M1 IDD

Mathematics for Data Science Project
Topic: Moment Sum-Of-Squares Hierarchy

Yacine Abdelouhab

Tamazouzt Ait Eldjoudi

Sepanta Farzollahi

Instructor: Clément W. Royer

February 7, 2025

Abstract

This project aims to explore a powerful method for solving polynomial optimization problems, which are not necessarily convex, by using relaxations based on known properties of sum-of-squares polynomials. Specifically, Lasserre proved that a polynomial optimization problem can be approximated through a hierarchy of semidefinite programs (SDPs) with increasing dimensions. This hierarchy converges to an SDP whose solution is equivalent to that of the original problem.

The approach presented in this paper fully relies on sum-of-squares (SOS) polynomials.

In this document, we will systematically go through the different stages of solving polynomial optimization problems using Lasserre's relaxations. We will illustrate these stages through an example, which highlights the application of the first two relaxations. Our goal is to simplify the concepts as much as possible so that they can be understood without any prerequisites beyond those covered in the "Mathematics for Data Science" course.

Contents

Chapter 1

Dual Formulation of a Semidefinite Program (SDP)

The aim of this chapter is to provide an equivalent formulation of semidefinite programming, which we will use to define our relaxations. Additionally, we will present the formula for the dual of an SDP, which is useful for solving such problems. However,in the application we will present at the end, we do not solve any SDP explicitly, as we only compute the first two relaxations—insufficient for the hierarchy to converge to the optimal solution of our initial problem.

1.1 Problem Definition

Let $n \in \mathbb{N}^*$, $l \in \mathbb{N}^*$, and $b \in \mathbb{R}^l$. Denote by $\mathbb{R}^{n\times n}_{\text{sym}}$ the space of symmetric matrices of size $n\times n$. Let A_k (with $k \in \{0, \dots, n\}$) represent matrices where each $A_k = (a^{(k)}_{ij})_{1\leq i,j\leq n} \in \mathbb{R}^{n\times n}_{\text{sym}}$.

The optimization problem (P) defined below is known as a Semidefinite Program (SDP):

$$
\text{(P)} \quad \begin{cases} \text{minimize}_{X\in\mathbb{R}^{n\times n}_{\text{sym}}} & \text{trace}(A_0^T X), \\ \text{subject to} & \text{trace}(A_k^T X) - b_k = 0, \ \forall k = 1, \dots, l, \\ & X \succeq 0, \ X \in \mathbb{R}^{n\times n}_{\text{sym}}. \end{cases} \tag{1.1}
$$

where $X \succeq 0$ indicates that X is a symmetric positive semidefinite matrix.

1.2 Equivalent Formulation

The problem (P) can be equivalently written in expanded form as:

$$
\text{(P')} \begin{cases} \text{minimize}_{X\in\mathbb{S}^n} & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} a^{(0)}_{ij} x_{ij}, \\ \text{subject to} & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} a^{(k)}_{ij} x_{ij} = b_k, \ \forall k = 1, \dots, l, \\ & X \succeq 0. \end{cases} \tag{1.2}
$$

Here, $\mathbb{S}^n$ denotes the space of symmetric matrices of order n. The objective function and the constraints are linear, except for the semidefinite constraint $X \succeq 0$, which is nonlinear. This corresponds to a standard linear program with an additional nonlinearity due to $X \succeq 0$.

1.3 Positive Semidefinite Property

Proposition 1.1. *Z is positive semidefinite (i.e., $Z \succeq 0$) if and only if for all $X \in S_+^n$, $\text{trace}(Z^T X) \geq 0$, where S_+^n is the space of semidefinite matrices* (d'Aspremont (2020)).

Proof Idea. The equivalence can be shown using the spectral decomposition of Z and X, followed by an explicit calculation of $\text{trace}(Z^T X)$.

1.4 Lagrangian Formulation

Lemma 1.1. *Consider the problem $(\tilde{P})$ where we want to minimize $f(X)$ subject to $X \in S_+^n$, where S_+^n is the space of semidefinite matrices. To handle the constraint $X \succeq 0$, we introduce the dual variable $Z \in S_+^n$, and the Lagrangian is written as :*

$$\mathcal{L}(X, Z) = f(X) - \text{trace}(Z^T X) \tag{1.3}$$

Proposition (1.1) ensures that non-semi-definite matrices are consistently penalized. Specifically, if $X \notin S_+^n$, then by Proposition (1.1), we have $-\text{trace}(Z^T X) \geq 0$, and as a result, the value of the Lagrangian increases when $X \notin S_+^n$.

This property ensures that the Lagrangian pushes X towards the feasible set S_+^n, as the process seeks to minimize the Lagrangian. It guarantees that any violation of $X \in S_+^n$ is explicitly penalized, thus preventing infeasible solutions from being optimal.

The Lagrangian $\mathcal{L}(X, Z)$ provides a mechanism for enforcing the constraint $X \in S_+^n$ in the optimization process, and as shown, its behavior aligns with the penalization described in Proposition (1.1).

1.5 Lagrangian Derivation

We have that, for all $X \in \mathbb{R}^{n\times n}_{\text{sym}}$, $Z \in \mathbb{R}^{n\times n}_{\text{sym}}$, and $\mu \in \mathbb{R}^l$:

$$\mathcal{L}(X, Z, \mu) = \text{trace}(A_0^T X) - \text{trace}(Z^T X) + \sum_{k=1}^{l} \mu_k \left(\text{trace}(A_k^T X) - b_k\right). \tag{1.4}$$

This is equivalent to :

$$\mathcal{L}(X, Z, \mu) = \text{trace}((A_0 - Z)^T X) + \sum_{k=1}^{l} \mu_k \left(\text{trace}(A_k^T X) - b_k\right). \tag{1.5}$$

Expanding further, we obtain:

$$\mathcal{L}(X, Z, \mu) = \sum_{i=1}^{n}\sum_{j=1}^{n} \left(a_{ij}^{(0)} - z_{ij}\right) x_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{l} \mu_k a_{ij}^{(k)} x_{ij} - \sum_{k=1}^{l} \mu_k b_k. \tag{1.6}$$

Rearranging terms:

$$\mathcal{L}(X, Z, \mu) = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \left(a_{ij}^{(0)} - z_{ij} + \sum_{k=1}^{l} \mu_k a_{ij}^{(k)}\right) - \mu^T b. \tag{1.7}$$

This simplifies to:

$$\mathcal{L}(X, Z, \mu) = \text{trace}\left((A_0 - Z + \sum_{k=1}^{l} A_k \mu_k)^T X\right) - \mu^T b, \tag{1.8}$$

which is an affine function in X.

1.6 Computation of the Dual

Let's define the dual function $d : \mathbb{R}^{n\times n}_{\text{sym}} \times \mathbb{R}^l \to \mathbb{R}$ as:

$$d(Z, \mu) = \begin{cases} \inf_{X \in \mathbb{R}^{n\times n}_{\text{sym}}} \mathcal{L}(X, Z, \mu), & \text{if } Z \in S^n_+, \\ -\infty, & \text{otherwise.} \end{cases} \tag{1.9}$$

For all $(Z, \mu) \in S^n_+ \times \mathbb{R}^l$, this simplifies to:

$$d(Z, \mu) = \inf_{X \in \mathbb{R}^{n\times n}_{\text{sym}}} \text{trace}\left(\left(A_0 - Z + \sum_{k=1}^{l} A_k \mu_k\right)^T X\right) - \mu^T b. \tag{1.10}$$

Additionally, we have the following:

$$d(Z, \mu) = -\mu^T b + \inf_{X \in \mathbb{R}^{n\times n}_{\text{sym}}} \text{trace}\left(\left(A_0 - Z + \sum_{k=1}^{l} A_k \mu_k\right)^T X\right), \tag{1.11}$$

which is an affine function of μ.

This is equivalent to:

$$d(Z, \mu) = \begin{cases} -\mu^T b, & \text{if } A_0 - Z + \sum_{k=1}^{l} A_k \mu_k = 0, \\ -\infty, & \text{otherwise.} \end{cases} \tag{1.12}$$

Thus, for all $Z \in \mathbb{R}^{n\times n}_{\text{sym}}$ and $\mu \in \mathbb{R}^l$:

$$d(Z, \mu) = \begin{cases} -\mu^T b, & \text{if } Z \succeq 0 \text{ and } Z = A_0 + \sum_{k=1}^{l} A_k \mu_k, \\ -\infty, & \text{otherwise.} \end{cases} \tag{1.13}$$

The dual problem can now be written as:

$$(D) \quad \begin{cases} \text{maximize}_{\mu \in \mathbb{R}^l} & -\mu^T b \\ \text{subject to} & A_0 + \sum_{k=1}^{l} A_k \mu_k \succeq 0. \end{cases} \tag{1.14}$$

Equivalently:

$$(D) \quad \begin{cases} \text{minimize}_{\ \mu \in \mathbb{R}^l} & \mu^T b \\ \text{subject to} & A_0 + \sum_{k=1}^{l} A_k \mu_k \succeq 0. \end{cases} \tag{1.15}$$

Chapter 2

Sum-of-Squares Polynomials

In this chapter, we will introduce the essential foundations of sum-of-squares (SOS) polynomials to ensure a shared understanding for the following steps.

2.1 Definition of Multivariate Polynomials

Let $n \in \mathbb{N}^*$, $x \in \mathbb{R}^n$, and $\alpha \in \mathbb{N}^n$. We define the monomial in x with exponent α by:

$$x^\alpha := \prod_{i=1}^{n} x_i^{\alpha_i} = x_1^{\alpha_1} \cdots x_n^{\alpha_n}. \tag{2.1}$$

A polynomial f in x is a finite linear combination of monomials in x:

$$\forall x \in \mathbb{R}^n, \quad f(x) = \sum_{\alpha \in I} f_\alpha x^\alpha, \tag{2.2}$$

where $f_\alpha \in \mathbb{R}$ and $I \subseteq \mathbb{N}^n$ [1].

Remark 2.1. *Any polynomial f can also be rewritten in matrix form as follows:*

$$\forall x \in \mathbb{R}^n, \quad f(x) = M(x)^T C, \tag{2.3}$$

where $M(x)$ is the vector of all monomials of f, and C is the vector of coefficients associated with each monomial.

The degree of a monomial is the sum of its exponents:

$$\deg(x^\alpha) = \sum_{i=1}^{n} \alpha_i. \tag{2.4}$$

The degree of a polynomial f is defined as the maximum degree among all monomials appearing in f:

$$\deg(f) = \max_{\alpha \in I} \deg(x^\alpha). \tag{2.5}$$

Finally, we denote $\mathbb{R}^n[x]$ as the set of all polynomials in x, and $\mathbb{R}^n_d[x]$ as the set of polynomials of degree at most d.

[1] The definitions in this chapter are inspired by Naldi (2017).

2.2 Sum of Square (SOS) Polynomials: An Interesting Class of Polynomials

A polynomial $f \in \mathbb{R}^n[x]$ is a *sum of square (SOS)* polynomial if there exists a finite number of polynomials $h_i(x) \in \mathbb{R}^n[x]$ for $i = 1, \ldots, m$, such that for all $x \in \mathbb{R}^m$, we have:

$$f(x) = \sum_{i=1}^{m} (h_i(x))^2. \tag{2.6}$$

Remark 2.2. *f has an even degree.*

Remark 2.3. *f is non-negative, i.e., $f(x) \geq 0$ for all $x \in \mathbb{R}^n$.*

We will denote the set of SOS polynomials by $\mathbb{R}^n_{\text{SOS}}[x]$.

Proposition 2.1. *A polynomial $f \in \mathbb{R}^n_{SOS}[x]$ if and only if there exists a positive semidefinite matrix $Q \in \mathbb{R}^{l \times l}$, where $l = \binom{n+d}{d}$ and $2d = \deg(f)$, such that for all $x \in \mathbb{R}^n$, we have:*

$$f(x) = B(x)^T Q B(x), \tag{2.7}$$

where $B(x) \in \mathbb{R}^l$ is the vector of monomials of degree less than or equal to d, and $Q \succeq 0$ is a positive semidefinite matrix.

Proof ($\Longleftarrow$): For all $x \in \mathbb{R}^n$, assume that:

$$f(x) = B(x)^T Q B(x), \tag{2.8}$$

where Q is a symmetric positive semidefinite matrix. By the spectral decomposition theorem, Q can be written as:

$$Q = P^T D P, \tag{2.9}$$

where P is an orthogonal matrix (whose columns are eigenvectors of Q) and D is a diagonal matrix containing the eigenvalues of Q, which are nonnegative.

Next, define $\sqrt{D}$ as the diagonal matrix whose entries are the square roots of the eigenvalues of Q. Substituting $Q = P^T D P$ into the expression for $f(x)$, we obtain:

$$f(x) = B(x)^T P^T \sqrt{D}\sqrt{D} P B(x). \tag{2.10}$$

Now, let us define $L = \sqrt{D}P$. This allows us to simplify the expression for $f(x)$ as:

$$f(x) = (LB(x))^T (LB(x)). \tag{2.11}$$

Finally, using the Euclidean norm $\|\cdot\|_2$, we rewrite $f(x)$ as:

$$f(x) = \|LB(x)\|_2^2. \tag{2.12}$$

Expanding the norm explicitly, we obtain:

$$f(x) = \sum_{i=1}^{l} |(LB(x))_i|^2, \tag{2.13}$$

which shows that $f(x)$ is indeed a sum of squares.

Proof ($\Longrightarrow$): Now, we show the reverse direction. Suppose $f(x)$ is given by:

$$f(x) = \sum_{i=1}^{m} (h_i(x))^2, \tag{2.14}$$

for some polynomials $h_i(x) \in \mathbb{R}^n[x]$. Assume that each $h_i(x)$ can be expressed as:

$$h_i(x) = C_i^T M(x), \tag{2.15}$$

where $C_i \in \mathbb{R}^n$ are coefficient vectors and $M(x)$ is the vector of monomials of degree less than or equal to d. Substituting this into the sum of squares expression, we get:

$$f(x) = \sum_{i=1}^{m} \left(C_i^T M(x) \right)^2. \tag{2.16}$$

This can be written in matrix form as:

$$f(x) = \begin{bmatrix} C_1^T M(x) \\ \vdots \\ C_m^T M(x) \end{bmatrix}^T \begin{bmatrix} C_1^T M(x) \\ \vdots \\ C_m^T M(x) \end{bmatrix}. \tag{2.17}$$

Expanding this expression, we get:

$$f(x) = M(x)^T C^T C M(x), \tag{2.18}$$

where $C = \begin{bmatrix} C_1^T \\ \vdots \\ C_m^T \end{bmatrix}$. Thus, we can rewrite $f(x)$ as:

$$f(x) = M(x)^T Q M(x), \tag{2.19}$$

where $Q = C^T C \succeq 0$ is positive semidefinite.

This characterization of SOS polynomials gives us an efficient way to determine if a certain polynomial f belongs to $\mathbb{R}^n_{\text{SOS}}[x]$, the set of sum of squares polynomials.

Let $f \in \mathbb{R}^n[x]$ be a polynomial of even degree. Then:

$$f \in \mathbb{R}^n_{\text{SOS}}[x] \quad \Longleftrightarrow \quad \exists Q \succeq 0 \text{ such that } f(x) = B(x)^T Q B(x), \tag{2.20}$$

where $B(x)$ is the vector of basis monomials of degree $f/2$ and Q is a positive semidefinite matrix.

To verify this condition, one must find a positive semidefinite matrix Q such that the coefficients of $f(x)$ match the coefficients of $B(x)^T Q B(x)$. This process corresponds to solving a system of linear equations.

Example: Consider the polynomial (Parrilo and Lall (2003)):

$$f(x, y) = 2x^4 + 5y^4 - x^2y^2 + 2x^3y. \tag{2.21}$$

To express $f(x, y)$ as a sum of squares, let

$$B(x,y) = \begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ y^2 \\ xy \end{bmatrix}, \tag{2.22}$$

Then:

$$f(x,y) = B(x,y)^T Q B(x,y), \tag{2.23}$$

where Q is a symmetric 6×6 matrix:

$$Q = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} \\ q_{12} & q_{22} & q_{23} & q_{24} & q_{25} & q_{26} \\ q_{13} & q_{23} & q_{33} & q_{34} & q_{35} & q_{36} \\ q_{14} & q_{24} & q_{34} & q_{44} & q_{45} & q_{46} \\ q_{15} & q_{25} & q_{35} & q_{45} & q_{55} & q_{56} \\ q_{16} & q_{26} & q_{36} & q_{46} & q_{56} & q_{66} \end{bmatrix}, \tag{2.24}$$

Expanding this matrix expression, we get:

$$\begin{aligned} f(x,y) = q_{11} &+ 2q_{12}x + 2q_{13}y + q_{22}x^2 + q_{33}y^2 + 2q_{23}xy \\ &+ 2q_{14}x^2 + 2q_{15}y^2 + 2q_{16}xy + 2q_{24}x^3 + 2q_{25}xy^2 + 2q_{26}x^2y \\ &+ 2q_{34}x^2y + 2q_{35}y^3 + 2q_{36}xy^2 + q_{44}x^4 + q_{55}y^4 + q_{66}x^2y^2 \\ &+ 2q_{45}xy^3 + 2q_{46}x^3y + 2q_{56}xy^3 + q_{33}y^2 + q_{22}x^2. \end{aligned} \tag{2.25}$$

Matching the coefficients of $f(x, y)$, we obtain the following linear constraints:

$$q_{44} = 2, \quad q_{55} = 5, \quad q_{66} + 2q_{45} = -1, \quad 2q_{46} = 2. \tag{2.26}$$

Additionally:

$$q_{ij} = 0, \quad \forall (i,j) \in (\{1..6\} \times \{1..6\}) \setminus \{(4,4),(5,5),(6,6),(4,5),(5,4),(4,6),(6,4)\}. \tag{2.27}$$

The existence of a positive semidefinite matrix Q satisfying these constraints is equivalent to the feasibility of a semidefinite programming (SDP) problem. Specifically:

$$\text{minimize } 0, \quad \text{subject to } Q \succeq 0 \text{ and the constraints in (2.26) and (2.27).} \tag{2.28}$$

Why this example? In some optimization problems, a constraint may require a polynomial to be non-negative on $\mathbb{R}^n$. Determining positivity for polynomials of degree greater than 4 is computationally hard. However, requiring a polynomial to be SOS provides a relaxation of this problem. If f is SOS, then f is positive. To check positivity, one can verify whether f is SOS. If f is SOS, it is guaranteed to be positive. Otherwise, no conclusion can be drawn about the positivity of f.

Chapter 3

Polynomial Optimization Reformulation

Now, we will finally begin exploring polynomial optimization problems and their connection to sum-of-squares (SOS) polynomials in general.

3.1 Polynomial Optimization Problem

Let $f \in \mathbb{R}^n[x]$ and $g_i \in \mathbb{R}^n[x]$ for $i \in \{1, \dots, m\}$. Consider the following optimization problem:

$$\text{(P)} \quad \begin{cases} \text{minimize}_{x \in \mathbb{R}^n} & f(x), \\ \text{subject to} & g_i(x) \geq 0, \ \forall i = 1, \dots, m. \end{cases} \tag{3.1}$$

This is called a *polynomial optimization problem*. Equality constraints, if present, can be reformulated as two inequality constraints. For example:

$$g(x) = 0 \quad \text{is equivalent to} \quad g(x) \geq 0 \quad \text{and} \quad -g(x) \geq 0.$$

3.2 Epigraph Reformulation

An equivalent representation of the problem (P) is its epigraph reformulation, which is defined as follows:

$$\text{(E)} \quad \begin{cases} \text{maximize}_{(x,\lambda) \in \mathbb{R}^n \times \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \geq 0, \\ & g_i(x) \geq 0, \ \forall i = 1, \dots, m. \end{cases} \tag{3.2}$$

Remark 3.1. *If the problem (P) is unconstrained (i.e., no $g_i(x)$), the epigraph reformulation becomes:*

$$\text{(E')} \quad \begin{cases} \text{maximize}_{(x,\lambda) \in \mathbb{R}^n \times \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \geq 0. \end{cases} \tag{3.3}$$

This is equivalent to:

$$\text{(E'')} \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & \forall x \in \mathbb{R}^n, \ f(x) - \lambda \geq 0. \end{cases} \tag{3.4}$$

3.3 SOS Relaxation of the Problem

From the results in the previous chapter, the optimization problem (E") (Equation (3.4)) can be relaxed using the concept of *Sum of Squares* (SOS). Instead of requiring that $f(x)$ is exactly equal to the original polynomial, we relax it using the SOS condition. This leads to the following Semidefinite Program (SDP) for solving the problem.

$$\text{(SOS*)} \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & \text{coefficients of } f(x) - \lambda \text{ match with } B(x)^T Q B(x), \\ & Q \succeq 0. \end{cases} \tag{3.5}$$

Here, Q is a positive semidefinite matrix that encapsulates the relaxation of the polynomial optimization problem, and $B(x)$ represents a vector of monomials of degree $f/2$.

Remark 3.2. *If $f(x) - \lambda \in \mathbb{R}^n_{SOS}[x]$, then $f(x) - \lambda \geq 0$ for all $x \in \mathbb{R}^n$. However, the converse is not always true: if $f(x) - \lambda \geq 0$, it does not guarantee that $f(x) - \lambda$ is a sum of squares.*

Remark 3.3. *For the Sum of Squares (SOS) relaxation to be valid, the polynomial $f(x)$ must have an even degree. If $f(x)$ has an odd degree, the relaxation via SOS is not possible.*

3.4 Semialgebraic Set

Let $g_i(x) \in \mathbb{R}^n[x]$ for $i \in \{1, \dots, m\}$. The set K is defined by:

$$K = \{x \in \mathbb{R}^n \mid g_i(x) \geq 0, \ \forall i \in \{1, \dots, m\}\}. \tag{3.6}$$

This set K is called a *semialgebraic set*. It corresponds to the feasible set of the optimization problem (P) as defined in Equation (3.1).

3.5 Reformulation to a Problem with Infinitely Many Constraints

We can now reformulate the epigraph formulation (previously defined in Equation (3.2)) as a problem with infinitely many constraints. Let us denote this new formulation by (E^∞):

$$(\mathrm{E}^\infty) \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \geq 0, \ \forall x \in K. \end{cases} \tag{3.7}$$

In this formulation, we aim to maximize λ, subject to the condition that for all $x \in K$, the inequality $f(x) - \lambda \geq 0$ holds. Here, K is the semialgebraic set defined in Equation (3.6).

We cannot use the relaxation (SOS*) (as defined in Equation (3.5)) to solve the problem (E^∞) because we are now restricting ourselves to the set K, which is a subset of $\mathbb{R}^n$. The relaxation (SOS*) was formulated without considering such restrictions. Thus, we need more advanced tools to relax this problem.

The next chapter will detail the search for positivity certificates over the set K to provide a proper relaxation for this problem.

Chapter 4

Moment-SOS Hierarchy and SDP Relaxations

Things are getting a bit more serious now—we will introduce new notations and definitions to account for the constraints of the problem.

4.1 Quadratic Module and Truncated Quadratic Module

Let $g_i(x) \in \mathbb{R}^n[x]$ for $i \in \{1, \dots, m\}$. We define $g = [g_i]_{i=1,\dots,m}$ as the vector containing all the constraints g_i. Let K be the semialgebraic set associated with g.

The set

$$Q(g) = \left\{ \sigma_0 + \sum_{i=1}^{m} \sigma_i g_i \;\middle|\; \sigma_i \in \mathbb{R}^n_{\mathrm{SOS}}[x],\ i = 0, \dots, m \right\} \tag{4.1}$$

is called the **quadratic module** of g. It represents a conic combination of the constraints.

For $2t \geq \max_{i=1,\dots,m} \deg(g_i)$, the set

$$Q_t(g) = \left\{ \sigma_0 + \sum_{i=1}^{m} \sigma_i g_i \;\middle|\; \sigma_i \in \mathbb{R}^n_{\mathrm{SOS}}[x],\ \deg(\sigma_0) \leq 2t,\ \deg(\sigma_i g_i) \leq 2t, \forall i \in \{1, \dots, m\} \right\} \tag{4.2}$$

is called the **t-truncated quadratic module** of g.

4.2 Positivity Certificates and Putinar's Positivstellensatz

Theorem 4.1. *Suppose $Q(g)$ is convex. Then for any $f \in Q(g)$, we have $f \geq 0$ on K (Nie (2010)).*

Theorem 4.2 (Putinar's Positivstellensatz)**.** *Suppose K is Archimedean. Then, for any f such that $f(x) > 0$ for all $x \in K$, we have $f \in Q(g)$ (Mevissen (2007) and Nie (2010)).*

Proposition 4.1. *If there exists g_i such that the set $\{x \in \mathbb{R}^n \mid g_i(x) \geq 0\}$ is compact, then K is Archimedean (Mevissen (2007)).*

Remark 4.1. *An Archimedean set is a stronger hypothesis than compactness. The* Proposition (4.1) *provides a sufficient condition for K to be Archimedean.*

4.3 First Relaxation via the Quadratic Module

Under the hypothesis of Theorem (4.1), where we assume that the quadratic module $Q(g)$ is convex, and the hypothesis of Theorem (4.2), where we assume that the set K is Archimedean, we obtain a good relaxation of (E^{∞}) (Equation (3.7)).

We replace the constraint

$$f(x) - \lambda \geq 0, \quad \forall x \in K \tag{4.3}$$

by the condition

$$f(x) - \lambda \in Q(g), \tag{4.4}$$

which is equivalent to:

$$f(x) - \lambda = \sigma_0(x) + \sum_{i=1}^{m} \sigma_i(x) g_i(x), \tag{4.5}$$

where $\sigma_i(x) \in \mathbb{R}^n_{\mathrm{SOS}}[x]$ for $i = 0, \ldots, m$, with $\deg(\sigma_0) \geq \max(\deg(f), \max_i \deg(g_i))$.

To simplify notation, define:

$$l = \max_{0 \leq i \leq m} \deg(\sigma_i). \tag{4.6}$$

Rearranging, we obtain:

$$f(x) - \lambda + \sum_{i=1}^{m} \sigma_i(x) g_i(x) = \sigma_0(x), \tag{4.7}$$

which implies that the left-hand side belongs to $\mathbb{R}^n_{\mathrm{SOS}}[x]$ according to Proposition (2.1).

Equivalently, this can be rewritten as:

$$f(x) - \lambda - \sum_{i=1}^{m} B(x)^T Q_i B(x) g_i(x) = B(x)^T Q_0 B(x), \tag{4.8}$$

where $Q_i \in \mathbb{S}^l_+$ for $i = 0, \ldots, m$, and $B(x)$ consists of monomials of degree at most $l/2$.

4.4 SOS Relaxation via Semidefinite Programming

At this stage, we can directly apply the SOS relaxation (Equation (3.5)), leading to the following semidefinite relaxation.

$$(\mathrm{SOS}^*) \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}, Q_i \in \mathbb{S}^l_+} & \lambda, \\ \text{subject to} & \text{the coefficients of } f(x) - \lambda - \sum_{i=1}^{m} B(x)^T Q_i B(x) g_i(x) \\ & \text{match with those of } B(x)^T Q_0 B(x), \\ & Q_i \succeq 0, \quad i = 0, \ldots, m. \end{cases} \tag{4.9}$$

Solving (SOS*) is equivalent to determining the degree l. However, l is unknown a priori. A natural strategy is to iteratively solve (SOS*) for increasing values of l until feasibility is achieved.

4.5 Hierarchy of SOS Relaxations

Lasserre proved that if we replace the condition (4.4) by the relaxed condition

$$f(x) - \lambda \in Q_t(g), \tag{4.10}$$

where $2t \geq \max_{i=1,\dots,m} \deg(g_i)$ and $2t \geq \deg(f)$, we obtain a hierarchy of relaxations indexed by t (Laurent (2021)).

For each t, we define the semidefinite relaxation:

$$(\mathrm{SOS})_t \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \in Q_t(g). \end{cases} \tag{4.11}$$

Each solution λ_t^* of $(\mathrm{SOS})_t$ can be computed via a semidefinite program, and we have the following convergence result:

$$\lim_{t \to \infty} \lambda_t^* = \lambda^* = \min_{x \in K} f(x), \tag{4.12}$$

where λ^* is the optimal solution of (E^∞) (Equation (3.7)) [1].

4.6 Semidefinite Formulation of the Hierarchy

For each t, the semidefinite relaxation $(\mathrm{SOS})_t$ can be written as:

$$(\mathrm{SOS})_t \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}, Q_i \in \mathbb{S}_+^{d_i}} & \lambda, \\ \text{subject to} & \text{the coefficients of the polynomial} \\ & f(x) - \lambda - \sum_{i=1}^m B_i(x)^T Q_i B_i(x) g_i(x) \\ & \text{match those of } B_0(x)^T Q_0 B_0(x), \\ & Q_i \succeq 0, \quad i = 0, \dots, m, \\ & \deg(B_i(x)^T Q_i B_i(x) g_i(x)) \leq 2t, \quad i = 1, \dots, m, \\ & \deg(B_0(x)^T Q_0 B_0(x)) \leq 2t. \end{cases} \tag{4.13}$$

Here, the vectors of monomials $B_i(x)$ are chosen such that the degree constraints are satisfied:

$$\deg(B_i(x)^T Q_i B_i(x) g_i(x)) \leq 2t, \quad \deg(B_0(x)^T Q_0 B_0(x)) \leq 2t. \tag{4.14}$$

This formulation allows for an increasingly tight approximation of λ^* as $t \to \infty$.

Remark 4.2. *The matrices Q_i belong to $\mathbb{R}^{d_i \times d_i}$, where d_i is chosen to ensure consistency of polynomial degrees in the equality constraints. This aspect will be further developed in the next chapter.*

For more details on the theoretical foundations, we refer to Jasour (2019), which served as an inspiration for this chapter.

[1] Based on Lasserre (2024).

Chapter 5

Analysis of a Given Instance

Now, we have all the necessary tools to apply these concepts in an example.

5.1 Problem Formulation

We consider the following polynomial optimization problem:

$$\text{minimize}_{x\in\mathbb{R}^3} \quad f(x) = x_1^6 + x_2^6 + x_3^6 + 3x_1^2x_2^2x_3^2 - x_1^4\left(x_2^2 + x_3^2\right) - x_2^4\left(x_3^2 + x_1^2\right) - x_3^4\left(x_1^2 + x_2^2\right) \tag{5.1}$$

subject to the constraint:

$$x_1^2 + x_2^2 + x_3^2 = 1. \tag{5.2}$$

We denote this problem as:

$$(P) \quad \begin{cases} \text{minimize}_{x\in\mathbb{R}^3} & f(x), \\ \text{subject to} & x_1^2 + x_2^2 + x_3^2 = 1. \end{cases} \tag{5.3}$$

5.2 Reformulation with Inequality Constraints

The constraint $x_1^2 + x_2^2 + x_3^2 = 1$ can be rewritten using two inequalities:

$$g_1(x) := x_1^2 + x_2^2 + x_3^2 - 1 \geq 0, \quad g_2(x) := -x_1^2 - x_2^2 - x_3^2 + 1 \geq 0. \tag{5.4}$$

Thus, the original problem is equivalent to:

$$(P) \quad \begin{cases} \text{minimize}_{x\in\mathbb{R}^3} & f(x), \\ \text{subject to} & g_1(x) \geq 0, \quad g_2(x) \geq 0. \end{cases} \tag{5.5}$$

5.3 Epigraph Reformulation and Moment-SOS Hierarchy

Using the epigraph reformulation (previously defined in Equation (3.7)), we obtain the following equivalent formulation:

$$(E^\infty) \quad \begin{cases} \text{maximize}_{\lambda\in\mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \geq 0, \quad \forall x \in K, \end{cases} \tag{5.6}$$

where $K = \{x \in \mathbb{R}^3 \mid g_1(x) \geq 0, g_2(x) \geq 0\}$.
We observe that for all $x \in \mathbb{R}^n$,

$$f(x) - \lambda = B_0(x)^T F B_0(x). \tag{5.7}$$

where $B_0(x)$ is a basis vector of monomials of degree ≤ 3:

$$B_0(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_3 \\ x_1^2 \\ x_2^2 \\ x_3^2 \\ x_1x_2 \\ x_1x_3 \\ x_2x_3 \\ x_1^3 \\ x_1^2x_2 \\ x_1^2x_3 \\ x_1x_2^2 \\ x_1x_2x_3 \\ x_1x_3^2 \\ x_2^3 \\ x_2^2x_3 \\ x_2x_3^2 \\ x_3^3 \end{bmatrix}, \tag{5.8}$$

and the coefficient matrix F is a 20×20 symmetric matrix given by:

$$F = \begin{bmatrix}
-\lambda & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & 1 & . & . & . & . & -1 & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & -1 & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & 3 & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & -1 & . & . & . & . & 1 & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & -1 \\
. & . & . & . & . & . & . & . & . & . & . & . & . & . & . & . & -1 & . & . & . \\
. & . & . & . & . & . & . & . & . & . & . & . & -1 & . & . & . & . & . & . & 1
\end{bmatrix}. \tag{5.9}$$

This formulation of $B_0(x)$ and F will be useful in the next steps.

5.4 Archimedean Property of the Set

We now show that K is Archimedean. In fact, the set

$$\{x \in \mathbb{R}^3 \mid g_2(x) \geq 0\} \tag{5.10}$$

is compact, which means it is closed and bounded in $\mathbb{R}^3$ by Proposition (4.1), and K is Archimedean. Hence, the result holds.

In fact, we can rewrite this set as follows:

$$\{x \in \mathbb{R}^3 \mid x_1^2 + x_2^2 + x_3^2 \leq 1\}, \tag{5.11}$$

which is further equivalent to:

$$\{x \in \mathbb{R}^3 \mid \|x\| \leq 1\}, \tag{5.12}$$

and finally, we have:

$$\{x \in \mathbb{R}^3 \mid g_2(x) \geq 0\} = \overline{B}(0_{\mathbb{R}^3}, 1), \tag{5.13}$$

where $\overline{B}(0_{\mathbb{R}^3}, 1)$ denotes the closed ball in $\mathbb{R}^3$ centered at $0_{\mathbb{R}^3}$ with radius 1. Since this set is closed and bounded, it is compact.

5.5 Convexity of the Quadratic Module

We define the function g as:

$$g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}. \tag{5.14}$$

We now show that the quadratic module $Q(g)$ is convex.

Let $\alpha \in [0, 1]$ and $(q, \tilde{q}) \in Q^2(g)$. There exist sums of squares (SOS) polynomials $\sigma_0, \sigma_1, \sigma_2, \tilde{\sigma}_0, \tilde{\sigma}_1, \tilde{\sigma}_2$ such that:

$$q = \sigma_0 + \sigma_1 g_1 + \sigma_2 g_2, \quad \tilde{q} = \tilde{\sigma}_0 + \tilde{\sigma}_1 g_1 + \tilde{\sigma}_2 g_2. \tag{5.15}$$

For all $x \in \mathbb{R}^3$, we have:

$$\alpha q(x) + (1-\alpha)\tilde{q}(x) = (\alpha\sigma_0(x) + (1-\alpha)\tilde{\sigma}_0(x)) + \sum_{i=1}^{2} \left(\alpha\sigma_i(x)g_i(x) + (1-\alpha)\tilde{\sigma}_i(x)g_i(x)\right). \tag{5.16}$$

Since $\alpha \geq 0$ and $1 - \alpha \geq 0$, it follows that:

$$\alpha\sigma_i(x) \in \mathbb{R}_{\mathrm{SOS}}[x] \quad \text{and} \quad (1-\alpha)\tilde{\sigma}_i(x) \in \mathbb{R}_{\mathrm{SOS}}[x], \quad \forall i \in \{0, 1, 2\}. \tag{5.17}$$

Since the sum of SOS polynomials is also an SOS polynomial, we conclude that:

$$(\alpha\sigma_i(x) + (1-\alpha)\tilde{\sigma}_i(x)) \in \mathbb{R}_{\mathrm{SOS}}[x], \quad \forall i \in \{0, 1, 2\}. \tag{5.18}$$

Thus, we obtain:

$$\alpha q(x) + (1-\alpha)\tilde{q}(x) \in Q(g), \tag{5.19}$$

which proves the convexity of $Q(g)$.

5.6 Lasserre's Relaxation and Sum of Squares Formulation

Building on the previous results, our objective is to apply Lasserre's relaxation. Thus, as established in Section (5.5), the condition $f(x) - \lambda \geq 0, \forall x \in K$, can be reformulated as the membership constraint $f(x) - \lambda \in Q_t(g)$.

Thus, we obtain the first relaxation:

$$(\mathrm{SOS})_t \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda \in Q_t(g). \end{cases} \tag{5.20}$$

This is equivalent to:

$$(\mathrm{SOS})_t \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}, \sigma_i \in \mathbb{R}_{\mathrm{SOS}}[x]} & \lambda, \\ \text{subject to} & f(x) - \lambda = \sigma_0(x) + \sum_{i=1}^{2} \sigma_i(x) g_i(x), \quad \forall x \in \mathbb{R}^3, \\ & \deg(\sigma_0) \leq 2t, \quad \deg(\sigma_i g_i) \leq 2t, \quad i \in \{1, 2\}. \end{cases} \tag{5.21}$$

Since $\deg(f) = 6$ and $\deg(g_1) = \deg(g_2) = 2$, the first relaxation starts at $2t = 6$, as lower values would not ensure equality between the polynomial $f - \lambda$ and $\sigma_0 + \sum_{i=1}^{2} \sigma_i g_i$.

We can further rewrite this as:

$$(\mathrm{SOS})_t \quad \begin{cases} \text{maximize}_{\lambda \in \mathbb{R}} & \lambda, \\ \text{subject to} & f(x) - \lambda = B_0(x)^T Q_0 B_0(x) + \sum_{i=1}^{2} B_i(x)^T Q_i B_i(x) g_i(x), \quad \forall x \in \mathbb{R}^3, \\ & Q_i \succeq 0, \quad \forall i \in \{0, 1, 2\}. \end{cases} \tag{5.22}$$

where $B_0(x)$ consists of monomials of degree at most 3 in three variables, while $B_1(x)$ and $B_2(x)$ consist of monomials of degree at most 2 in three variables.

We have:

$$Q_0 \in \mathbb{R}^{20 \times 20}, \quad Q_1 \in \mathbb{R}^{10 \times 10}, \quad Q_2 \in \mathbb{R}^{10 \times 10}, \quad B_0(x) \in \mathbb{R}^{20}, \quad B_1(x) \in \mathbb{R}^{10}. \tag{5.23}$$

and the relation:

$$f(x) - \lambda = \sigma_0(x) + \sigma_1(x)(x_1^2 + x_2^2 + x_3^2 - 1) + \sigma_2(x)(-x_1^2 - x_2^2 - x_3^2 + 1), \quad \forall x \in \mathbb{R}^3. \tag{5.24}$$

This can be rewritten as:

$$f(x) - \lambda - \overbrace{\underbrace{\sigma_1(x)}_{4}(x_1^2 + x_2^2 + x_3^2 - 1)}^{2\times 3 = 2\times t} - \overbrace{\underbrace{\sigma_2(x)}_{4}(-x_1^2 - x_2^2 - x_3^2 + 1)}^{2\times 3 = 2\times t} = \overbrace{\sigma_0(x)}^{2\times 3 = 2\times t}, \tag{5.25}$$

Next, we have the equivalent expression:

$$f(x) - \lambda + (\sigma_2(x) - \sigma_1(x))(x_1^2 + x_2^2 + x_3^2 - 1) = \sigma_0(x), \tag{5.26}$$

which simplifies further to:

$$B_0(x)^T F B_0(x) + B_1(x)^T (Q_2 - Q_1) B_1(x)(x_1^2 + x_2^2 + x_3^2 - 1) = B_0(x)^T Q_0 B_0(x). \tag{5.27}$$

Our goal is to ensure:

$$B_0(x)^T (\text{some matrices}) B_0(x) = B_0(x)^T Q_0 B_0(x), \tag{5.28}$$

to match the coefficients and establish the linear constraints.

The first constraint in Equation (5.22) can be written as:

$$B_0(x)^T F B_0(x) + B_1(x)^T (Q_2 - Q_1) B_1(x)(x_1^2 + x_2^2 + x_3^2 - 1) = B_0(x)^T Q_0 B_0(x). \tag{5.29}$$

Equivalently, we can express it as:

$$B_0(x)^T F B_0(x) + \left(\sum_{i=1}^{3} (B_1(x)x_i)^T (Q_2 - Q_1) (B_1(x)x_i) \right) + (B_1(x))^T (Q_1 - Q_2)(B_1(x)) = B_0(x)^T Q_0 B_0(x). \tag{5.30}$$

Remark 5.1. *For each $B_1(x)x_i$ (with $i = 1, 2, 3$), we can add the remaining monomials to obtain $B_0(x)$, ensuring that all necessary terms are included for consistency in the formulation. Rearrangement may be performed if required to align with the structure of the problem.*

Remark 5.2. *We arrange $Q_2 - Q_1$ accordingly by adding columns or rows of zeros, or by permuting some columns and rows, in order to maintain the correct structure for subsequent computations.*

The expression for $B_1(x)$ is given by:

$$B_1(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_3 \\ x_1^2 \\ x_2^2 \\ x_3^2 \\ x_1x_2 \\ x_1x_3 \\ x_2x_3 \end{bmatrix}. \tag{5.31}$$

For each $B_1(x)x_i$, where $i \in \{1, 2, 3\}$, we have:

$$B_1(x)x_i = \begin{bmatrix} x_i \\ x_1x_i \\ x_2x_i \\ x_3x_i \\ x_1^2x_i \\ x_2^2x_i \\ x_3^2x_i \\ x_1x_2x_i \\ x_1x_3x_i \\ x_2x_3x_i \end{bmatrix}. \tag{5.32}$$

Finally, we will have:

$$B_0(x) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_3 \\ x_1^2 \\ x_2^2 \\ x_3^2 \\ x_1x_2 \\ x_1x_3 \\ x_2x_3 \\ x_1^3 \\ x_1^2x_2 \\ x_1^2x_3 \\ x_1x_2^2 \\ x_1x_2x_3 \\ x_1x_3^2 \\ x_2^3 \\ x_2^2x_3 \\ x_2x_3^2 \\ x_3^3 \end{bmatrix}. \tag{5.33}$$

We now apply Remark (5.2) following the approach described in Remark (5.1).

Let us denote by $Q_2^{(i)} - Q_1^{(i)} \in \mathbb{R}^{20\times 20}$ the operations performed on $Q_2 - Q_1$[1].

The first constraint in Equation (5.22) can be rewritten as:

$$B_0(x)^T F B_0(x) + \left(\sum_{i=1}^{3} B_0(x)^T (Q_2^{(i)} - Q_1^{(i)}) B_0(x) \right) + B_0(x)^T (Q_1 - Q_2) B_0(x) = B_0(x)^T Q_0 B_0(x). \tag{5.34}$$

Rearranging, this is equivalent to:

$$B_0(x)^T \left(F + \sum_{i=1}^{3} (Q_2^{(i)} - Q_1^{(i)}) + Q_1 - Q_2 \right) B_0(x) = B_0(x)^T Q_0 B_0(x). \tag{5.35}$$

Since this must hold for all x, we equate the coefficients:

$$F + \sum_{i=1}^{3} (Q_2^{(i)} - Q_1^{(i)}) + Q_1 - Q_2 = Q_0. \tag{5.36}$$

From this, we extract linear constraints, preparing for a semidefinite program (SDP). The Sum-of-Squares (SOS) relaxation at level 3 is then formulated as:

$$(\mathrm{SOS}_3) \quad \begin{cases} \text{maximize}_{\lambda\in\mathbb{R},\ Q_i \succeq 0} & \lambda, \\ \text{subject to} & F + \sum_{i=1}^{3} (Q_2^{(i)} - Q_1^{(i)}) + Q_1 - Q_2 = Q_0, \\ & Q_i \succeq 0, \quad i = 0, 1, 2. \end{cases} \tag{5.37}$$

This formulation corresponds to Equation (4.13) from the previous chapter.

[1]The matrix transformation operations were implemented in Python. A text file containing the entire SDP is generated, detailing the process for (SOS_3) and (SOS_4) relaxations.

References

d'Aspremont, A. (2020), 'Semidefinite programming', Master M1 ENS Course, Convex Optimization.
URL: *https://www.di.ens.fr/~aspremon/PDF/ENS/SDPapps.pdf*

Jasour, A. (2019), 'Sum of squares for nonlinear optimization', MIT 16.S498: Risk Aware and Robust Nonlinear Planning (RARNOP) Course.
URL: *https://rarnop.mit.edu/sites/default/files/Lecture3_SOS.pdf*

Lasserre, J. B. (2024), 'The moment-sos hierarchy: Applications and related topics', *Cambridge University Press* .
URL: *https://www.cambridge.org/core/services/aop-cambridge-core/content/view/83B5483C5956603346E598010DDC69200/S0962492923000053a.pdf/the-moment-sos-hierarchy-applications-and-related-topics.pdf*

Laurent, M. (2021), 'Sums of squares, moments, and applications to polynomial optimization', Talk at the Fields Institute, University of Toronto.
URL: *http://www.fields.utoronto.ca/talks/Sums-squares-moments-and-applications-polynomial-optimization-0*

Mevissen, M. (2007), 'Introduction to concepts and advances in polynomial optimization', Lecture Notes, Institute for Operations Research, ETH Zurich.
URL: *https://people.inf.ethz.ch/fukudak/semi/optpast/FS07/opt_abs/PolynomialOptimization.pdf*

Naldi, S. (2017), 'Polynomial optimization: Equivalence of semialgebraic optimization and positivity', Lecture Notes, Université de Limoges.
URL: *https://www.unilim.fr/pages_perso/simone.naldi/notesSAO/lec1.pdf*

Nie, J. (2010), 'Algebraic methods for multivariate polynomial optimization', Lecture Notes.
URL: *https://legacy.slmath.org/attachments/workshops/327/553_Lecture-notes_week1.pdf*

Parrilo, P. A. and Lall, S. (2003), 'Semidefinite programming relaxations and algebraic optimization in control', Course given at the European Control Conference, Cambridge UK.
URL: *https://www.mit.edu/~parrilo/ecc03_course/05_sum_of_squares.pdf*