}(X)'$, see Definition 2.4. The action of a measure $\mu \in \mathscr{M}(X)$ on a test function $v \in \mathscr{C}(X)$ can be modeled with the duality pairing

$$\langle v,\, \mu \rangle \;:=\; \int_X v(x)\, d\mu(x).$$

Let us denote by $\mathscr{C}_+(X)$ the cone of positive continuous functions on X, whose dual can be identified to the cone of positive measures on X, i.e. $\mathscr{M}_+(X) = \mathscr{C}_+(X)'$.

Definition 3.5 (Monomial) *Given a real vector $x \in \mathbb{R}^n$ and an integer vector $\alpha \in \mathbb{N}^n$, a monomial is defined as*

$$x^\alpha := \prod_{k=1}^n x_k^{\alpha_k}.$$

The degree of a monomial with exponent $\alpha \in \mathbb{N}^n$ is equal to $|\alpha| := \sum_{k=1}^n \alpha_k$.

Definition 3.6 (Moment) *Given a measure $\mu \in \mathscr{M}(X)$, the real number*

$$y_\alpha := \int_X x^\alpha \mu(dx) \tag{3.1}$$

is called its moment of order $\alpha \in \mathbb{N}^n$.

Example 3.3 *For $x \in \mathbb{R}^2$, second order moments are*

$$y_{20} = \int x_1^2 \mu(dx),\;\; y_{11} = \int x_1 x_2 \mu(dx),\;\; y_{02} = \int x_2^2 \mu(dx).$$

The sequence $(y_\alpha)_{\alpha \in \mathbb{N}^n}$ is called the sequence of moments of the measure μ, and given $d \in \mathbb{N}$, the truncated sequence $(y_\alpha)_{|\alpha| \leq d}$ is the vector of moments of degree d.

Definition 3.7 (Representing measure) *If y is the sequence of moments of a measure μ, i.e. if identity (3.1) holds for all $\alpha \in \mathbb{N}^n$, we say that μ is a representing measure for y.*

A basic problem in the theory of moments concerns the characterization of (infinite or truncated) sequences that are moments of some measure. Practically speaking, instead of manipulating a measure, which is a rather abstract object, we manipulate its moments. Indeed, a measure on a compact set is uniquely determined by the (infinite) sequence of its moments.

3.2 Riesz functional, moment and localizing matrices

Measures on $X \subset \mathbb{R}^n$ are manipulated with their moments, i.e. via their actions on monomials. The choice of monomials $(x^\alpha)_\alpha$ is motivated mainly for notational and simplicity reasons. In particular, the product of two monomials is a monomial, i.e. $x^\alpha x^\beta = x^{\alpha+\beta}$. Any other choice of basis $(b_\alpha(x))_\alpha$ would be appropriate to manipulate measures, as soon as the basis is dense w.r.t. the supremum norm in the space of continuous functions $\mathscr{C}(X)$. Numerically speaking, other bases than monomials may be more appropriate, but we do not elaborate further on this issue in this document.

In order to manipulate functions in $\mathscr{C}(X)$ we use polynomials. A polynomial $p \in \mathbb{R}[x]$ of degree $d \in \mathbb{N}$ is understood as a linear combination of monomials:

$$p(x) := \sum_{|\alpha| \leq d} p_\alpha x^\alpha$$

and $p := (p_\alpha)_{|\alpha| \leq d}$ is the vector of its coefficients in the monomial basis $(x^\alpha)_\alpha$. Note that we use the same notation for a polynomial and for its vector of coefficients when no ambiguity is possible. Otherwise we use the notation $p(x)$ to emphasize that we deal the polynomial as a function, not as a vector.

Example 3.4 *The polynomial*

$$x \in \mathbb{R}^2 \;\mapsto\; p(x) = 1 + 2x_2 + 3x_1^2 + 4x_1x_2$$

has a vector of coefficients $p \in \mathbb{R}^6$ *with entries* $p_{00} = 1$, $p_{10} = 0$, $p_{01} = 2$, $p_{20} = 3$, $p_{11} = 4$, $p_{02} = 0$.

Definition 3.8 (Riesz functional) *Given a sequence* $y = (y_\alpha)_{\alpha \in \mathbb{N}^n}$*, we define the Riesz linear functional* $\ell_y : \mathbb{R}[x] \to \mathbb{R}$ *which acts on polynomials* $p(x) = \sum_\alpha p_\alpha x^\alpha$ *as follows:* $\ell_y(p(x)) = \sum_\alpha p_\alpha y_\alpha$.

We can interpret the Riesz functional as an operator that linearizes polynomials. If sequence y has a representing measure μ, integration of a polynomial p w.r.t. μ is obtained by applying the Riesz functional ℓ_y on p, since

$$\ell_y(p) = \sum_\alpha p_\alpha y_\alpha = \sum_\alpha p_\alpha \int x^\alpha \mu(dx) = \int \sum_\alpha p_\alpha x^\alpha \mu(dx) = \int p(x)\mu(dx).$$

Note that formally the Riesz functional is the linear form $p(x) \mapsto \ell_y(p(x))$ and its existence is independent of the choice of basis to represent polynomial $p(x)$. However, for notational simplicity, we use the monomial basis and hence we represent explicitly the Riesz functional with the inner product of the vector $(p_\alpha)_{|\alpha| \leq d}$ of coefficients of the polynomial with the truncated sequence $(y_\alpha)_{|\alpha| \leq d}$.

Example 3.5 *For the polynomial of Example 3.4, the Riesz functional reads*

$$p(x) = 1 + 2x_2 + 3x_1^2 + 4x_1x_2 \;\mapsto\; \ell_y(p) = y_{00} + 2y_{01} + 3y_{20} + 4y_{11}.$$

If we apply the Riesz functional on the square of a polynomial $p(x)$, then we obtain a form which is quadratic in the coefficients of $p(x)$:

Definition 3.9 (Moment matrix) *The moment matrix of order d is the Gram matrix of the quadratic form $p(x) \mapsto \ell_y(p^2(x))$ where polynomial $p(x)$ has degree d, i.e. the matrix $M_d(y)$ such that $\ell_y(p^2(x)) = p' M_d(y) p$.*

Example 3.6 *If $n = 2$ then*

$$M_0(y) = y_{00}, \quad M_1(y) = \begin{pmatrix} y_{00} & y_{10} & y_{01} \\ y_{10} & y_{20} & y_{11} \\ y_{01} & y_{11} & y_{02} \end{pmatrix}, \quad M_2(y) = \begin{pmatrix} y_{00} & y_{10} & y_{01} & y_{20} & y_{11} & y_{02} \\ y_{10} & y_{20} & y_{11} & y_{30} & y_{21} & y_{12} \\ y_{01} & y_{11} & y_{02} & y_{21} & y_{12} & y_{03} \\ y_{20} & y_{30} & y_{21} & y_{40} & y_{31} & y_{22} \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & y_{13} \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} \end{pmatrix}.$$

Note that $M_d(y) \in \mathbb{S}^{\binom{n+d}{n}}$ where

$$\binom{n+d}{n} = \binom{n+d}{d} = \frac{(n+d)!}{n!\, d!}$$

is the number of monomials of n variables of degree at most d. The rows and columns of the moment matrix are indexed by vectors $\alpha \in \mathbb{N}^n$ and $\beta \in \mathbb{N}^n$. Inspection reveals that indeed the entry (α, β) in the moment matrix is the moment $y_{\alpha+\beta}$. By construction, the moment matrix $M_d(y)$ is symmetric and linear in y.

If we apply the Riesz functional on the product of the square of a polynomial $p(x)$ of degree d with a given polynomial $q(x)$, then we obtain a form which is quadratic in the coefficients of $p(x)$.

Definition 3.10 (Localizing matrix) *Given a polynomial $q(x)$, its localizing matrix of order d is the Gram matrix of the quadratic form $p(x) \mapsto \ell_y(q(x)p^2(x))$ where polynomial $p(x)$ has degree d, i.e. the matrix $M_d(q\, y)$ such that $\ell_y(q(x)p^2(x)) = p' M_d(q\, y) p$.*

Note that we use the notation $M_d(q\, y)$ to emphasize the fact that the localizing matrix is bilinear in q and y. When polynomial $q(x) = \sum_\alpha q_\alpha x^\alpha$ is given, matrix $M_d(q\, y)$ is symmetric and linear in y. The localizing matrix can be interpreted as a linear combination of moment matrices, in the sense that its entry (α, β) is equal to $\sum_\gamma q_\gamma y_{\alpha+\beta+\gamma}$.

Example 3.7 *If $n = 2$ and $q(x) = 1 + 2x_1 + 3x_2$ then*

$$M_1(q\, y) = \begin{pmatrix} y_{00} + 2y_{10} + 3y_{01} & y_{10} + 2y_{20} + 3y_{11} & y_{01} + 2y_{11} + 3y_{02} \\ y_{10} + 2y_{20} + 3y_{11} & y_{20} + 2y_{30} + 3y_{21} & y_{11} + 2y_{21} + 3y_{12} \\ y_{01} + 2y_{11} + 3y_{02} & y_{11} + 2y_{21} + 3y_{12} & y_{02} + 2y_{12} + 3y_{03} \end{pmatrix}.$$

Finally, given an infinite-dimensional sequence y, let us denote the infinite dimensional moment and localized matrices, or linear operators, as follows

$$M(y) := M_\infty(y), \quad M(q\, y) := M_\infty(q\, y).$$

3.3 Linking measures and moments

The matrices just introduced allow to explicitly model the constraint that a sequence y has a representing measure μ on a compact basic semialgebraic set X. Under a mild assumption on the representation of X, it turns out that this constraint is an infinite-dimensional LMI.

Assumption 3.1 (Compactness) *Assume that X is a compact basic semialgebraic set*

$$X := \{x \in \mathbb{R}^n \,:\, p_k(x) \geq 0,\ k = 1, \ldots, n_X\}$$

for given $p_k \in \mathbb{R}[x]$, $k = 1, \ldots, n_X$. Moreover, assume that one of the polynomial inequalities $p_k(x) \geq 0$ is of the form $R - \sum_{i=1}^n x_i^2 \geq 0$ where R is a sufficiently large positive constant.

On the one hand, Assumption 3.1 is a little bit stronger than compactness of X. On the other hand, if we assume only that X is compact, this is without loss of generality that a constraint can be added to the description of X so that Assumption 3.1 is satisfied.

Proposition 3.1 (Putinar's Theorem) *Let set X satisfy Assumption 3.1. Then sequence y has a representing measure in $\mathscr{M}_+(X)$ if and only if $M(y) \geq 0$ and $M(p_k\, y) \geq 0$, $k = 1, \ldots, n_X$.*

Note that if we have an equality constraint $p_k(x) = 0$ instead of an inequality constraint in the definition of X, the corresponding localizing constraint becomes $M(p_k\, y) = 0$, which is a set of linear equations in y.

Since matrices $M(y)$ and $M(p_k y)$ are symmetric and linear in y, sequences with representing measures belong to an infinite-dimensional spectrahedron, following the terminology introduced in Section 2.3. To manipulate these objects, we will consider finite-dimensional truncations.

3.4 Measure LP

Let X be the compact basic semialgebraic set defined above for given polynomials $p_k \in \mathbb{R}[x]$, $k = 1, \ldots, n_X$, and satisfying Assumption 3.1. Let $p_0 \in \mathbb{R}[x]$ be a given a polynomial. Consider the optimization problem consisting of minimizing p_0 over X, namely

$$\begin{array}{rcll} p^* & = & \min & p_0(x) \\ & & \text{s.t.} & p_k(x) \geq 0,\ k = 1, \ldots, n_K. \end{array} \tag{3.2}$$

The above minimum is w.r.t. $x \in \mathbb{R}^n$ and since we assume that X is compact, the minimum is attained at a given point $x^* \in X$.

We do not have any convexity property on p_0 or X, so that problem (3.2) may feature several local minima, and possibly several global minima. In the sequel we describe a

hierarchy of LMI relaxations of increasing size, indexed by a relaxation order, and that generates an asymptotically convergent mononotically nondecreasing sequence of lower bounds on p^*.

The key idea is to notice that nonconvex polynomial optimization problem (3.2) over (finite-dimensional set) $X \subset \mathbb{R}^n$ is equivalent to a linear, hence convex, optimization problem over the (infinite-dimensional set) of probability measures supported on X. More specifically, consider the problem

$$\begin{array}{rcll} p_M^* & = & \inf & \int p_0(x)\mu(dx) \\ & & \text{s.t.} & \mu(X) = 1 \\ & & & \mu \in \mathscr{M}_+(X) \end{array} \tag{3.3}$$

which is linear in the decision variable μ, a probability measure supported on X.

Proposition 3.2 (Measure LP formulation of polynomial optimization) *The infimum in LP problem (3.3) is attained, and $p_M^* = p^*$.*

The proof is immediate: for any feasible $\xi \in X$, it holds $p_0(\xi) = \int p_0(x)\mu(dx)$ for the Dirac measure $\mu = \delta_\xi$, showing $p^* \geq p_M^*$. Conversely, as $p_0(x) \geq p^*$ for all $x \in X$, it holds $\int_X p_0(x)\mu(dx) \geq \int_X p^*\mu(dx) = p^* \int_X \mu(dx) = p^*$ since μ is a probability measure, which shows that $p_M^* \geq p^*$. It follows that $p_M^* = p^*$ and that the infimum in problem (3.3) is attained by a Dirac measure $\mu = \delta_{x^*}$ where x^* is a global optimum of problem (3.2).

3.5 Moment LP

In Section 2 we studied LP problems (2.1) in finite-dimensional cones. In the context of polynomial optimization, we came up with infinite-dimensional LP (3.3) which is a special instance of the measure LP

$$\begin{array}{rcll} p^* & = & \inf & \langle c, \mu \rangle \\ & & \text{s.t.} & \mathcal{A}\mu = b \\ & & & \mu \in \mathscr{M}_+(X) \end{array} \tag{3.4}$$

where the decision variable μ is in the cone of nonnegative measures supported on X, a given compact subset of $\mathbb{R}^n$. Linear operator $\mathcal{A} : \mathscr{M}(X) \to \mathbb{R}^m$ takes a measure and returns an m-dimensional vector of real numbers. Vector $b \in \mathbb{R}^m$ is given. The objective function is the duality pairing between a given continuous function $c \in \mathscr{C}(X)$ and μ. Problem (3.4) has a dual (or more rigorously, a predual) problem in the cone of nonnegative functions, but we will not describe it in this document.

If the linear operator $\mathcal{A}$ is described through given continuous functions $a_j \in \mathscr{C}(X)$, $j = 1, \ldots, m$ we can write the LP problem (3.4) more explicitly as

$$\begin{array}{rcll} p^* & = & \inf & \int_X c(x)\mu(dx) \\ & & \text{s.t.} & \int_X a_j(x)\mu(dx) = b_j, \;\; j = 1, \ldots, m \\ & & & \mu \in \mathscr{M}_+(X). \end{array} \tag{3.5}$$

Now suppose that all the functions are polynomials, i.e. $a_j(x) \in \mathbb{R}[x]$, $j = 1, \ldots, m$, $c(x) \in \mathbb{R}[x]$, so that measure μ can be manipulated via the sequence $y := (y_\alpha)_{\alpha \in \mathbb{N}}$ of its moments (3.1). The measure LP (3.5) becomes a moment LP

$$\begin{array}{rcll} p^* & = & \inf & \sum_\alpha c_\alpha y_\alpha \\ & & \text{s.t.} & \sum_\alpha a_{j\alpha} y_\alpha = b_j, \ \ j = 1, \ldots, m \\ & & & y \text{ has a representing measure } \mu \in \mathscr{M}_+(X) \end{array} \tag{3.6}$$

called a generalized problem of moments.

The idea is then to use the explicit LMI conditions of Section 3.3 to model the constraints that a sequence has a representing measure. If the semialgebraic set

$$X := \{x \in \mathbb{R}^n \ : \ p_k(x) \geq 0, \ k = 1, \ldots, n_X\}$$

satisfies Assumption 3.1, problem (3.6) becomes

$$\begin{array}{rcll} p^* & = & \inf & \sum_\alpha c_\alpha y_\alpha \\ & & \text{s.t.} & \sum_\alpha a_{j\alpha} y_\alpha = b_j, \ \ j = 1, \ldots, m \\ & & & M(y) \geq 0, \ M(p_k\, y) \geq 0, \ k = 1, \ldots, n_X \end{array}$$

where the constraints $\sum_\alpha a_{j\alpha} y_\alpha = b_j$, $j = 1, \ldots, m$ model finitely many linear constraints on infinitely many decision variables. In the sequel, we will consider finite-dimensional truncations of this problem, and generate a hierarchy of LMI relaxations called Lasserre's hierarchy in the context of polynomial optimization.

Moment LP (3.6) has a dual in the cone of positive polynomials, and finite-dimensional truncations of this problem correspond to the search of polynomial sum-of-squares representations, which can be formulated with a hierarchy of dual LMI problems, but we will not elaborate more on this point in this document.

3.6 Lasserre's LMI hierarchy

Now remark that LP problem (3.3) is a special instance of the moment LP problem (3.5) with data $c(x) = p_0(x) = \sum_\alpha p_{0\alpha} x^\alpha$, $a(x) = 1$, $b = 1$, so that, as in Section 3.5, problem (3.2) can be equivalently written as

$$\begin{array}{rcll} p^* & = & \inf & \sum_\alpha p_{0\alpha} y_\alpha \\ & & \text{s.t.} & y_0 = 1 \\ & & & M(y) \geq 0, \ M(p_k\, y) \geq 0, \ k = 1, \ldots, n_X. \end{array}$$

Let us denote by r_k the smallest integer not less than half the degree of polynomial p_k, $k = 0, 1, \ldots, n_X$, and let $r_X := \max\{1, r_1, \ldots, r_{n_X}\}$. For $r \geq r_X$, consider **Lasserre's LMI hierarchy**

$$\begin{array}{rcll} p_r^* & = & \inf & \sum_\alpha p_{0\alpha} y_\alpha \\ & & \text{s.t.} & y_0 = 1 \\ & & & M_r(y) \geq 0, \ M_{r-r_k}(p_k\, y) \geq 0, \ k = 1, \ldots, n_X. \end{array} \tag{3.7}$$

The LMI constraints in this problem are truncated, or relaxed versions of the infinite-dimensional LMI constraints of Proposition 3.1. When the **relaxation order** $r \in \mathbb{N}$ tends to infinity, we obtain the following result.

Proposition 3.3 (Lasserre's LMI hierarchy converges) *It holds* $p_r^* \leq p_{r+1}^* \leq p^*$ *and* $\lim_{r\to\infty} p_r^* = p^*$.

Lasserre's LMI relaxations (3.7) can be solved with semidefinite programming, see Chapter 2, and this provides us with a monotonically nondecreasing sequence of lower bounds on the global minimum of nonconvex polynomial optimization problem (3.2).

Proposition 3.4 (Generic finite convergence) *In the finite-dimensional space of coefficients of polynomials* p_k, $k = 0, 1, \ldots, n_X$ *defining problem (3.2), there is a low-dimensional algebraic set which is such that if we choose an instance of problem (3.2) outside of this set, then Lasserre's LMI relaxations have finite convergence, i.e. there exists a finite* r^* *such that* $p_r^* = p^*$ *for all* $r \geq r^*$.

Equivalently, finite convergence occurs under arbitrary small perturbations of the data of problem (3.2), and problems for which finite convergence does not occur are exceptional and degenerate in some sense.

Example 3.8 *Consider the polynomial optimization problem*

$$\begin{array}{rcll} p^* & = & \min & -x_2 \\ & & \text{s.t.} & 3 - 2x_2 - x_1^2 - x_2^2 \geq 0 \\ & & & -x_1 - x_2 - x_1 x_2 \geq 0 \\ & & & 1 + x_1 x_2 \geq 0 \end{array}$$

where the minimum is w.r.t. $x \in \mathbb{R}^2$. *The first LMI relaxation is*

$$\begin{array}{rcll} p_1^* & = & \min & -y_{01} \\ & & \text{s.t.} & y_{00} = 1 \\ & & & \begin{pmatrix} y_{00} & y_{10} & y_{01} \\ y_{10} & y_{20} & y_{11} \\ y_{01} & y_{11} & y_{02} \end{pmatrix} \geq 0 \\ & & & 3y_{00} - 2y_{01} - y_{20} - y_{02} \geq 0 \\ & & & -y_{10} - y_{01} - y_{11} \geq 0 \\ & & & y_{00} + y_{11} \geq 0 \end{array}$$

and the second LMI relaxation is

$$
\begin{array}{rcll}
p_2^* & = & \min & -y_{01} \\
 & & \text{s.t.} & y_{00} = 1 \\
 & & & \begin{pmatrix} y_{00} & y_{10} & y_{01} & y_{20} & y_{11} & y_{02} \\ y_{10} & y_{20} & y_{11} & y_{30} & y_{21} & y_{12} \\ y_{01} & y_{11} & y_{02} & y_{21} & y_{12} & y_{03} \\ y_{20} & y_{30} & y_{21} & y_{40} & y_{31} & y_{22} \\ y_{11} & y_{21} & y_{12} & y_{31} & y_{22} & y_{13} \\ y_{02} & y_{12} & y_{03} & y_{22} & y_{13} & y_{04} \end{pmatrix} \geq 0 \\
 & & & \begin{pmatrix} 3y_{00} - 2y_{01} - y_{20} - y_{02} & 3y_{10} - 2y_{11} - y_{30} - y_{12} & 3y_{01} - 2y_{02} - y_{21} - y_{03} \\ 3y_{10} - 2y_{11} - y_{30} - y_{12} & 3y_{20} - 2y_{21} - y_{40} - y_{22} & 3y_{11} - 2y_{12} - y_{31} - y_{13} \\ 3y_{01} - 2y_{02} - y_{21} - y_{03} & 3y_{11} - 2y_{12} - y_{31} - y_{13} & 3y_{02} - 2y_{03} - y_{22} - y_{04} \end{pmatrix} \geq 0 \\
 & & & \begin{pmatrix} -y_{10} - y_{01} - y_{11} & -y_{20} - y_{11} - y_{21} & -y_{11} - y_{02} - y_{12} \\ -y_{20} - y_{11} - y_{21} & -y_{30} - y_{21} - y_{31} & -y_{21} - y_{12} - y_{22} \\ -y_{11} - y_{02} - y_{12} & -y_{21} - y_{12} - y_{22} & -y_{12} - y_{03} - y_{13} \end{pmatrix} \geq 0 \\
 & & & \begin{pmatrix} y_{00} + y_{11} & y_{10} + y_{21} & y_{01} + y_{12} \\ y_{10} + y_{21} & y_{20} + y_{31} & y_{11} + y_{22} \\ y_{01} + y_{12} & y_{11} + y_{22} & y_{02} + y_{13} \end{pmatrix} \geq 0.
\end{array}
$$

It can be checked that $p_1^* = -2 \leq p_2^* = p^* = -\frac{1+\sqrt{5}}{2}$. *Note that Assumption 3.1 is satisfied for this example, since the constraint* $3 - 2x_2 - x_1^2 - x_2^2 \geq 0$ *certifies boundedness of the feasibility set.*

3.7 Global optimum recovery

From Proposition 3.4 we know that finite convergence of Lasserre's LMI hierarchy is ensured generically, yet we do not know a priori at which relaxation order it occurs. To certify finite convergence, we can use the following condition.

Proposition 3.5 (Certificate of finite convergence) *Let* y^* *be the solution of LMI problem (3.7) at a given relaxation order* $r \geq r_X$. *If*

$$\operatorname{rank} M_{r-r_X}(y^*) = \operatorname{rank} M_r(y^*)$$

then $p_r^* = p^*$.

If the moment matrix rank conditions of Proposition 3.5 are satisfied, then we can use numerical linear algebra to extract $\operatorname{rank} M_r(y^*)$ global optima for problem (3.2). However, we do not describe the algorithm in this document.

Proposition 3.6 (Rank-one moment matrix) *The condition of Proposition 3.5 is satisfied if*

$$\operatorname{rank} M_r(y^*) = 1.$$

If the rank condition of Proposition 3.6 is satisfied, first order moments readily yield a global optimum: $x^* = (y^*_\alpha)_{|\alpha|=1}$.

Example 3.9 *For the polynomial optimization problem of Example 3.8, we obtain at the second LMI relaxation a rank-one matrix $M_2(y^*) \geq 0$ and the global optimum $x^*_1 = y^*_{10} = \frac{1-\sqrt{5}}{2}$, $x^*_2 = y^*_{01} = \frac{1+\sqrt{5}}{2}$.*

3.8 Complexity estimates

Consider a polynomial optimization problem

$$\begin{array}{rll} p^* = & \min & p_0(x) \\ & \text{s.t.} & p_k(x) \geq 0,\ k = 1, \ldots, n_X \end{array}$$

as in (3.2), with $x \in \mathbb{R}^n$, and its hierarchy of LMI relaxations (3.7).

Let us denote by M the number of variables, i.e. the size of vector y, in the LMI relaxation of order r. It is equal to the number of monomials of n variables of degree $2r$, namely $M = \binom{n+2r}{n}$. If the number of variables n is fixed (e.g. for a given polynomial optimization problem) then M grows in $O(r^n)$, that is polynomially in the relaxation order r. If the relaxation order r is fixed (say to the smallest possible value, the first LMI relaxation in the hierarchy), then M grows in $O(n^r)$, that is polynomially in the number of variables n.

In practice, given the current state-of-the-art in general-purpose SDP solvers and personal computers, we can expect an LMI problem to be solved in a matter of a few minutes provided the problem is reasonably well-conditioned and the number of variables M is less than 5000, say.

3.9 Convex hulls of semialgebraic sets

Let us use the notations defined in Section 3.6, and let X be the compact basic semialgebraic set defined there and satisfying Assumption 3.1. For $r \geq r_X$ consider the spectrahedral shadow

$$X_d := \{(y_\alpha)_{|\alpha|=1} \in \mathbb{R}^n \ :\ y_0 = 1,\ M_r(y) \geq 0,\ M_{r-r_k}(p_k\, y) \geq 0,\ k = 1, \ldots, n_X\}.$$

Proposition 3.7 (Convex outer approximations of semialgebraic sets) *X_r is an outer approximation of X, i.e. $X \subset X_r$. Moreover, $X_{r+1} \subset X_r$, and $X_\infty = \operatorname{conv} X$.*

The result is also true if X is a compact algebraic set, defined by finitely many polynomial equations. In particular, if X is finite-dimensional, i.e. the union of a finite number of points of $\mathbb{R}^n$, then X_∞ is a polytope.

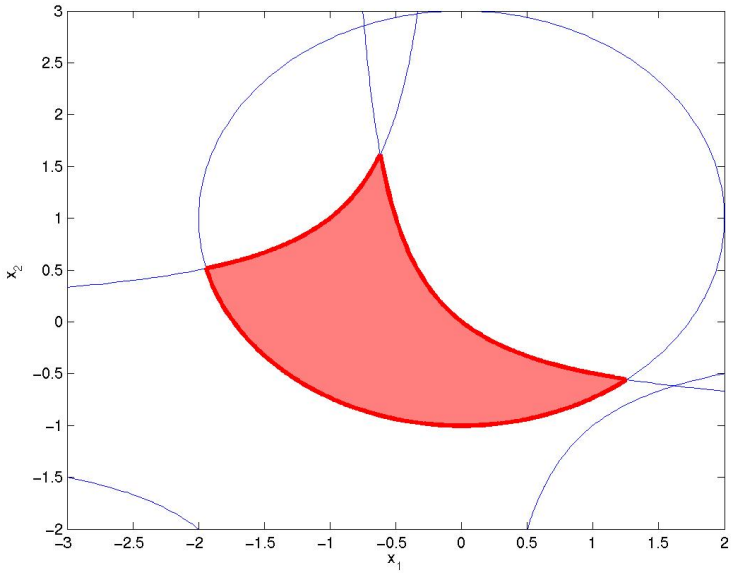

Figure 3.1: Nonconvex semialgebraic set X (in red).

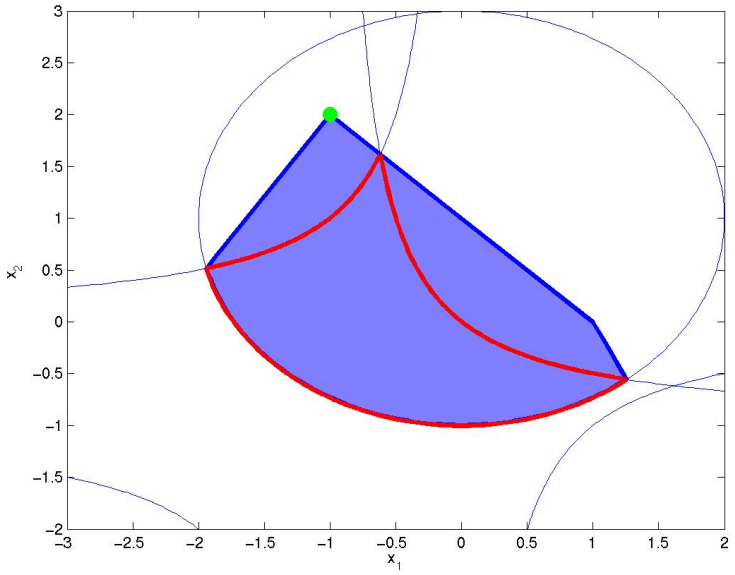

Figure 3.2: First spectrahedral shadow $X_1 \supset X$ (in blue) with boundary of X (in red) and suboptimal point (in green).

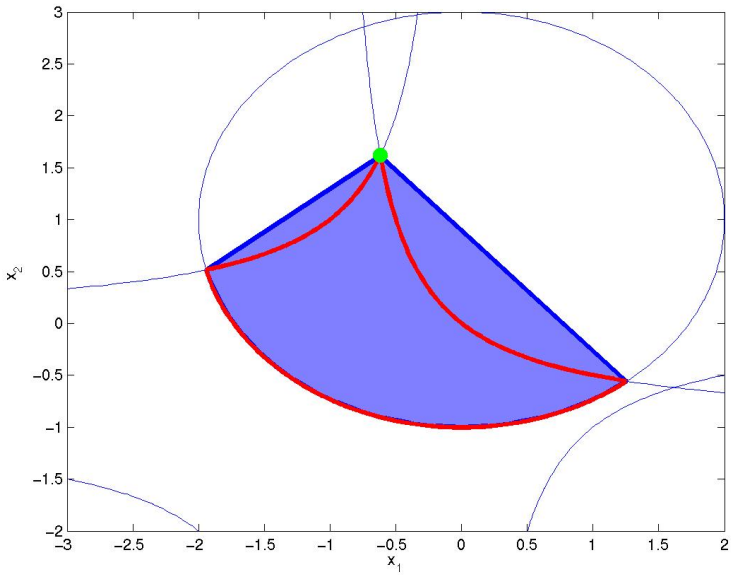

Figure 3.3: Second spectrahedral shadow $X_2 \supset X$ (in blue) with boundary of X (in red) and optimal point (in green).

Example 3.10 *The polynomial optimization problem of Example 3.8 has a compact basic semialgebraic feasible set*

$$X = \{x \in \mathbb{R}^2 \,:\, 3 - 2x_2 + x_1^2 - x_2^2 \geq 0,\ -x_1 - x_2 - x_1x_2 \geq 0,\ 1 + x_1x_2 \geq 0\}$$

represented in red on Figure 3.1. The first spectrahedral shadow $X_1 \supset X$, corresponding to the projection on the plane of first-order moments of the 5-dimensional spectrahedron of the first LMI relaxation, is represented in blue on Figure 3.2. Also represented in green is the point corresponding to the minimization of $-y_{01}$, yielding the lower bound p_1^*. The second spectrahedral shadow $X_2 \supset X$, corresponding to the projection of the plane of first-order moments of the 14-dimensional spectrahedron of the second LMI relaxation, is represented in blue on Figure 3.3. Also represented in green is the point corresponding to the minimization of $-y_{01}$, yielding the lower bound p_2^*. Apparently, $X_2 = \operatorname{conv} X$, so minimizing $-x_2$ on X or $-y_{01}$ on X_2 makes no difference.

3.10 Software interfaces

A `Matlab` interface called `GloptiPoly` has been designed to construct Lasserre's LMI relaxations in a format understandable by the SDP solver `SeDuMi`, but also any other SDP solver interfaced via `YALMIP`. It can be used to construct an LMI relaxation (3.7) of given order corresponding to a polynomial optimization problem (3.2) with given polynomial data entered symbolically. More generally, it can be used to model generalized problems of moments (3.6). A numerical algorithm is implemented in `GloptiPoly` to detect global optimality of an LMI relaxation, using the rank tests of Propositions 3.5 and 3.6. The

algorithm also extracts numerically the global optima from a singular value decomposition of the moment matrix. Another `Matlab` interface called `SOSTOOLS` was developed independently and concurrently. It focuses on the dual polynomial sum-of-squares decompositions mentioned at the end of Section 3.5, but not described in this document. Note however that there is no global optimality detection and global optima extraction algorithm in `SOSTOOLS`. Specialized `moment` and `sos` modules are available in the interface `YALMIP` that implement some of the algorithms of `GloptiPoly` and `SOSTOOLS`. For sparse polynomial optimization problems (with polynomial data featuring a few number of nonzero monomials), a specialized interface called `SparsePOP` is available. It generates reduced-size LMI relaxations by exploiting the problem structure.

Note that these interfaces only generate the LMI relaxations in a format understandable by general-purpose SDP solvers. There is currently no working implementation of a dedicated SDP solver for problems coming from polynomial optimization.

3.11 Back to the motivating example

Let us address the eigenvalue assignment problem of Section 1.1. We formulate it as a nonconvex polynomial optimization problem

$$\begin{array}{ll} \min & p_0(x) \\ \text{s.t.} & p_k(x) = 0,\ k = 1, \ldots, n \end{array}$$

where the objective function is a positive definite convex quadratic form

$$p_0(x) := \sum_{i,j=1}^{n} (x_i - x_j)^2.$$

This choice is motivated by physical reasons, and it corresponds to the search of a solution x with entries x_i as identical as possible.

First, let us generate the system of polynomial equations $p_k(x) = 0$, $k = 1, \ldots, n$ with the following Maple script:

```
with(LinearAlgebra):with(PolynomialTools):
n:=3:B:=Matrix(n):for i from 1 to n-1 do
 B(i,i):=2: B(i,i+1):=-1: B(i+1,i):=-1:
end do: B(n,n):=(n+1)/n;
K:=Matrix(n,Vector(n,symbol=k),shape=diagonal):
q:=product(x-1/((2*j)^2-1),j=1..n):
p:=CoefficientList(collect(charpoly(MatrixInverse(B).F,x)-q,x),x);
```

For $n = 3$ this code generates the following polynomials

$$\begin{array}{rcl} p_1(x) & = & \frac{5}{6}x_1 + \frac{4}{3}x_2 + \frac{3}{2}x_3 - \frac{3}{7} \\ p_2(x) & = & \frac{2}{3}x_1x_2 + x_1x_3 + x_2x_3 - \frac{53}{1575} \\ p_3(x) & = & \frac{1}{2}x_1x_2x_3 - \frac{1}{1575}. \end{array}$$

These polynomials are then converted into `Matlab` format, and we use the following `GloptiPoly` code for inputing problem and solving the smallest possible LMI relaxation, i.e. $r = 2$ in problem (3.7):

```
mpol x 3
X = [5/6*x(1)+4/3*x(2)+3/2*x(3)-3/7
     2/3*x(1)*x(2)+x(1)*x(3)+x(2)*x(3)-53/1575
     1/2*x(1)*x(2)*x(3)-1/1575];
obj = 0;
for i = 1:length(x)
 for j = 1:length(x)
  obj = obj+(x(i)-x(j))^2;
 end
end
P = msdp(min(obj),X==0);
[stat,obj] = msol(P);
double(x)
```

With this code and the SDP solver `SeDuMi`, we obtain the unique solution

$$x \approx (9.3786 \cdot 10^{-2},\ 8.6296 \cdot 10^{-2},\ 1.5690 \cdot 10^{-1})$$

(to 5 significant digits) certified numerically by a rank one moment matrix, see Proposition 3.6, after less than one second of CPU time on a standard desktop computer.

The cases $n = 2, 3, 4, 5$ are solved very easily (in a few seconds) but the solution (obtained with `SeDuMi`) is not very accurate. We have not investigated the possibility of refining the solution with e.g. Newton's method. We have not investigated either the possibility of certifying rigorously the solution using e.g. `VSDP` or multiprecision arithmetic.

The case $n = 6$ is solved in a few minutes, and the case $n = 7$ is significantly harder: it takes a few hours to be solved. Finally, solving the case $n = 8$ takes approximately 15 hours on our computer.

3.12 Notes and references

Accessible introductions to measure theory and relevant notions of functional analysis and probability theory are [50, 23, 34, 53]. Lasserre's hierarchy of LMI relaxations for polynomial optimization were originally proposed in [27, 28] with a proof of convergence (Proposition 3.3) relying on Putinar's Positivstellensatz (Proposition 3.1) described in [46]. The genericity result of Proposition 3.4 is described in [39]. Examples 3.8 and 3.10 are from [18]. The rank condition of Proposition 3.5 relies on flat extension results by Curto and Fialkow, see e.g. [9]. The algorithm implemented in `GloptiPoly` for extracting global optima was described in [19], see also [32] and [30] for more comprensive descriptions. The use of dual polynomial sum-of-squares was proposed in [41], see also [4]. Optimization over polynomial sum-of-squares and more generally squared functional systems and the connection with SDP was studied in [37], mostly in the univariate case. An

excellent survey of this material (on both the primal problems on moments and the dual problems on polynomial sum-of-squares) is [32], and the reader is refered to [30] for a more advanced treatment. The outer approximation result of Proposition 3.7 follows from the convergence proof of Lasserre's relaxation [28] and elementary duality arguments. Finally, the structured eigenvalue assignment problem of Sections 1.1 and 3.11 is comprehensively described in [14].

Chapter 4

Infinite-dimensional polynomial optimization

We extend the approach of the previous chapter to optimization over the infinite-dimensional sets of solutions of ordinary differential equations with polynomial vector fields.

4.1 Occupation measures

Let us consider the nonlinear ordinary differential equation (ODE)

$$\dot{x}(t) = f(t, x(t)) \tag{4.1}$$

for $t \in [0, T]$ with a given terminal time $T > 0$, where $x : [0, T] \to \mathbb{R}^n$ is a time-dependent n-dimensional state vector, and vector field $f : [0, T] \times \mathbb{R}^n \to \mathbb{R}^n$ is a smooth map. Given a set $X \subset \mathbb{R}^n$, we assume that dynamics f and terminal time T are such that there is a solution to the Cauchy problem for ODE (4.1). Since vector field f is smooth, this solution is unique for any given initial condition $x(0) = x_0 \in X$. Any such solution, or trajectory, $x(t)$ is an absolutely continuous function of time with values in X, and to emphasize the dependence of the solution on the initial condition we write $x(t \mid x_0)$.

Now think of initial condition x_0 as a random variable in X, or more abstractly as a probability measure $\xi_0 \in \mathscr{P}(X)$, that is a map from the Borel σ-algebra $\mathscr{B}(X)$ of subsets of X to the interval $[0, 1] \subset \mathbb{R}$ such that $\xi_0(X) = 1$. For example, the expected value of x_0 is the vector $E[x_0] = \int_X x \, \xi_0(dx)$ of first order moments of ξ_0.

Now solve ODE (4.1) for a trajectory, given this random initial condition. At each time t, the state can also be interpreted as a random variable, i.e. a probability measure that we denote by $\xi \in \mathscr{P}(X)$. We say that the measure is transported by the flow of the ODE. We also use the notation $\xi(dx \mid t)$ if we want to emphasize the fact that ξ is a conditional probability measure, or stochastic kernel, i.e. a probability measure acting on subsets of $\mathscr{B}(X)$ for each given, or frozen value of t.

This one-dimensional family, or path of measures, satisfies a partial differential equation (PDE) which turns out to be linear in the space of probability measures. This PDE is

usually called Liouville's equation. Conversely, the nonlinear ODE follows by applying Cauchy's method of characteristics to the linear transport PDE.

Let us now derive the Liouville equation explicitly.

Definition 4.1 (Indicator function) *The indicator function of a set A is the function $x \mapsto I_A(x)$ such that $I_A(x) = 1$ when $x \in A$ and $I_A(x) = 0$ when $x \notin A$.*

Definition 4.2 (Occupation measure) *Given an initial condition x_0, the occupation measure of a trajectory $x(t \mid x_0)$ is defined by*

$$\mu(A \times B \mid x_0) := \int_A I_B(x(t \mid x_0))dt$$

for all $A \in \mathscr{B}([0,T])$ and $B \in \mathscr{B}(X)$.

A geometric interpretation is that μ measures the time spent by the graph of the trajectory $(t, x(t \mid x_0))$ in a given subset $A \times B$ of $[0,T] \times X$. An analytic interpretation is that integration w.r.t μ is equivalent to time-integration along a system trajectory, i.e.

$$\int_0^T v(t, x(t \mid x_0))dt = \int_0^T \int_X v(t,x)\mu(dt, dx \mid x_0)$$

for every test function $v \in \mathscr{C}([0,T] \times X)$.

Example 4.1 (Occupation measure for a scalar linear system) *Consider the one-dimensional ODE $\dot{x}(t) = -x(t)$ with initial condition $x(0) = x_0 \geq 0$, whose solution is $x(t) = x_0 e^{-t}$. Given $a \geq 0$, the occupation measure of the trajectory is such that*

$$\begin{array}{rcll} \mu([0,1] \times [0,a] \mid x_0) & = & 1 & \text{if } x_0 \leq a \\ & = & 1 - \log \frac{x_0}{a} & \text{if } a \leq x_0 \leq ae \\ & = & 0 & \text{if } x_0 > ae \end{array}$$

where $e \approx 2.71828$ is Euler's number.

Now define the linear operator $\mathcal{L} : \mathscr{C}^1([0,T] \times X) \to \mathscr{C}([0,T] \times X)$ by

$$v \mapsto \mathcal{L}v := \frac{\partial v}{\partial t} + \sum_{i=1}^n \frac{\partial v}{\partial x_i} f_i = \frac{\partial v}{\partial t} + (\operatorname{grad} v)' f$$

and its adjoint operator $\mathcal{L}' : \mathscr{C}([0,T] \times X)' \to \mathscr{C}^1([0,T] \times X)'$ by the relation

$$\langle v, \mathcal{L}'\mu \rangle := \langle \mathcal{L}v, \mu \rangle = \int_0^T \int_X \mathcal{L}v(t,x,u)\mu(dt, dx)$$

for all $\mu \in \mathscr{M}([0,T] \times X) = \mathscr{C}([0,T] \times X)'$ and $v \in \mathscr{C}^1([0,T] \times X)$. This operator can also be expressed as

$$\mu \mapsto \mathcal{L}'\mu = -\frac{\partial \mu}{\partial t} - \sum_{i=1}^n \frac{\partial (f_i \mu)}{\partial x_i} = -\frac{\partial \mu}{\partial t} - \operatorname{div} f\mu$$

where the derivatives of measures are understood in the sense of distributions (i.e. via their action on smooth test functions), and the change of sign comes from the integation by parts formula.

Given a test function $v \in \mathscr{C}^1([0,T] \times X)$ it follows from the above definition of the occupation measure that

$$\begin{array}{rcl} v(T, x(T \mid x_0)) & = & v(0,x_0) + \int_0^T \dot{v}(t, x(t \mid x_0))dt \\ & = & v(0,x_0) + \int_0^T \mathcal{L}v(t, x(t \mid x_0))dt \\ & = & v(0,x_0) + \int_0^T \int_X \mathcal{L}v(t,x)\mu(dt, dx \mid x_0). \end{array} \tag{4.2}$$

Definition 4.3 (Initial measure) *The initial measure $\xi_0 \in \mathscr{P}(X)$ is a probability measure that rules the distribution in space of the initial condition x_0.*

Definition 4.4 (Average occupation measure) *Given an initial measure ξ_0, the average occupation measure of the flow of trajectories is defined by*

$$\mu(A \times B) := \int_X \mu(A \times B \mid x_0)\xi_0(dx_0)$$

for all $A \in \mathscr{B}([0,T])$ and $B \in \mathscr{B}(X)$.

Example 4.2 *Returning to the scalar linear ODE of Example 4.1, with initial conditions uniformly distributed on $[0,1]$, i.e. $\xi_0(dx) = I_{[0,1]}(x)dx$, the average occupation measure is such that*

$$\mu([0,1] \times [0,a]) = \int_0^1 \mu([0,1] \times [0,a] \mid x_0)dx_0 = \int_0^a dx_0 + \int_a^{ae} \left(1 - \log \frac{x_0}{a}\right) dx_0 = a(e-1)$$

for any given $a \geq 0$.

Definition 4.5 (Terminal measure) *The terminal measure $\xi_T \in \mathscr{P}(X)$ is a probability measure that rules the distribution in space of the terminal condition $x(T)$. It is defined by*

$$\xi_T(B) := \int_X I_B(x(T \mid x_0))\xi_0(dx_0)$$

for all $B \in \mathscr{B}(X)$.

It follows by integrating equation (4.2) with respect to ξ_0 that

$$\int_X v(T,x)\xi_T(dx) = \int_X v(0,x)\xi_0(dx) + \int_0^T \int_X \mathcal{L}v(t,x)\mu(dt,dx)$$

or more concisely

$$\langle v(T,.), \xi_T \rangle = \langle v(0,.), \xi_0 \rangle + \langle \mathcal{L}v, \mu \rangle \tag{4.3}$$

which is a linear equation linking the initial measure ξ_0, the terminal measure ξ_T and the occupation measure μ, for all $v \in \mathscr{C}^1([0,T] \times X)$.

Letting

$$\mu_0(dt, dx) := \delta_0(dt)\, \xi_0(dx), \quad \mu_T(dt, dx) := \delta_T(dt)\, \xi_T(dx),$$

we can write $\langle v(0,.), \xi_0 \rangle = \langle v, \mu_0 \rangle$ and $\langle v(T,.), \xi_T \rangle = \langle v, \mu_T \rangle$. Then, equation (4.3) can be rewritten equivalently using the adjoint linear operator as

$$\langle v, \mathcal{L}'\mu \rangle = \langle v, \mu_T \rangle - \langle v, \mu_0 \rangle$$

and since this equation is required to hold for all test functions $v \in \mathscr{C}^1([0,T] \times X)$, we obtain a linear PDE on measures $\mathcal{L}'\mu = \mu_T - \mu_0$ that we write

$$\frac{\partial \mu}{\partial t} + \text{div}\, f\mu = \mu_0 - \mu_T \tag{4.4}$$

where the derivatives should be understood in the sense of distributions. This equation is classical in fluid mechanics and statistical physics, and it is called the equation of conservation of mass, or the continuity equation, or the advection equation, or **Liouville's equation**.

Note that we can disintegrate the average occupation measure as follows

$$\mu(dt, dx) = dt\, \xi(dx \mid t)$$

where $\xi(. \mid t) \in \mathscr{P}(X)$ is the conditional of μ w.r.t. t, and dt is the marginal of μ w.r.t. t, here the Lebesgue measure. Liouville's equation (4.4) can be also written as a linear PDE satisfied by probability measure ξ, namely

$$\frac{\partial \xi}{\partial t} + \text{div}\, f\xi = 0 \tag{4.5}$$

with a given initial measure $\xi(. \mid t = 0) = \xi_0$.

Proposition 4.1 *Given $\xi_0 \in \mathscr{P}(X)$, there is a unique solution $\xi(.|t) \in \mathscr{P}(X)$ solving equation (4.5). Letting $\xi_T := \xi(.|t = T) \in \mathscr{P}(X)$, there is a unique solution $\mu \in \mathscr{M}_+([0,T] \times X)$ solving equation (4.4).*

Note that in particular if $\xi_0 = \delta_{x_0}$, then $\xi(.|t) = \delta_{x(t \,|\, x_0)}$ is the Dirac measure supported on the trajectory $x(t|x_0)$ starting from x_0. The geometric picture behind Liouville's equation (4.4) is that it encodes a superposition of all classical solutions solving the Cauchy problem (4.1). The main advantage of the Liouville PDE is that it is a linear equation (in the infinite-dimensional space of measures), whereas the original Cauchy ODE is nonlinear (in the infinite-dimensional space of absolutely continuous trajectories).

4.2 Measure LP

Consider now the following dynamic optimization problem with polynomial differential constraints

$$\begin{array}{rcll} p^* & = & \inf & \int_0^T l(t, x(t))dt \\ & & \text{s.t.} & \dot{x}(t) = f(t, x(t)),\ \ x(t) \in X,\ \ t \in [0,T] \\ & & & x(0) \in X_0,\ \ x(T) \in X_T \end{array} \tag{4.6}$$

with given polynomial dynamics $f \in \mathbb{R}[t,x]$ and Lagrangian $l \in \mathbb{R}[t,x]$, and state trajectory $x(t)$ constrained in a compact basic semialgebraic set

$$X = \{x \in \mathbb{R}^n \,:\, p_k(t,x) \geq 0,\ k = 1,\ldots,n_X\}$$

for given polynomials $p_k \in \mathbb{R}[t,x]$. Finally, initial and terminal states are constrained in compact basic semialgebraic sets

$$X_0 = \{x \in \mathbb{R}^n \,:\, p_{0k}(x) \geq 0,\ k = 1,\ldots,n_0\} \subset X$$

and

$$X_T = \{x \in \mathbb{R}^n \,:\, p_{Tk}(x) \geq 0,\ k = 1,\ldots,n_T\} \subset X$$

for given polynomials $p_{0k}, p_{Tk} \in \mathbb{R}[x]$. In problem (4.6) the infimum is w.r.t. a trajectory $x(t)$ starting in X_0, ending in X_T, and staying in X.

Using the framework described in Section 4.1, we encode the state trajectory $x(t)$ in an occupation measure μ and we come up with an infinite-dimensional LP problem

$$\begin{array}{rcll} p^* & = & \inf & \int l\mu \\ & & \text{s.t.} & \int \left(\frac{\partial v}{\partial t} + (\text{grad}\, v)' f\right)\, \mu = \int v\mu_T - \int v\mu_0 \end{array} \tag{4.7}$$

for all smooth test functions $v \in \mathscr{C}^1([0,T]\times X)$ and where the infimum is w.r.t. occupation measure $\mu \in \mathscr{M}_+([0,T]\times X)$, initial measure $\mu_0 \in \mathscr{P}(\{0\}\times X_0)$, terminal measure $\mu_T \in \mathscr{P}(\{T\}\times X_T)$, and terminal time T. Note that μ_0 resp. μ_T and T can be free, or given. More abstractly, problem (4.7) can be written as a measure LP

$$\begin{array}{rcll} p^* & = & \inf & \langle l, \mu\rangle \\ & & \text{s.t.} & \frac{\partial \mu}{\partial t} + \text{div}\, f\mu = \mu_0 - \mu_T \end{array} \tag{4.8}$$

where the linear constraint is Liouville's equation, and the minimum is w.r.t. measures $(\mu,\, \mu_0,\, \mu_T) \in \mathscr{M}_+([0,T]\times X) \times \mathscr{M}_+(\{0\}\times X_0) \times \mathscr{M}_+(\{T\}\times X_T)$. If the three measures μ, μ_0 and μ_T are unknown, then an additional linear constraint like $\mu_0(\{0\}\times X_0) = 1$ or $\mu_T(\{T\}\times X_T) = 1$ must be enforced to rule out the trivial zero solution.

Remark 4.1 (Autonomous case) *If the terminal time T is free and the Lagrangian l and the dynamics f do not depend explicitly on time t, then it can be shown without loss of generality that in problem (4.8) the measures do not depend explicitly on time either, and the terminal time is equal to the mass of the occupation measure, i.e. $T = \mu(X)$. The measure LP becomes*

$$\begin{array}{rcll} p^* & = & \inf & \langle l, \mu\rangle \\ & & \text{s.t.} & \text{div}\, f\mu = \mu_0 - \mu_T \end{array} \tag{4.9}$$

where the minimum is taken w.r.t. $(\mu,\, \mu_0,\, \mu_T) \in \mathscr{M}_+(X) \times \mathscr{M}_+(X_0) \times \mathscr{M}_+(X_T)$.

Example 4.3 *Consider again the scalar linear ODE of Example 4.1*

$$\dot{x} = -x$$

with initial measure $\mu_0(dt,dx) := \delta_0(dt)\,\xi_0(dx)$ *with state distribution* $\xi_0 \in \mathscr{P}(X_0)$ *supported on*

$$X_0 := \{x \in \mathbb{R} \,:\, p_0(x) := \frac{1}{4} - \left(x - \frac{3}{2}\right)^2 \geq 0\},$$

with terminal measure $\mu_T(dt,dx) := \delta_T(dt)\,\xi_T(dx)$ *with state distribution* $\xi_T \in \mathscr{P}(X_T)$ *supported on*

$$X_T := \{x \in \mathbb{R} \,:\, p_T(x) := \frac{1}{4} - x^2 \geq 0\},$$

and with average occupation measure $\mu(dt,dx) := dt\,\xi(dx \mid t)$ *with state conditional* $\xi(dx \mid t) \in \mathscr{P}(X)$ *supported for each* $t \in [0,T]$ *on*

$$X := \{x \in \mathbb{R} \,:\, p(x) := 4 - x^2 \geq 0\}.$$

We want to find trajectories minimizing the state energy $\int_0^T x^2(t)dt$.

The linear measure problem (4.8) reads

$$\begin{array}{rcll} p^* & = & \inf & \langle x^2, \mu\rangle \\ & & \text{s.t.} & \frac{\partial \mu}{\partial t} - \text{div}\, x\mu = \mu_0 - \mu_T \end{array}$$

where the minimum is w.r.t. terminal time T *and nonnegative measures* μ, μ_0 *and* μ_T *supported respectively on* $[0,T]\times X$, $\{0\}\times X_0$ *and* $\{T\}\times X_T$, *and we have to enforce the additional normalization constraint* $\mu_0(\{0\}\times X_0) = 1$.

This problem can be solved analytically, with optimal trajectory $x(t) = e^{-t}$ *leaving* X_0 *at* $x(0) = 1$ *and reaching* X_T *at* $x(T) = \frac{1}{2}$ *for* $T = \log 2 \approx 0.6931$. *So the optimal measures solving the above LP are*

$$\mu(dt,dx) = dt\,\delta_{e^{-t}}(dx), \quad \mu_0(dt,dx) = \delta_0(dt)\,\delta_1(dx), \quad \mu_T(dt,dx) = \delta_{\log 2}(dt)\,\delta_{\frac{1}{2}}(dx)$$

and $p^* = \int_0^{\log 2} e^{-2t}dt = \frac{3}{8}$.

Alternatively, following Remark 4.1, since the trajectory optimization problem is autonomous, we can also formulate the measure LP problem

$$\begin{array}{rcll} p^* & = & \inf & \langle x^2, \mu\rangle \\ & & \text{s.t.} & -\text{div}\, x\mu = \mu_0 - \mu_T \end{array}$$

w.r.t. nonnegative measures μ, μ_0 *and* μ_T *supported respectively on* X, X_0 *and* X_T, *and the optimal solution of the problem is now*

$$\mu(dx) = \int_0^T \delta_{e^{-t}}(dx)dt, \quad \mu_0(dx) = \delta_1(dx), \quad \mu_T(dx) = \delta_{\frac{1}{2}}(dx).$$

4.3 Moment LP and LMI relaxations

Let us write problem (4.8) as as special instance of a more general measure LP

$$\begin{array}{rcll} p^* & = & \inf & \langle c, \nu\rangle \\ & & \text{s.t.} & \mathcal{A}\nu = \beta \\ & & & \nu \in \mathscr{M}_+^n \end{array}$$

where the decision variable is an n-dimensional vector of nonnegative measures ν. Linear operator $\mathcal{A} : \mathscr{M}^n \to \mathscr{M}^m$ takes an n-dimensional vector of measures and returns an m-dimensional vector of measures. The right hand side $\beta \in \mathscr{M}^m$ is a given m-dimensional vector of measures. The objective function is the duality pairing between a given n-dimensional vector of continuous functions $c \in \mathscr{C}^n$ and ν, i.e. $\langle c, \nu \rangle = \sum_{i=1}^n \langle c_i, \nu_i \rangle = \sum_{i=1}^n \int c_i \nu_i$. If we suppose that all the functions are polynomials, i.e. $a_{ij}(x) \in \mathbb{R}[x]$, $c_i(x) \in \mathbb{R}[x]$, $i = 1, \ldots, n$, $j = 1, \ldots, m$, then each measure ν_i can be manipulated via the sequence $y_i := (y_{i\alpha})_{\alpha \in \mathbb{N}}$ of its moments. The measure LP becomes a moment LP

$$\begin{array}{rcll} p^* & = & \inf & \sum_{i=1}^n \sum_\alpha c_{i\alpha} y_{i\alpha} \\ & & \text{s.t.} & \sum_{i=1}^n \sum_\alpha a_{ij\alpha} y_{i\alpha} = b_j, \;\; j = 1, \ldots, m \\ & & & y_i \text{ has a representing measure } \nu_i \in \mathscr{M}_+(X_i), \; i = 1, \ldots, n. \end{array} \tag{4.10}$$

As in Section 3.6, we use the explicit LMI conditions of Section 3.3 to model the constraints that a sequence has a representing measure. If each semialgebraic set

$$X_i := \{x \in \mathbb{R}^n \; : \; p_{ik}(x) \geq 0, \; k = 1, \ldots, n_i\}$$

satisfies Assumption 3.1 for $i = 1, \ldots, n$, problem (4.10) becomes

$$\begin{array}{rcll} p^* & = & \inf & \sum_{i=1}^n \sum_\alpha c_{i\alpha} y_{i\alpha} \\ & & \text{s.t.} & \sum_{i=1}^n \sum_\alpha a_{ij\alpha} y_{i\alpha} = b_j, \;\; j = 1, \ldots, m \\ & & & M(y_i) \geq 0, \; M(p_{ik}\, y_i) \geq 0, \; i = 1, \ldots, n, \; k = 1, \ldots, n_i. \end{array}$$

Then, in order to solve problem (4.8), we can build a hierarchy of finite-dimensional LMI relaxations. This generates a monotonically nondecreasing sequence of lower bounds asymptotically converging to p^*. Details are omitted.

Example 4.4 *At the end of Example 4.3 we came up with the autonomous measure LP*

$$\begin{array}{rcll} p^* & = & \inf & \langle x^2, \mu \rangle \\ & & \text{s.t.} & -\text{div}\, x\mu = \mu_0 - \mu_T \end{array}$$

in the decision variables $(\mu, \, \mu_0, \, \mu_T) \in \mathscr{M}_+(X) \times \mathscr{M}_+(X_0) \times \mathscr{M}_+(X_T)$ *with normalization constraint* $\mu_0(X_0) = 1$. *The corresponding moment LP problem (4.10) reads*

$$\begin{array}{rcll} p^* & = & \inf & \int x^2 \mu(dx) \\ & & \text{s.t.} & \int \mu_0(dx) = 1 \\ & & & -\alpha \int x^\alpha \mu(dx) = \int x^\alpha \mu_T(dx) - \int x^\alpha \mu_0(dx), \quad \alpha = 0, 1, 2, \ldots \end{array}$$

or equivalently

$$\begin{array}{rcll} p^* & = & \inf & y_2 \\ & & \text{s.t.} & y_{00} = 1 \\ & & & -\alpha y_\alpha = y_{T\alpha} - y_{0\alpha}, \;\; \alpha = 0, 1, 2 \ldots \\ & & & y \text{ has a representing measure } \mu \in \mathscr{M}_+(X) \\ & & & y_0 \text{ has a representing measure } \mu_0 \in \mathscr{M}_+(X_0) \\ & & & y_T \text{ has a representing measure } \mu_T \in \mathscr{M}_+(X_T) \end{array}$$

and the corresponding LMI relaxation of order r is given by

$$\begin{array}{rcll} p_r^* & = & \inf & y_2 \\ & & \text{s.t.} & y_{00} = 1 \\ & & & -\alpha y_\alpha = y_{T\alpha} - y_{0\alpha},\ \alpha = 0, 1, \ldots, 2r \\ & & & M_r(y) \geq 0,\ M_{r-1}(p\, y) \geq 0 \\ & & & M_r(y_0) \geq 0,\ M_{r-1}(p_0\, y_0) \geq 0 \\ & & & M_r(y_T) \geq 0,\ M_{r-1}(p_T\, y_T) \geq 0 \end{array} .$$

From the analytic solution described in Example 4.3 we can compute the entries of the moment vector y of measure μ, namely $y_0 = \log 2$ and

$$y_\alpha = \int x^\alpha \mu(dx) = \int_0^{\log 2} e^{-\alpha t} dt = \frac{1 - 2^{-\alpha}}{\alpha}, \quad \alpha = 1, 2, \ldots$$

4.4 Optimal trajectory recovery

Once an LMI relaxation of given order is solved, we expect vector y to contain approximate moments of the optimal occupation measure corresponding to the optimal trajectory (if it is unique), or at least a superposition (convex combination) of optimal trajectories. In some cases, we can recover approximately the trajectory from the knowledge of its moments. The dual LMI relaxations can be useful for this purpose. However, we do not elaborate further on this point in this document.

Example 4.5 *Solving the LMI relaxations of Example 4.4, we observe that the moment matrices of the initial and terminal measures both have rank one (to numerical roundoff errors), with respective moment vectors*

$$y_{0\alpha} = \int x^\alpha \mu_0(dx) = 1, \quad y_{T\alpha} = \int x^\alpha \mu_T(dx) = 2^{-\alpha}, \quad \alpha = 0, 1, 2, \ldots$$

From this it follows that $\mu_0 = \delta_1$, $\mu_T = \delta_{\frac{1}{2}}$ and the unique optimal trajectory starts from $x(0) = 1$ and reaches $x(T) = \frac{1}{2}$.

4.5 Extension to piecewise polynomial dynamics

The framework can be extended readily to differential equations with terminal cost and piecewise polynomial dynamics

$$\begin{array}{rcll} p^* & = & \inf & f_0(x(T)) + \int_0^T l(t, x(t))dt \\ & & \text{s.t.} & \dot{x}(t) = f_j(t, x(t)),\ x(t) \in X_j,\ j = 1, \ldots, N,\ t \in [0, T] \\ & & & x(0) \in X_0,\ x(T) \in X_T \end{array} \tag{4.11}$$

with given polynomial dynamics $f_j \in \mathbb{R}[t, x]$, Lagrangian $l \in \mathbb{R}[t, x]$, terminal cost $f_0 \in \mathbb{R}[x]$ and state trajectory $x(t)$ constrained in compact basic semialgebraic sets X_j.

We assume that the state-space partitioning sets, or cells X_j, are disjoint, i.e. all their respective intersections have zero Lebesgue measure in $\mathbb{R}^n$, and they all belong to a given compact semialgebraic set X, e.g. a Euclidean ball of large radius. Initial and terminal states are constrained in given compact basic semialgebraic sets X_0 and X_T.

We can then extend the measure LP framework to several measures μ_j, one supported on each cell X_j, so that the global occupation measure is

$$\mu = \sum_{j=1}^{N} \mu_j.$$

The measure LP reads

$$\begin{array}{rcll} p^* & = & \inf & \langle f_0, \mu_T \rangle + \sum_{j=1}^N \langle l, \mu_j \rangle \\ & & \text{s.t.} & \sum_{j=1}^N \left(\frac{\partial \mu_j}{\partial t} + \text{div } f_j \mu_j \right) + \mu_T = \mu_0. \end{array}$$

It can solved numerically with a hiearchy of LMI relaxations as in Section 4.3.

4.6 Back to the motivating example

To address our motivating problem of Section 1.2 we formulate an optimization problem (4.11) with systems dynamics defined as locally affine functions in three cells X_j, $j = 1, 2, 3$ corresponding respectively to the linear regime of the torque saturation

$$X_1 = \{x \in \mathbb{R}^2 \ : \ |F'x| \leq L\}, \quad f_1(x) = \left[\begin{array}{c} x_1 \\ -F'x \end{array} \right]$$

the upper saturation regime

$$X_2 = \{x \in \mathbb{R}^2 \ : \ F'x \geq L\}, \quad f_2(x) = \left[\begin{array}{c} x_1 \\ -L \end{array} \right]$$

and the lower saturation regime

$$X_3 = \{x \in \mathbb{R}^2 \ : \ F'x \leq -L\}, \quad f_3(x) = \left[\begin{array}{c} x_1 \\ L \end{array} \right].$$

The objective function has no integral term and a concave quadratic terminal term $f_0(x) = -x(T)^T x(T)$ which we would like to minimize, so as to find trajectories with terminal states of largest norm. If we can certify that for every initial state $x(0)$ chosen in X_0 the final state $x(T)$ belongs to set included in the deadzone region, we have validated our controlled system. In the measure LP problem of Section 4.5, the 3 measures μ, μ_0 and μ_T are unknown so we have to insert a normalization constraint to rule out the trivial zero solutions:

$$\begin{array}{rcll} p^* & = & \inf & \langle f_0, \mu_T \rangle + \sum_{j=1}^N \langle l, \mu_j \rangle \\ & & \text{s.t.} & \sum_{j=1}^N \left(\frac{\partial \mu_j}{\partial t} + \text{div } f_j \mu_j \right) = \mu_0 - \mu_T \\ & & & \mu_0(\{0\} \times X_0) = 1. \end{array}$$

The resulting GloptiPoly script, implementing some elementary scaling strategies to improve numerical behavior of the SDP solver, is as follows:

```
I = 27500; % inertia
kp = 2475; kd = 19800; % controller gains
L = 380; % input saturation level
dz1 = 0.2*pi/180; dz2 = 0.05*pi/180; % deadzone levels
thetamax = 50; omegamax = 5; % bounds on initial conditions
epsilon = sqrt(1e-5); % bound on norm of terminal condition
T = 50; % final time

r = input('order of relaxation ='); r = 2*r;

% measures
mpol('x1',2); m1 = meas(x1); % linear regime
mpol('x2',2); m2 = meas(x2); % upper sat
mpol('x3',2); m3 = meas(x3); % lower sat
mpol('x0',2); m0 = meas(x0); % initial
mpol('xT',2); mT = meas(xT); % terminal

% dynamics on normalized time range [0,1]
% saturation input y normalized in [-1,1]
K = -[kp kd]/L;
y1 = K*x1; f1 = T*[x1(2); L*y1/I]; % linear regime
y2 = K*x2; f2 = T*[x2(2); L/I]; % upper sat
y3 = K*x3; f3 = T*[x3(2); -L/I]; % lower set

% test functions for each measure = monomials
g1 = mmon(x1,r); g2 = mmon(x2,r); g3 = mmon(x3,r);
g0 = mmon(x0,r); gT = mmon(xT,r);

% unknown moments of initial measure
y0 = mom(g0);

% unknown moments of terminal measure
yT = mom(gT);

% input LMI moment problem
cost = mom(xT'*xT);
Ay = mom(diff(g1,x1)*f1)+...
     mom(diff(g2,x2)*f2)+...
     mom(diff(g3,x3)*f3); % dynamics
% trajectory constraints
X = [y1^2<=1; y2>=1; y3<=-1];
% initial constraints
X0 = [x0(1)^2<=thetamax^2, x0(2)^2<=omegamax^2];
% terminal constraints
XT = [xT'*xT<=epsilon^2];
% bounds on trajectory
```

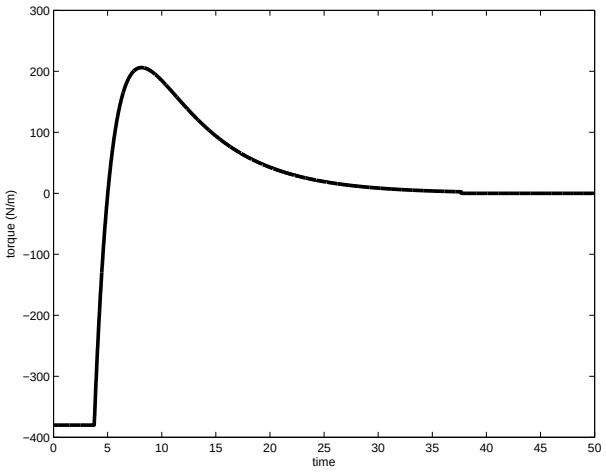

Figure 4.1: Torque input with lower saturation during approx. 7% of the time range.

```
B = [x1'*x1<=4; x2'*x2<=4; x3'*x3<=4];

% input LMI moment problem
P = msdp(max(cost), ...
         mass(m1)+mass(m2)+mass(m3)==1, ...
         mass(m0)==1, ...
         Ay==yT-y0, ...
         X, X0, XT, B);

% solve LMI moment problem
[status,obj] = msol(P)
```

With the help of this script and the SDP solver SeDuMi, we obtain the following sequence of upper bounds (since we maximize) on the maximum squared Euclidean norm of the final state:

relaxation order r	1	2	3	4
upper bound	$1.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$
CPU time (sec.)	0.2	0.5	0.7	0.9
number of moments	30	75	140	225

In the table we also indicate the CPU time (in seconds, on a standard desktop computer) and the total number of moments (size of vector y in the LMI relaxation). We see that the bound obtained at the first relaxation ($r = 1$) is not modified for higher relaxations. This clearly indicates that all initial conditions are captured in the deadzone region at time T, which is the box $[-2, 2]\frac{10^{-1}\pi}{180} \times [-5, 5]\frac{10^{-2}\pi}{180} \supset \{x \in \mathbb{R}^2 \,:\, x^T x \leq 10^{-5}\}$.

If we want to use this approach to simulate a particular trajectory, in the code we must modify the definition of the initial measure. For example for initial conditions $x_1(0) = 50$, $x_2(0) = -1$, we must insert the following sequence:

```
% given moments of initial measure = Dirac at x0
p = genpow(3,r); p = p(:,2:end); % powers
theta0 = 50; omega0 = -1; % in degrees
y0 = ones(size(p,1),1)*[theta0 omega0]*pi/180;
y0 = prod(y0.^p,2);
```

As previously, the sequence of bounds on the maximum squared Euclidean norm of the final state is constantly equal to $1.0 \cdot 10^{-5}$, and in the following table we represent as functions of the relaxation order r the masses of measures μ_k, $k = 1, 2, 3$ which are indicators of the time spent by the trajectory in the respective linear, upper saturation and lower saturation regimes:

relaxation order r	1	2	3	4	5	6	7
$\int d\mu_1$	37	89	92	92	93	93	93
$\int d\mu_2$	32	5.3	0.74	0.30	0.21	0.15	0.17
$\int d\mu_3$	32	5.1	7.1	6.9	6.8	6.9	7.0

This indicates that most of the time (approx. 93%) is spent in the linear regime, with approx. 7% of the time spent in the lower saturation regime, and a negligible amount of time is spent in the upper saturation regime. This is confirmed by simulation, see Figure 4.1.

4.7 Notes and references

Historically, the idea of reformulating nonconvex nonlinear ordinary differential equations (ODE) into convex LP, and especially linear partial differential equations (PDE) in the space of probability measures, can be tracked back to the early 19th century. It was Joseph Liouville in 1838 who first introduced the linear PDE involving the Jacobian of the transformation exerted by the solution of an ODE on its initial condition [31]. The idea was then largely expanded in Henri Poincaré's work on dynamical systems at the end of the 19th century, see in particular [43, Chapitre XII (Invariants intégraux)]. This work was pursued in the 20th century in [25], [36, Chapter VI (Systems with an integral invariant)] and more recently in the context of optimal transport by e.g. [47], [59] or [1]. The proof of Proposition 4.1 can be found e.g. in [1, Chapter 8]. Poincaré himself in [44, Section IV] mentions the potential of formulating nonlinear ODEs as linear PDEs, and this programme has been carried out to some extent by [7], see also [26], [24], [21]. Our contribution is to apply the approach described in [29], see also [13], to address polynomial trajectory optimization problems. The use of LMI and measures was also investigated in [45] for building Lyapunov barrier certificates, and based on a dual to Lyapunov's theorem described in [48]. Our approach is similar, in the sense that optimization over systems trajectories is formulated as an LP in the infinite-dimensional space of measures. This LP problem is then approached as a generalized moment problem via a hierarchy of LMI relaxations, following the strategy described extensively in [30]. Finally, our control law validation problem of Sections 1.2 and 4.6 is comprehensively described in [20].

Chapter 5

Polynomial optimal control

Our general setup for an **optimal control problem** is the following:

$$\begin{array}{rll} p^* := & \inf & \int_0^T l(t, x(t), u(t))dt \\ & \text{s.t.} & \dot{x}(t) = f(t, x(t), u(t)), \\ & & x(t) \in X, \;\; u(t) \in U, \;\; t \in [0, T], \\ & & x(0) \in X_0, \quad x(T) \in X_T \end{array} \tag{5.1}$$

where the infimum is with respect to a control law $u : [0, T] \to \mathbb{R}^m$ which is a measurable function of time with values constrained to a given set $U \subset \mathbb{R}^m$, and such that the resulting state trajectory

$$x(t \mid x_0, u) = x_0 + \int_0^t f(s, x(s), u(s))ds$$

starts at $x(0) = x_0$ in a given set X_0, terminates at time $T > 0$ in a given set X_T, and stays in a given set X in between. It is assumed that the given dynamics f is smooth, so that there is a unique trajectory given x_0 and u, which motivates our notation $x(t \mid x_0, u)$. Also given is a smooth Lagrangian l. The terminal time T is either given or free.

5.1 Controlled occupation measures

As in Chapter 4 we use occupation measures to model problem (5.1) as an infinite-dimensional LP. The main difference however is that the occupation measures will now depend on the control.

Definition 5.1 (Controlled occupation measure) *Given an initial condition x_0 and a control law $u(t)$, the controlled occupation measure of a trajectory $x(t \mid x_0, u)$ is defined as*

$$\mu(A \times B \times C \mid x_0, u) := \int_A I_B(x(t \mid x_0, u))dt$$

for all $A \in \mathscr{B}([0, T])$, $B \in \mathscr{B}(X)$ and $C \in \mathscr{B}(U)$.

A geometric interpretation is that μ measures the time spent by the graph of the trajectory $(t, x(t|x_0, u), u(t))$ in a given subset $A \times B \times C$ of $[0, T] \times X \times U$. An analytic interpretation is that integration w.r.t μ is equivalent to time-integration along a system trajectory.

If the initial condition $x_0 \in X$ is not a vector, but an initial probability measure $\xi_0 \in \mathscr{P}(X)$, see Definition 4.3, we can proceed as in Section 4.1 and model the whole flow of trajectories with a measure.

Definition 5.2 (Average controlled occupation measure) *Given an initial measure ξ_0 and a control law $u(t)$, the average controlled occupation measure of the flow of trajectories is defined as*

$$\mu(A \times B \times C \mid u) := \int_X \mu(A \times B \times C \mid x_0, u)\xi_0(dx_0)$$

for all $A \in \mathscr{B}([0, T])$, $B \in \mathscr{B}(X)$ and $C \in \mathscr{B}(U)$.

We also use the terminal measure ξ_T as in Definition 4.5, and let $\mu_0 := \delta_0 \, \xi_0$, $\mu_T := \delta_T \, \xi_T$. Measures μ, μ_0 and μ_T are linked by a linear PDE. Let us now derive this equation with the help of test functions v depending on t and x only. There is no dependence of v on the control variable u since the control law is an unknown in optimal control problem (5.1).

Define the linear operator $\mathcal{L} : \mathscr{C}^1([0, T] \times X) \to \mathscr{C}([0, T] \times X \times U)$ by

$$v \mapsto \mathcal{L}v := \frac{\partial v}{\partial t} + (\text{grad } v)' f$$

and its adjoint operator $\mathcal{L}' : \mathscr{C}([0, T] \times X \times U)' \to \mathscr{C}^1([0, T] \times X)'$ by

$$\mu \mapsto \mathcal{L}'\mu = -\frac{\partial \mu}{\partial t} - \text{div } f\mu.$$

Given a test function $v \in \mathscr{C}^1([0, T] \times X)$, it holds

$$\begin{aligned} v(T, x(T)) &= v(0, x(0)) + \textstyle\int_0^T \dot{v}(t, x(t \mid x_0, u)dt \\ &= v(0, x(0)) + \textstyle\int_0^T \mathcal{L}v(t, x((t \mid x_0, u), u(t))dt \\ &= v(0, x(0)) + \textstyle\int_0^T \int_X \int_U \mathcal{L}v(t, x, u)\mu(dt, dx, du \mid x_0, u) \end{aligned}$$

and integrating w.r.t. ξ_0 we obtain $\int \mathcal{L}v\mu = \int v\mu_T - \int v\mu_0$ for all v, which can be written in the sense of distributions as $\mathcal{L}'\mu = \mu_T - \mu_0$ or more explicitly

$$\frac{\partial \mu}{\partial t} + \text{div } (f\mu) = \mu_0 - \mu_T. \tag{5.2}$$

This is the **controlled Liouville equation**. The difference with the uncontrolled Liouville equation (4.4) is that both μ and f now also depend on the control variable u. An occupation measure μ satisfying equation (5.2) encodes state trajectories but also control trajectories.

5.2 Relaxed control

We can disintegrate the occupation measure as

$$\mu(dt, dx, du) = dt\, \xi(dx \mid t)\, \omega(du \mid t, x) \tag{5.3}$$

where the three components are as follows:

- dt is the time marginal, the Lebesgue measure of time, corresponding to the property that time flows uniformly;
- $\xi(dx \mid t) \in \mathscr{P}(X)$ is the distribution of state conditional on t, or state stochastic kernel, a probability measure on X for each $t \in [0, T]$, which models the state interpreted as a time-dependent random variable;
- $\omega(du|t, x) \in \mathscr{P}(U)$ is the distribution of the control conditional on t and x, or control stochastic kernel, a probability measure on U for each $t \in [0, T]$ and $x \in X$, which models the control interpreted as a time- and state-dependent random variable.

It means that instead of a control law u which is a measurable function of time in $[0, T]$ with values in U, we have a **relaxed control**, a probability measure

$$\omega \in \mathscr{P}(U)$$

parametrized in time $t \in [0, T]$ and space $x \in X$. Such parametrized probability measures are called Young measures in the calculus of variations and PDE literature. Our control, originally chosen as a measurable function (of time and state), is therefore relaxed to a probability measure (parametrized in time and state). Observe that the space of probability measures is larger than any Lebesgue space, since for the particular choice of a time-dependent Dirac measure

$$\omega(du \mid t, x) = \delta_{u(t,x)}$$

with $u(t, x) \in U$ we retrieve a classical control law which is a function of time and state.

The controlled Liouville equation (5.2) can be written as

$$\begin{aligned} \int v\mu_T - \int v\mu_0 &= \int \mathcal{L}v\mu \\ &= \int_T \int_X \int_U \left(\tfrac{\partial v(t,x)}{\partial t} + (\operatorname{grad} v(t, x))' f(t, x, u) \right) \omega(du \mid t, x)\xi(dx \mid t)dt \\ &= \int_T \int_X \left(\tfrac{\partial v(t,x)}{\partial t} + (\operatorname{grad} v(t, x))' \left[\int_U f(t, x, u)\omega(du \mid t, x) \right] \right) \xi(dx \mid t)dt \end{aligned}$$

for all test functions $v \in \mathscr{C}^1([0, T] \times X)$. It is now apparent that the trajectories modeled by the controlled Liouville equation are generated by a family of absolutely continuous trajectories of the relaxed controlled ODE

$$\dot{x}(t) = \int_U f(t, x(t), u)\omega(du \mid t, x).$$

Indeed, in optimal control problem (5.1), the original control system

$$\dot{x}(t) = f(t, x(t), u(t)), \quad u(t) \in U \tag{5.4}$$

can be interpreted as a differential inclusion

$$\dot{x}(t) \in f(t, x(t), U) := \{f(t, x(t), u) \ : \ u \in U\}$$

where the state velocity vector $\dot{x}(t)$ can be chosen anywhere in the set $f(t, x(t), U) \subset \mathbb{R}^n$. In contrast, any triplet of measures (μ, μ_0, μ_T) satisfying the controlled Liouville equation (5.2) corresponds to a family of trajectories of the relaxed, or convexified differential inclusion

$$\dot{x}(t) \in \text{conv } f(t, x(t), U).$$

In that sense, the set of trajectories modeled by the controlled Liouville equation (5.2) is larger than the set of trajectories of the control system (5.4). As will be seen in the numerical example section, this is an advantage of the occupation measure framework, in the sense that we will be able to construct relaxed or stochastic control laws that cannot can obtained using functions.

Based on the above discussion, we can define the **relaxed optimal control problem**

$$\begin{array}{rll} p_R^* := & \inf & \int_0^T l(t, x(t), u(t))dt \\ & \text{s.t.} & \dot{x}(t) \in \text{conv } f(t, x(t), U), \\ & & x(t) \in X, \ \ u(t) \in U, \ \ t \in [0, T], \\ & & x(0) \in X_0, \quad x(T) \in X_T \end{array} \tag{5.5}$$

and it holds $p_R^* \leq p^*$. Contrived optimal control problems (e.g. with stringent state constraints) can be cooked up such that $p_R^* < p^*$, but generically (in a sense to be defined rigorously, but not in this document), the following assumption will be satisfied.

Assumption 5.1 (No relaxation gap) *We assume that* $p_R^* = p^*$.

5.3 Measure LP

Using the controlled occupation measure and relaxed controls of the previous sections, and under Assumption 5.1, relaxed optimal control problem (5.5) can be formulated as an infinite-dimensional measure LP

$$\begin{array}{rll} p^* = & \inf & \langle l, \mu \rangle \\ & \text{s.t.} & \frac{\partial \mu}{\partial t} + \text{div } f\mu = \mu_0 - \mu_T \end{array}$$

where the infimum is w.r.t. measures $(\mu, \ \mu_0, \ \mu_T) \in \mathscr{M}_+([0, T] \times X \times U) \times \mathscr{M}_+(\{0\} \times X_0)) \times \mathscr{M}_+(\{T\} \times X_T)$. We can then rely on the results of Section 4.3 to build a hierarchy of finite-dimensional LMI relaxations for this problem. This generates a monotonically nondecreasing sequence of lower bounds asymptotically converging to p^*. Details are omitted.

Remark 5.1 (Autonomous case) *If the terminal time T is free and the Lagrangian l and the dynamics f do not depend explicitly on time t, then the measure LP becomes*

$$\begin{array}{rll} p^* = & \inf & \langle l, \mu \rangle \\ & \text{s.t.} & \operatorname{div} f\mu = \mu_0 - \mu_T \end{array}$$

where the minimum is now taken w.r.t. $(\mu,\, \mu_0,\, \mu_T) \in \mathscr{M}_+(X \times U) \times \mathscr{M}_+(X_0) \times \mathscr{M}_+(X_T)$, *i.e. the measures do not depend explicitly on time either.*

Example 5.1 (Linear quadratic regulator) *Consider the elementary scalar linear quadratic regulator problem*

$$\begin{array}{rll} p^* = & \inf & \int_0^T (x^2(t) + u^2(t))dt \\ & \text{s.t.} & \dot{x}(t) = u(t), \quad t \in [0, T] \\ & & x(0) = 1, \quad x(T) = 0 \end{array}$$

with given initial and terminal conditions. The corresponding autonomous measure LP is

$$\begin{array}{rll} p^* = & \inf & \langle x^2 + u^2, \mu \rangle \\ & \text{s.t.} & \operatorname{div} u\, \mu = \delta_1 - \delta_0 \end{array}$$

where the minimum is w.r.t. occupation measure μ. Its moment LP problem reads

$$\begin{array}{rll} p^* = & \inf & \int (x^2 + u^2)\mu(dx, du) \\ & \text{s.t.} & \alpha \int x^{\alpha-1} u \mu(dx, du) = -1, \quad \alpha = 0, 1, 2 \ldots \end{array}$$

or equivalently

$$\begin{array}{rll} p^* = & \inf & y_{20} + y_{02} \\ & \text{s.t.} & y_{01} = 2y_{11} = 3y_{21} = \cdots = -1 \\ & & M(y) \geq 0 \end{array}$$

where the minimum is w.r.t. the moments of the occupation measure

$$y_\alpha = \int x^{\alpha_1} u^{\alpha_2} \mu(dx, du), \quad \alpha = 0, 1, 2 \ldots$$

Solving the first LMI relaxation

$$\begin{array}{rll} p_1^* = & \inf & y_{20} + y_{02} \\ & \text{s.t.} & y_{01} = 2y_{11} = -1 \\ & & M_1(y) = \begin{pmatrix} y_{00} & y_{10} & y_{01} \\ y_{10} & y_{20} & y_{11} \\ y_{01} & y_{11} & y_{02} \end{pmatrix} \geq 0. \end{array}$$

with the SDP solver SeDuMi yields (rounded to 3 significant digits)

$$M_1(y^*) = \begin{pmatrix} 3.66 & 1.00 & -1.00 \\ 1.00 & 0.500 & -0.500 \\ -1.00 & -0.500 & 0.500 \end{pmatrix}.$$

This example can be solved analytically (with a scalar Riccati equation) and the solution is the state-feedback $u(t) = -x(t)$ corresponding to the optimal trajectory $x(t) = e^{-t}$ with cost $p^* = \int_0^\infty 2e^{-2t}dt = 1$ and the optimal occupation measure

$$\mu(dx, du) = \int_0^\infty \delta_{e^{-t}}(dx)\delta_{-e^{-t}}(du)dt$$

with moments

$$y_\alpha = (-1)^{\alpha_2} \int_0^\infty e^{-(\alpha_1+\alpha_2)t}dt, \quad \alpha \in \mathbb{N}^2$$

equal to $y_{00} = \infty$, $y_{10} = 1$, $y_{01} = -1$, $y_{20} = \frac{1}{2}$, $y_{11} = -\frac{1}{2}$, $y_{02} = \frac{1}{2}$ etc. We observe that the numerical moments y^* closely match, except the mass y_{00} which should approximate the terminal time T. Note however that for this numerical value of $T \approx 3.66$, the cost is $\int_0^T 2e^{-2t}dt \approx 0.999$ almost equal to the optimal value $p^* = 1$.

5.4 Optimal control recovery

Once an LMI relaxation of given order is solved, we expect vector y to contain approximate moments of the optimal occupation measure corresponding to the optimal trajectory (if it is unique), or at least a superposition (convex combination) of optimal trajectories. To recover the optimal control, or the optimal state trajectory, we can use the dual problem, which is a relaxation of the Hamilton-Jacobi-Bellman PDE of optimal control. However, we do not elaborate further on these techniques in this document.

5.5 Back to the motivating example

Let us come back to Bolza's example of Section 1.3. We saw that the infimum can be approached by a control sequence switching increasingly quickly between -1 and $+1$, so the idea is to relax the ODE

$$\dot{x}(t) = f(t, x(t), u(t))$$

with the following differential equation

$$\dot{x}(t) = \int f(t, x(t), u)\omega(du \mid t)$$

where $\omega(du|t)$ is a probability measure parametrized in t. State trajectories are then obtained by integration w.r.t. time and control

$$x(t) = x(0) + \int_0^t \int f(s, x(s), u)\omega(du \mid s)ds.$$

Here for the Bolza example we choose

$$\omega(du \mid t) = \frac{1}{2}(\delta_{u=-1} + \delta_{u=+1})$$

a time-independent weighted sum of two Dirac measures at $u = -1$ and $u = +1$. The relaxed state trajectory is then equal to

$$\begin{aligned} x(t) &= \tfrac{1}{2}\left(\int_0^t f(s, x(s), -1)ds + \int_0^t f(s, x(s), +1)ds\right) \\ &= \tfrac{1}{2}\left(-\int_0^t ds + \int_0^t ds\right) = 0 \end{aligned}$$

and the relaxed objective function is equal to

$$\begin{aligned} \int_0^1 \int_U l(t, x(t), u)d\omega(u \mid t)dt &= \tfrac{1}{2}\left(\int_0^1 l(t, x(t), -1) + \int_0^1 l(t, x(t), +1)\right) \\ &= \int_0^1 x^4(t)dt = 0 \end{aligned}$$

so that the infimum $p^* = 0$ is reached.

The corresponding GloptiPoly script is as follows:

```
% initial point measure
mpol('t0'); mpol('x0');

% occupation measure
mpol('t'); mpol('x'); mpol('u');
meas(t,2); meas(x,2); meas(u,2);

% final point measure
mpol('tT'); mpol('xT',1);
meas(tT,3); meas(xT,3);

r = input('order of relaxation ='); r = 2*r;

% define test function arrays
v = mmon([t;x],r);
v0 = mmon([t0; x0],r);
vT = mmon([tT; xT],r);

% dynamics f(x,u) = u
dvdt = diff(v,t) + diff(v,x)*u;

% assign initial point
assign([t0; x0],[ 0; 0]);

% input LMI moment problem
P = msdp (min(x^2 + (1-u^2)^2 ), ...
   0 == mom(dvdt) + double(v0) - mom(vT), ...
   t*(1-t) >= 0, u^2 <= 1, x^2 <= 1, ...
   tT == 1, xT^2 <= 1);

% solve LMI moment problem
[status, obj]  =  msol(P)
```

5.6 Notes and references

The use of relaxations and LP formulations of optimal control problems (on ordinary differential equations and partial differential equations) is classical, and can be traced back to the work by L. C. Young [61], Filippov [12], Warga [60], Gamkrelidze [15] and Rubio [54] amongst many others. For more details and a historical survey, see e.g. [11, Part III]. Parametrized measures arising in the disintegration (5.3) of the occupation measures are called Young measures in the PDE literature, see e.g. [42] or [52]. Our contribution is to notice that hierarchies of LMI relaxations can be used to solve numerically the infinite-dimensional LP on measures arising from relaxed optimal control problems with polynomial data, following the methodology described originally in [29]. By the Filippov-Ważewski relaxation theorem [2], the trajectories of the optimal control problem (5.1) are dense (w.r.t. the metric of uniform convergence of absolutely continuous functions of time) in the set of trajectories of the relaxed optimal control problem (5.5). This justifies Assumption 5.1. Finally, our motivating Bolza problem of Sections 1.3 and 5.5 is a classical example of calculus of variations illustrating that an optimal control problem with smooth data (infinitely differentiable Lagrangian and dynamics, no state or input constraints) can have a highly oscillatory optimal solution, see e.g. [10, Example 4.8].

Bibliography

[1] L. Ambrosio, N. Gigli, G. Savaré. Gradient flows in metric spaces and in the space of probability measures. 2nd edition. Lectures in mathematics, ETH Zürich, Birkhäuser, Basel, 2008.

[2] J. P. Aubin, H. Frankowska. Set-valued analysis. Springer-Verlag, Berlin, 1990.

[3] A. Ben-Tal, A. Nemirovski. Lectures on modern convex optimization. SIAM, Philadelphia, 2001.

[4] G. Blekerman, P. A. Parrilo, R. R. Thomas (Editors). Semidefinite optimization and convex algebraic geometry. SIAM, Philadelphia, 2013.

[5] S. Boyd, L. El Ghaoui, E. Feron, V. Balakrishnan. Linear matrix inequalities in system and control theory. SIAM, Philadelphia, 1994.

[6] S. Boyd, L. Vandenberghe. Convex optimization. Cambridge Univ. Press, Cambridge, UK, 2005.

[7] T. Carleman. Application de la théorie des équations intégrales linéaires aux systèmes d'équations différentielles non linéaires. Acta Math. 59:63-87, 1932.

[8] D. A. Cox, J. B. Little, D. O'Shea. Ideals, varieties, and algorithms. 3rd edition, Springer-Verlag, New York, 2007.

[9] R. Curto, L. Fialkow. Truncated K-moment problems in several variables. J. Operator Theory, 54:189-226, 2005.

[10] B. Dacorogna. Direct methods in the calculus of variations. 2nd edition. Springer-Verlag, Berlin, 2007.

[11] H. O. Fattorini. Infinite dimensional optimization and control theory. Cambridge Univ. Press, Cambridge, UK, 1999.

[12] A. F. Filippov. Classical solutions of differential inclusions with multivalued right-hand side. SIAM J. Control 5(4):609-621, 1967.

[13] V. Gaitsgory, M. Quincampoix. Linear programming approach to deterministic infinite horizon optimal control problems with discounting. SIAM J. Control Optim. 48(4):2480-2512, 2009.

[14] S. Galeani, D. Henrion, A. Jacquemard, L. Zaccarian. Design of Marx generators as a structured eigenvalue assignment. `arXiv:1301.7741`, Jan. 2013.

[15] R. V. Gamkrelidze. Principles of optimal control theory. Plenum Press, New York, 1978. English translation of a Russian original of 1975.

[16] J. W. Helton, V. Vinnikov. Linear matrix inequality representation of sets. Comm. Pure Appl. Math. 60(5):654-674, 2007.

[17] J. W. Helton, J. Nie. Sufficient and necessary conditions for semidefinite representability of convex hulls and sets. SIAM J. Optim. 20(2):759–791, 2009.

[18] D. Henrion, J. B. Lasserre. Solving nonconvex optimization problems - How GloptiPoly is applied to problems in robust and nonlinear control. IEEE Control Systems Magazine, 24(3):72-83, 2004

[19] D. Henrion, J. B. Lasserre. Detecting global optimality and extracting solutions in GloptiPoly. pp. 293-320 in D. Henrion, A. Garulli (Editors). Positive polynomials in control. Lecture Notes on Control and Information Sciences, Vol. 312, Springer-Verlag, Berlin, 2005.

[20] D. Henrion, M. Ganet-Schoeller, S. Bennani. Measures and LMI for space launcher robust control validation. Proceedings of the IFAC Symposium on Robust Control Design, Aalborg, Denmark, June 2012. See also `arXiv:1205.2168`, May 2012.

[21] O. Hernández-Lerma, J. B. Lasserre. Markov chains and invariant probabilities. Birkhäuser, Basel, 2003.

[22] J.-B. Hiriart-Urruty, C. Lemaréchal. Fundamentals of convex analysis. Springer-Verlag, Berlin, 2001.

[23] A. N. Kolmogorov, S. V. Fomin. Introductory real analysis. Dover Publications, New York, 1970. English translation of a Russian original of 1968.

[24] K. Kowalski, W. H. Steeb. Dynamical systems and Carleman linearization. World Scientific, Singapore, 1991.

[25] N. Kryloff, N. Bogoliouboff. La théorie générale de la mesure dans son application à l'étude des systèmes dynamiques de la mécanique non linéaire. Annals Math. 38(1):65-113, 1937.

[26] A. Lasota, M. C. Mackey. Probabilistic properties of deterministic systems. Cambridge Univ. Press, Cambridge, UK, 1985.

[27] J. B. Lasserre, Optimisation globale et théorie des moments. Comptes Rendus de l'Académie des Sciences Paris, Série I, Mathématique, 331(11):929-934, 2000.

[28] J. B. Lasserre. Global optimization with polynomials and the problem of moments. SIAM J. Optim. 11(3):796–817, 2001.

[29] J. B. Lasserre, D. Henrion, C. Prieur, E. Trélat. Nonlinear optimal control via occupation measures and LMI relaxations. SIAM J. Control Optim. 47(4):1643-1666, 2008.

[30] J. B. Lasserre. Moments, positive polynomials and their applications. Imperial College Press, London, UK, 2009.

[31] J. Liouville. Sur la théorie de la variation des constantes arbitraires. Journal de Mathématiques Pures et Appliquées, 3:342-349, 1838.

[32] M. Laurent. Sums of squares, moment matrices and optimization over polynomials. Pages 157-270 in M. Putinar, S. Sullivant (Editors). Emerging applications of algebraic geometry, Vol. 149 of IMA Volumes in Mathematics and its Applications, Springer-Verlag, New York, 2009.

[33] M. Laurent, F. Rendl. Semidefinite programming and integer programming. Pages 393-514 in K. Aardal, G. Nemhauser, R. Weismantel (Editors). Handbook on Discrete Optimization, Elsevier, North Holland, 2005.

[34] D. G. Luenberger. Optimization by vector space methods. John Wiley and Sons, New York, 1969.

[35] A. Nemirovski. Advances in convex optimization: conic programming. Pages 413-444 in M. Sanz-Sol, J. Soria, J. L. Varona, J. Verdera (Editors). Proceedings of International Congress of Mathematicians, Madrid, Spain, August 2006. Vol. 1, EMS Publishing House, 2007.

[36] V. V. Nemytskii, V. V. Stepanov. Qualitative theory of differential equations. Princeton Univ. Press, Princeton, NJ, 1960. English translation of a Russian original of 1947.

[37] Y. Nesterov. Squared functional systems and optimization problems. Pages 405-440 in H. Frenk, K. Roos, T. Terlaky (Editors). High performance optimization. Kluwer Academic Publishers, Dordrecht, 2000.

[38] Y. Nesterov, A. Nemirovski. Interior-point polynomial algorithms in convex programming. SIAM, Philaldelphia, 1994.

[39] J. Nie. Optimality conditions and finite convergence of Lasserre's hierarchy. `arXiv:1206.0319`, June 2012.

[40] J. Nie, K. Ranestad, B. Sturmfels. The algebraic degree of semidefinite programming. Math. Prog. Ser. A 122(2):379-405, 2010.

[41] P. A. Parrilo. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. PhD Thesis, Calif. Inst. Tech., Pasadena, 2000.

[42] P. Pedregal. Parametrized measures and variational principles. Birkhäuser, Basel, 1997.

[43] H. Poincaré. Méthodes nouvelles de la mécanique céleste. Tome III. Gauthier-Villars, Paris, 1899.

[44] H. Poincaré. L'avenir des mathématiques. Revue générale des sciences pures et appliquées, 19:930-939, 1908.

[45] S. Prajna, A. Rantzer. Convex programs for temporal verification of nonlinear dynamical systems. SIAM J. Control Optim. 46(3):999-1021, 2007.

[46] M. Putinar. Positive polynomials on compact semi-algebraic sets. Indiana Univ. Math. J. 42:969-984, 1993.

[47] S. T. Rachev, L. Rüschendorf. Mass transportation problems. Volume I: theory. Springer-Verlag, Berlin, 1998.

[48] A. Rantzer. A dual to Lyapunov's stability theorem. Syst. Control Letters 42:161-168, 2001.

[49] J. Renegar. Hyperbolic programs, and their derivative relaxations. Found. Comput. Math. 6(1):59-79, 2006.

[50] F. Riesz, B. Sz.-Nagy. Leçons d'analyse fonctionnelle. 3ème édition. Gauthier-Villars, Paris, Akadémiai Kiadó, Budapest, 1955.

[51] R. T. Rockafellar. Convex analysis. Princeton Univ. Press, Princeton, 1970.

[52] T. Roubíček. Relaxation in optimization theory and variational calculus. Walter De Gruyter, Berlin, 1997.

[53] H. Royden, P. Fitzpatrick. Real analysis. 4th edition. Prentice Hall, Boston, 2010.

[54] J. E. Rubio. Control and optimization. The linear treatment of nonlinear problems. Manchester Univ. Press, Manchester, UK, 1986.

[55] M. Safey El Din, L. Zhi. Computing rational points in convex semi-algebraic sets and sums of squares decompositions. SIAM J. Optim. 20(6):2876-2889, 2010.

[56] C. Scheiderer. Semidefinite representation for convex hulls of real algebraic curves. `arXiv:1208.3865`, Aug. 2012.

[57] M. J. Todd. Semidefinite optimization. Acta Numerica 10:515-560, 2001.

[58] L. Vandenberghe, S. Boyd. Semidefinite programming. SIAM Review 38(1):49-95, 1996.

[59] C. Villani. Topics in optimal transportation. Amer. Math. Society, Providence, 2003.

[60] J. Warga. Optimal control of differential and functional equations. Academic Press, New York, 1972.

[61] L. C. Young. Lectures on the calculus of variations and optimal control theory, W. B. Saunders, Philadelphia, 1969.