Semidefinite Programming and Polynomial Optimization

Undergraduate Semester Project(UGP-III)
Monsoon Semester 2015-16

Author:
Kirtan Padh

Supervisors:
Prof. Rajat Mittal
Prof. Shobha Madan

Abstract

Optimizing a polynomial over some set is a natural problem and subsumes several different types of optimization problems that we are interested in solving. For example every constraint satisfaction problem could be written in this form.
The *Sum of Squares(SOS)* method is a *semidefinite programming(SDP)* hierarchy that can be used to optimize polynomials over a semialgebraic set. A *Positivstellensatz* is a characterization of polynomials which are positive on a semialgebraic set. The *Unique Games Conjecture(UGC)* is a conjecture in computational complexity which is currently a very active area of research.
We start by describing *SDP*. We will then look at dual views of the problem of optimizing a polynomial over a semialgabraic set. We see how we can use positivstellensatz along with *SDPs* to solve a relaxation of this problem. We will finally look at some connections between the *UGC* and the *SOS* method.

Contents

1 Semidefinite Programming

Semidefinite programming can be said to be the generalisation of linear programming(LP). In linear programming, we optimize a linear objective function subject to a set of linear constraints over the non-negative orthant. Formally, let $c, x \in \mathbb{R}^n$. Then the following is the general form of a linear program:

$$\begin{aligned} \text{minimize} \quad & c^T x \\ \text{s.t.} \quad & a_i^T x \geq b_i \quad \forall \quad i = 1, 2, \ldots, m \\ & x \in \mathbb{R}^n_+ \end{aligned} \tag{1.1}$$

where $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \quad | \quad x \geq 0\}$ and $c, x, a_i \in \mathbb{R}^n$. We call the closed convex cone $\mathbb{R}^n_+$ the non-negative orthant. Linear programming is already a powerful tool and can be used in many applications, industrial as well as theoretical, like:

- Modelling network flow problems. Network flow has several important applications in problems such as scheduling, VLSI design, matching etc.
- Many theoretical problems can be modelled as integer linear programs($x_i \in \mathbb{Z}$). Such programs can often be relaxed to linear programs and we can get an approximation to the original problem by rounding algorithms.

In semidefinite programming, we require the variable to lie in the cone of positive semidefinite matrices, which is more general than saying that it lies in the positive orthant. Before formally defining semidefinite programming we explain what we mean by positive semidefinite matrices and give some useful characterizations for it.

1.1 Positive semidefinite matrices

The cone of positive semidefinite matrices has been studied extensively. Positive definite matrices arise naturally in many areas, including differential equations, statistics, combinatorial optimization and control theory.

Definition 1.1 (*Positive semidefinite matrix*)**.** A symmetric matrix $A \in S\mathbb{R}^{n\times n}$ is called Positive semidefinite(PSD) if $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$. We denote the set of symmetric $n \times n$ positive semidefinite matrices my $\mathcal{S}_+^n$. If A is positive semidefinite, we denote $A \succeq 0$.

We similarly define a *positive definite* matrix. A symmetric matrix $A \in S\mathbb{R}^{n\times n}$ is called Positive definite(PD) if $x^T Ax > 0$ for all $x \in \mathbb{R}^n$. We denote the set of symmetric $n\times n$ positive semidefinite matrices my $\mathcal{S}_{++}^n$. If A is positive definite, we denote $A \succ 0$.

Recall the following two properties of a real symmetric matrix A:

- All eigenvalues of A are real.
- A has orthogonal eigenvectors, which can of course be chosen to be orthonormal.

Let us see some useful characterizations of positive semidefinite matrices.

Theorem 1.2. *We now give some characterizations of positive semidefinite matrices which will be useful later. The following statements are equivalent for a real symmetric matrix A:*

1. $A \succeq 0$
2. *All eigenvalues of A are non-negative.*
3. *All principal minors of A are non-negative.*
4. $A = U^T U$ *for some matrix U.*

Proof. (1. $\Rightarrow$ 2.) Let v be an eigenvector of A and λ be the corresponding eigenvalue. Therefore, from the assumption we have $v^T Av \geq 0$. This implies $\lambda v^T v \geq 0$. So $\lambda > 0$.
(2. $\Rightarrow$ 4.) We know by the spectral theorem that we can write $A = VCV^T$ where V is orthogonal and C is diagonal. The entries of C are the eigenvalues of A, which by assumption are non-negative. We therefore write $C = D^2$, where D is a diagonal matrix with real entries. Therefore $A = VD^T DV^T = (DV^T)^T(DV^T)$.
(4. $\Rightarrow$ 1.) $x^T Ax = x^T U^T Ux = (Ux)^T(Ux) \geq 0$.
(4. $\Rightarrow$ 3.) Suppose we delete the rows and columns from A which have their index in I. We get a minor B. Construct V by deleting the columns from U with index in I.

$\det(B) = \det(V^T V) = \det(V)^2 \geq 0$.
(3. $\Rightarrow$ 2.) We prove the contrapositive by induction. Assume that A has a unit eigenvector v with eigenvalue $\lambda_v < 0$. If A has only one eigenvalue < 0 then $det(A) < 0$ and we are done. Otherwise, choose a unit eigenvector u orthogonal to v with eigenvalue $\lambda_u < 0$. Now, choose $t \in R$ so that the vector $w = v + tu$ has at least one zero coordinate, say the i^{th} one. If A_0 is the matrix obtained from A by removing the i^{th} column and row and w_0 is obtained by removing the i^{th} coordinate of w, then we have $w'^T A' w' = w^T A w = \lambda_v + t^2 \lambda_u < 0$. So, A_0 is not semidefinite (by 1. $\Leftrightarrow$ 2.). The result follows from induction, and the fact that i can be chosen arbitrarily. $\square$

It is easy to see that if M and N are positive semidefinite, and $\alpha, \beta > 0$, then $\alpha M + \beta N$ is also positive semidefinite. So the cone of positive semidefinite matrices $\mathcal{S}^n_+$ is convex. In fact $\mathcal{S}^n_+ \subseteq S\mathbb{R}^{n\times n}$ is a proper cone [1], meaning that it is satisfies the following properties:

- $\mathcal{S}^n_+$ is closed.
- $\mathcal{S}^n_+$ is convex.
- $\mathcal{S}^n_+$ is solid. That is, it is non empty.
- $\mathcal{S}^n_+$ is pointed. That is, if $x \in \mathcal{S}^n_+$ and $-x \in \mathcal{S}^n_+$, then $x = 0$.

Since $\mathcal{S}^n_+$ is a proper cone, the relation $A \succeq B \iff A - B \succeq 0$ induces a partial order on $S\mathbb{R}^{n\times n}$. This is called the Löwner partial order.

We denote the *inner product* of A and B by $A \bullet B$. It is defined as follows,

$$A \bullet B = \sum_{i,j} A_{i,j} B_{i,j}$$

Having seen some properties of positive semidefinite matrices and some definitions related to the same, let us now see the formal definition of a semidefinite program. The the following is the standard form of a semidefinite program:

$$\begin{aligned} \text{minimize} \quad & C \bullet X \\ \text{subject to} \quad & A_i \bullet X = b_i \quad \forall \quad i = 1, 2, \ldots, m \\ & X \succeq 0 \end{aligned} \tag{1.2}$$

where $C, X, A_i \in S\mathbb{R}^{n\times n}$. The variable in this program is a matrix. But to get a better intuition of how it relates to linear programming we can think of the program as having n^2 variables $x_{i,j}$, the entries of the matrix X. Then our objective function is linear these n_2 variables, and so is each constraint of the form $A_i \bullet X = b_i$. The only non-linear constraint is the constraint that $X \succeq 0$. This is now a polynomial constraint in the n^2 variables. Note that SDP is more general than LP. A general LP can be written in the form of as SDP as follows:

- Let X be a diagonal matrix with $diag(X) = x \in \mathbb{R}^n$.
- Let C be a diagonal matrix with $diag(C) = c \in \mathbb{R}^n$.
- Let A_i be a diagonal matrix with $diag(A_i) = a_i \in \mathbb{R}^n$.

In this case the SDP as stated in 1.2 is equivalent to the LP as stated in 1.1. Semidefinite programs can be solved (more precisely approximated) in polynomial time within any specified accuracy either by the ellipsoid algorithm or through interior point algorithms [2] The above algorithms produce a strictly feasible solution or slightly infeasible for some versions of the ellipsoid algorithm and in fact the problem of deciding whether a semidenite program is feasible (exactly) is still open. Just like in LP, we also have a duality theory for SDP. We look at few results about SDP duality without giving the proof.

1.2 Duality

The dual problem of the SDP in standard form is defined as:

$$\begin{aligned}
\text{maximize} \quad & \sum_{i=1}^{m} y_i b_i \\
\text{subject to} \quad & \sum_{i=1}^{m} y_i A_i + S = C \quad \forall \quad i = 1, 2, \ldots, m \\
& S \succeq 0
\end{aligned} \tag{1.3}$$

The dual can of course be easily constructed from the primal 1.2. As in linear programming, we can switch between the formats(1.2 and 1.3) with great ease, and there is no loss of generality in assuming a specific format to be the primal or the dual.

The following theorem shows weak duality for SDP.

Theorem 1.3 (Weak duality)**.** *Given a feasible solution X of 1.2 and a feasible solution (y, S) of 1.3, the duality gap is* $C \bullet X - \sum_{i=1}^{m} y_i b_i \geq 0$. *Note that* $C \bullet X - \sum_{i=1}^{m} y_i b_i = S \bullet X$. *If* $C \bullet X - \sum_{i=1}^{m} y_i b_i = 0$*, then X and (y, S) are each optimal solutions to 1.2 and 1.3 respectively, and* $S \bullet X = 0$.

Unlike in the case of LP, the primal and dual may have a finite or infinite integrality gap. The primal and/or dual may not attain their optimal values unless some regularity conditions are satisfied. We give one such regularity condition below. A strong duality theorem for SDP could be as follows.

Theorem 1.4 (Strong duality(Slater's condition))**.** *Let o_p and o_d be the optimal values of the objective functions of 1.2 and 1.3 respectively. Suppose there exists a feasible solution X of 1.2 such that $X \succeq 0$ and a feasible solution (y, S) of 1.3 with $S \succeq 0$ then both 1.2 and 1.3 attain their optimal values with $o_p = o_d$.*

1.3 Some applications in combinatorial optimization

A number of NP-hard combinatorial optimization problems have relaxations that are semidefinite programs. In many instances, the optimal solution to the SDP relaxation can be converted to a feasible solution for the original problem with provably good approximation guarantees. Let us look at some examples of the use of SDP in combinatorial optimization. A good source to read more on the topic is [3]

1.3.1 Max-cut

Given a weighted undirected graph $G = (V, E)$ and a set of edge weights $w : E \to \mathbb{R}^+$, the max-cut problem is to partition the set of edges into S and S' so that the total weight of 'cut edges' is maximized. Here a cut edge is any edge (u, v) with $u \in S$ and $v \in S'$. Max-cut problem is an NP-hard problem which has attracted many researchers over the years. Since it is NP-hard there is almost no hope of finding a polynomial time algorithm for it. So various heuristics or combination of optimization and heuristic methods have been developed to approximate this problem. Among them is the efficient algorithm of Goemans and Williamson. Their algorithm combines Semidefinite programming and a rounding procedure to produce an approximate solution to the max-cut problem. The simple algorithm of Goemans and Williamson [4] gives a surprisingly good approximation ratio of 0.876. We will describe that algorithm.

We will first formulate the problem as an integer program. the Let us assume that $V = \{1, 2 \ldots, n\}$. We use an indicator variable y_i for each vertex to denote which set it is in. That is:

$$y_i = \begin{cases} 1 & \text{if } i \in S \\ -1 & \text{if } i \in S' \end{cases}$$

Notice that for each edge (i, j), $1 - y_i y_j = 2$ when (i, j) is a cut edge, and zero otherwise. So the max-cut problem can be stated as follows:

$$\begin{aligned} \text{maximize} \quad & 1/2 \sum_{(i,j)\in E} w_{ij}(1 - y_i y_j) \\ \text{subject to} \quad & y_i \in \{1, -1\} \quad \forall \quad i = 1, 2, \ldots, n \end{aligned} \tag{1.4}$$

Now we simply relax this integer program by allowing y_is to be unit vectors. So the relaxed program can be stated as follows

$$\begin{aligned} \text{maximize} \quad & 1/2 \sum_{(i,j)\in E} w_{ij}(1 - y_i^T y_j) \\ \text{subject to} \quad & \|y_i\| = 1 \quad \forall \quad i = 1, 2, \ldots, n \end{aligned} \tag{1.5}$$

This is a semidefnite program. To show this fact we can write it in the standard form as follows. Define $U \in \mathbb{R}^{n \times n}$ so that $Ue_i = y_i$, that is, the i^{th} column of U is y_i. Also, define $W \in \mathbb{R}^{n \times n}$ so that $W_{ij} = w_{ij}$. And define $Y = U^T U$. In this case, it is easy to see that 1.5 is equivalent to the following SDP:

$$\begin{aligned} \text{maximize} \quad & -\sum_{(i,j)\in E} W \bullet Y \\ \text{subject to} \quad & trace(Y) = 1 \quad \forall \quad i = 1, 2, \ldots, n \end{aligned} \tag{1.6}$$

Now, since 1.5 is an SDP we have an efficient way to solve it. Solving it will give us some solutions for y_is. Suppose we get vectors $y_i, y_2, \ldots, y_n$ as the solution to the SDP 1.5. Now our aim is to round these solutions to obtain a solution to the max-cut. We describe the rounding approach used in Goemans and Williamson's algorithm. The rounding algorithm is as follows:

1. Choose a random vector $v \in \mathbb{R}^{n \times n}$
2. If $v^T y_i > 0$, define $y'_i = 1$, else define $y'_i = -1$

$(y'_1, y'_2, \ldots y'_n)$ is the rounded solution.

Now what remains to show is that the solution we obtained is 'good'. Let OPT denote the optimal value of 1.4 and OPT_s denote the value of the rounded solution. Clearly, $\mathsf{OPT}_s \geq \mathsf{OPT}$. We would like to show that the ratio of $\mathsf{OPT}/\mathsf{OPT}_s$ is close to one. Since this is a randomized algorithm, we try to estimate $\mathbb{E}(\mathsf{OPT}_s)$.

$$\mathbb{E}_{v\in\mathbb{R}^n}(\mathsf{OPT}_s) = \mathbb{E}_{v\in\mathbb{R}^n}\left(\sum_{(i,j)\in E}(1 - y'_i y'_j)/2\right) = \sum_{(i,j)\in E}\mathbb{E}_{v\in\mathbb{R}^n}\Big((1 - y'_i y'_j)/2\Big)$$

Observe that, by the definition of expectation

$$\mathbb{E}_{v\in\mathbb{R}^n}\left(\frac{(1 - y'_i y'_j)}{2}\right) = Pr[y'_i \neq y'_j]$$

Since y_i and y_j are unit vectors, the angle between them is $\arccos {y_i}^T y_j$. So we have

$$Pr[y'_i \neq y'_j] = \frac{\arccos {y_i}^T y_j}{\pi}$$

Therefore, we get that

$$\mathbb{E}_{v\in\mathbb{R}^n}(\mathsf{OPT}_s) = \sum_{(i,j)\in E}\frac{\arccos {y_i}^T y_j}{\pi}$$

putting $\arccos {y_i}^T y_j = \theta_{ij}$, we get

$$\sum_{(i,j)\in E} \frac{\theta_{ij}}{\pi} = \sum_{(i,j)\in E} \frac{\theta_{ij}}{\pi(1-\cos\theta_{ij})}(1-y_i^T y_j)$$
$$\geq \min_{\theta\in[0,\pi]}\left(\frac{\theta}{\pi(1-\cos\theta)}\right)\sum_{(i,j)\in E}(1-y_i^T y_j)$$
$$\geq \min_{\theta\in[0,\pi]}\left(\frac{\theta}{\pi(1-\cos\theta)}\right)\mathsf{OPT}_s$$

It turns out that

$$\min_{\theta\in[0,\pi]}\left(\frac{\theta}{\pi(1-\cos\theta)}\right) = 0.876$$

So we finally obtain

$$\mathbb{E}_{v\in\mathbb{R}^n}(\mathsf{OPT}_s) \geq 0.876.\mathsf{OPT}_s$$

Using the fact that $\mathsf{OPT}_s \geq \mathsf{OPT}$ we get that

$$\mathbb{E}_{v\in\mathbb{R}^n}(\mathsf{OPT}_s) \geq 0.876.\mathsf{OPT}_s \geq 0.876.\mathsf{OPT}$$

This amounts to a 0.876-approximation algorithm, which is quite impressive. So we saw a typical example of how semidefinite programming can help us to get a good solution for a problem. Interestingly, it has been proved that assuming the unique games conjecture, this simple algorithm is optimal and it is NP-hard to do better in the sense of giving a better approximation ratio. Mahajan and Ramesh [5] later derandomized the algorithm. Let us now take a look at the lovasz theta function.

1.3.2 Lovasz theta function

Given a graph $G = (V, E)$, the independent set of the graph is the subset S of V, such that there is no edge between any two vertices of S. Maximum independent set problem is to find out the size of maximum independent set in the given graph. The problem is known to be NP-hard and hence it is not expected to have a polynomial time algorithm. We denote the size of the maximum independent set of a graph G by $\alpha(G)$. Lovasz theta function is an upper bound on $\alpha(G)$. We denote it by $\nu(G)$

As an eigenvalue bound we can get $\nu(G)$ as follows. Consider the set $P = \{A \in \mathcal{S}^n : a_{ij} = 1 \text{ if } (i,j) \notin \text{ E } \text{ or } i = j\}$. If there exists a stable set of size k the corresponding principal submatrix of any $A \in P$ will be J_k, the all ones matrix of size k. It is easy to see that $\lambda_{max}(A) \geq \lambda_{max}(J_k)$ for any $A \in P$. As a result $\min_{A \in P} \lambda_{max}(A)$ is an upper bound on $\lambda(G)$. This bound is one of the formulations of $\nu(g)$.

This naturally leads to a semidefinite program. The largest eigenvalue of a matrix can be formulated as a semidefinite program as follows:

$$\lambda_{max} = \min\{t : tI - A \succeq 0\}$$

To express $\nu(G)$ as a semidefinite program, we observe that $A \in P$ is equivalent to $A - J$ being generated by $E(i,j)$ for $(i,j) \in E$, where $E(i,j)$ is the matrix in which all except the $(i,j)^{th}$ and $(j,i)^{th}$ entries are zero and the $(i,j)^{th}$ and $(j,i)^{th}$ entries are 1. We can thus write $\nu(g)$ as an SDP as follows:

$$\begin{array}{ll} \min & t \\ \text{subject to} & tI + \displaystyle\sum_{(i,j) \in E} x_{ij} E_{ij} \succeq J \end{array} \tag{1.7}$$

$\nu(G)$ here is the objective value of the SDP.

2 Polynomial Optimization

Polynomial optimization and the problem of global nonnegativity of polynomials are very natural problems. They are active fields of research and remain in the focus of researchers from various areas as real algebra, semidefinite programming and operator theory. We will first look at the optimization of general functions and see how the problem becomes simpler when we consider the functions to be polynomials.

We will then look at the polynomial optimization problem from the view of moments as well as representation of positive polynomials over semi-algebraic sets and see how these representation lead to algorithms for the polynomial optimization problem by relaxing the general conditions. This will define SDP hierarchies.

2.1 Global optimization of general functions

Here we consider the following global optimization problem:

$$f^* = \inf_x\{f(x) : g_i(x) \geq 0, j = 1, 2, , \dots m\} \tag{2.1}$$

where $f, g_i(x) : \mathbb{R}^n \to \mathbb{R}$. We denote the feasible set in the above problem to be K. That is,

$$K = \{x : g_j(x) \geq 0, i = 1, 2, , \dots m\}$$

This is of course an NP-hard problem in general, since all integer linear programs and integer semidefinite programs can be written in this form. Let $\mathcal{M}(K)$ be the space of finite borel measures μ on K. Then we can reformulate 2.1 as follows:

$$\begin{aligned} \rho_{mom} \quad &= \quad \min_{\mu \in \mathcal{M}(K)} \left\{ \int_K f d\mu \right\} \\ \text{subject to:} \quad & \quad \mu(K) = 1 \end{aligned} \tag{2.2}$$

To prove that this is a reformulation of 2.1 we must prove that $\rho_{mom} = f^*$.

Theorem 2.1. *$\rho_{mom} = f^*$. That is, 2.1 is equivalent to 2.2.*

Proof. suppose f^* is infinite. Let $M \in \mathbb{R}$. Choose $x \in K$ such that $f(x) > M$. We can always choose such an M because f is unbounded. Now let $\mu = \delta_x$, the dirac measure on x. Then we have $\int_K f d\mu = f(x) > M$. Since M can be chosen to be arbitrarily large, we have ρ_{mom} to be infinite.

Now suppose that f^* is finite. Since f^* is the minimum we have:

$$\begin{aligned}
&f \geq f^* \\
&\quad \Rightarrow \int_K f d\mu \geq \int_K f^* d\mu = f^* \\
&\quad \Rightarrow \int_K f d\mu \geq f^* \quad \forall \mu \in \mathcal{M}(K)
\end{aligned}$$

This gives us $\rho_{mom} \geq f^*$ On the other hand, for $x \in K$ and $\mu = \delta_x$, we have $\int_K f d\mu = f(x)$. This shows that $f^* \geq \rho_{mom}$. This proves the result. □

Also, it is easy to see that 2.2 can also be reformulated as follows:

$$\begin{aligned}
\max \quad & t \\
\text{subject to} \quad & f - t \geq 0 \quad \forall x \in K
\end{aligned} \tag{2.3}$$

We will now show that 2.2 and 2.3 are LP duals of each other.

2.1.1 Duality

Observe that 2.2 can be thought of as an infinite dimensional LP, with the value of μ at a point $x \in K$ defining a variable of the LP for each $x \in K$. Then the objective function is summation of $\mu(x)f(x)$ with x varying over K, and the constraint is that $\mu(x)$ should sum to one over K. Let $\langle f, \mu \rangle$ denote the summation over K of the pointwise product of f and μ. Then 2.2 can be stated as folllows:

$$\begin{aligned}
\max_{\mu \in \mathcal{M}(K)} \quad & \langle f, \mu \rangle \\
\text{subject to} \quad & \langle 1_K, \mu \rangle = 1
\end{aligned} \tag{2.4}$$

Note that by our definition of $\langle f, \mu \rangle$, we have that $\langle f, \mu \rangle = \int_K f d\mu$. The dimension of the LP 2.4 is equal to the cardinality of K.

As in the case of finite dimensions, we can construct the dual of 2.4. It can easily be seen that the dual comes out to be as follows:

$$\begin{aligned} \max \quad & \lambda \\ \text{subject to} \quad & f - \lambda \geq 0 \quad \forall x \in K \end{aligned} \tag{2.5}$$

This shows the duality between the two formulations of the global optimization problem for general functions. We can see from the formulations that to solve, or at least approximate the LPs, we need a good characterization of either of:

- Measures μ with support in K or
- Functions nonnegative on K

These are very hard problems and cannot be solved in general.

Let us try to see how the situation changes if we assume that f is a polynomial. So now we have

$$f(x) = \sum_{\alpha} f_\alpha X^\alpha$$

where $\alpha \in \mathbb{N}^n$ and $X^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \dots x_n^{\alpha_n}$. The the objective function of 2.4 now becomes

$$\int_K f d\mu = \sum_\alpha f_\alpha \int_K X^\alpha$$

We denote $\int_K X^\alpha$ as y_α. We define

$$\Delta = \{y = (y_\alpha) : \exists \mu \in \mathcal{M}(K) \text{ s.t. } y_\alpha = \int_K X^\alpha d\mu, \forall \alpha\}$$

So we can write 2.4 as

$$\begin{aligned} \max_{\mu \in \mathcal{M}(K)} \quad & \sum_\alpha f_\alpha y_\alpha \\ \text{subject to} \quad & y \in \Delta \end{aligned} \tag{2.6}$$

This is now a finite dimensional problem.

We can also do a similar thing for the dual. If f is a polynomial of degree d, we define the following finite dimensional convex cone

$$\Theta_d = \{g \in \mathbb{R}[x]_d : g \geq 0 \text{ on } K\}$$

The dual 2.5 can now be formulated as

$$\begin{aligned} \max \quad & \lambda \\ \text{subject to} \quad & f - \lambda \in \Theta_d \end{aligned} \tag{2.7}$$

This is also a finite dimensional problem now.

We are interested in the case when f is a polynomial and K is semi-algebraic, which is to say that $g_i \in \mathbb{R}[X]$ for all $i = 1, 2 \ldots m$ in the definition of the set K. Here $X = \{x_1, x_2 \ldots x_n\}$. In this case we have

- A nice representation of polynomials which are positive on K
- A characterization of real sequences $y = (y_\alpha), \alpha \in \mathbb{N}$, such that y is a measure sequence for some borel measure μ, meaning that

$$y_\alpha = \int_K X^\alpha d\mu \quad \forall \alpha$$

We now describe both the representation of positive polynomials and moment sequences in more detail, before giving the algorithm for the problem. Let us first look at the characterization of positive polynomials on semi-algebraic sets.

2.2 Positive polynomials

It is quite often that positivity is associated with sum of squares. Whenever we are given a positive quantity of some form, it is natural to ask whether it can be written as a sum of squares.

One of the first person to ask this question for polynomials was Hilbert. Hilbert considered whether every positive polynomial can be written as a sum of squares of polynomials. Hilbert proved in [6] the existence of a real polynomial in two variables of degree six which is nonnegative on $\mathbb{R}^2$ but not a sum of squares of real polynomials.

According to [7] Hilberts proof used some basic results from the theory of algebraic curves. Apart from this his construction is completely elementary. The first explicit example of this kind was given by T. Motzkin only in 1967. Hilbert also showed in [6] that each nonnegative polynomial in two variables of degree four is a finite sum of squares of polynomials.

As usual we denote by $R[x_1, \ldots, x_n]$ and $\mathbb{R}(x_1, \ldots, x_n)$ the ring of polynomials, respectively the field of rational functions in $x_1, \ldots, x_n$ with real coefficients. The second important work [8] of Hilbert about this topic appeared in 1893. He proved that each nonnegative polynomial $p \in \mathbb{R}[x, y]$ on $\mathbb{R}^2$ is a finite sum of squares of rational functions from $\mathbb{R}(x, y)$. This led him to ask the following question "Suppose that $f \in \mathbb{R}(x1, \ldots, x_n)$ is nonnegative at all points of R_n where f is defined. Is f a finite sum of squares of rational functions?". In fact this was Hilbert's 17th problem in the list of problems that he famously presented and is the origin of research in characterizing non-negative and positive polynomials.

According to [7] the starting point of the history of Hilberts $17th$ problem was the oral defense of the doctoral dissertation of Hermann Minkowski at the University of Konigsberg in 1885. The 21 year old Minkowski expressed his opinion that there exist real polynomials which are nonnegative on the whole $\mathbb{R}^n$ and cannot be written as finite sums of squares of real polynomials. David Hilbert was an official opponent in this defense. Hilbert wrote later that Minkowski had convinced him about the truth of this statement. This question was answered positively by Artin [9] in 1927. He showed the following result.

Theorem 2.2. *If $f \in R[x_1, \ldots, x_n]$ is nonnegative on $\mathbb{R}^n$, then there are polynomials $q, p_1, \ldots, p_k \in R[x_1, \ldots, x_n], q \neq 0$, such that*

$$f = \frac{p_1^2 + \ldots + p_k^2}{q^2}$$

Since then there have been several generalisations and variants of this theorem that have been proved for more general sets. We review some of these basic results of real algebraic geometry on the representation of positive polynomials, among which are the fundamental Positivstellensatz of Krivine, Stengle, Schmudgen, Putinar and Jacobi and Prestel.

2.2.1 Positivstellensatz and SDP

Let $K \subseteq \mathbb{R}^n$ be a basic closed semi-algebraic set, that is, there exist polynomials $g_1(x), \ldots, g_m(x)$ such that

$$K = \{x \in \mathbb{R}^n : g_j(x) \geq 0, i = 1, 2, , \ldots m\}$$

A polynomial $f(x)$ is called nonnegative on K if

$$f(u) \geq 0 \quad \forall u \in K$$

A polynomial $f(x)$ is called positive on K if

$$f(u) > 0 \quad \forall u \in K$$

We will look at some sufficient/necessary conditions for a polynomial $f(x)$ to be positive or nonnegative on a set K. Let us take care of some definitions before we move on to the main part.

A polynomial is SOS means that it is a sum of squares of polynomials. We denote the set of sum of squares polynomials by S.

Definition 2.3 (Preordering generated by a set of polynomials)**.** The preordering generated by $g_1(x), \ldots, g_m(x)$ is the set

$$P(g_1(x), \ldots, g_m(x)) = \left\{ \sum_{v \in \{0,1\}^m} \sigma_v(x) g_1(x)^{v_1} g_2(x)^{v_2} \ldots g_m(x)^{v_m} : \quad \text{each } \sigma_v \text{ is } SOS \right\}$$

From this point onwards $g_1(x)^{v_1} g_2(x)^{v_2} \ldots g_m(x)^{v_m}$ will simply be denoted as g^v for the sake of convenience.

Definition 2.4 (Quadratic module generated by a set of polynomials)**.**

$$Q(g_1(x), \ldots, g_m(x)) = \left\{ \sigma_0 + \sum_{i=1}^{m} \sigma_i(x) g_i(x) : \quad \text{each } \sigma_v \text{ is } SOS \right\}$$

Obviously all polynomials in $P(g_1(x), \ldots, g_m(x))$ and $Q(g_1(x), \ldots, g_m(x))$ are non-negative on K, but in general these sets do not exhaust the set of non negative polynomials on K. We first describe the positivstellenstz of Krivine-Stengle which describe all non-negative/positive polynomials on K.

Theorem 2.5 (Krivine-Stengle Positivstellensatz)**.** *Let $f \in \mathbb{R}[x_1, \ldots, x_n]$ and be as defined earlier. Then we have the following.*

1. *$f(x) \geq 0$ on k if and only if $\exists p, q \in P(g_1(x), \ldots, g_m(x))$ and $m \in \mathbb{N}$ such that $pf = f^{2m} + q$.*
2. *$f(x > 0)$ on k if and only if $\exists p, q \in P(g_1(x), \ldots, g_m(x))$ such that $pf = 1 + g$.*

This was proved in 1974 Stengle [10]. If we let $g_1 = 1$ and $m = 1$ in Theorem2.5(1.) we get $K = \mathbb{R}^n$ and $P(g_1(x), \ldots, g_m(x) = S$. In this special case we get Artin's theorem (Theorem 2.2).

In 1991 Schmüdgen [11] gave an even nicer representation of polynomials which are positive on K, which is called Schmüdgen's positivstellensatz. It additionally requires the set K to be compact.

Theorem 2.6 (Schmüdgen's positivstellensatz)**.** *If a polynomial f is such that $f > 0$ on a compact semi-algebraic set K, then we have $f \in P(g_1(x), \ldots, g_m(x))$.*

The work of Putinar and Jacobi and prestel gave an even simpler representation of positive polynomials.

Theorem 2.7 (Putinar's positivstellensatz)**.** *Let K be a compact semialgebraic set. Suppose $\exists N > 0, N \in \mathbb{R}$, such that*

$$N - \|x\|_2^2 \in Q(g_1(x), \ldots, g_m(x))$$

If $f(x)$ is positive on K, then $f(x) \in Q(g_1(x), \ldots, g_m(x))$.

If $N - \|x\|_2^2 \in Q(g_1(x), \ldots, g_m(x))$, we say that the quadratic module generated by $(g_1(x), \ldots, g_m(x))$ is archimedian. If $Q(g_1(x), \ldots, g_m(x))$ is archimedian, then K must be compact, but if K is compact, $Q(g_1(x), \ldots, g_m(x))$ does not necessarily have to be archimedian. In general, given a compact semi-algebraic set K generated by

$g_1(x), \ldots, g_m(x)$, it is a difficult problem to check whether $Q(g_1(x), \ldots, g_m(x))$ is archimedian. In fact the decidability of the problem was only established in 2009 in the Phd. thesis of Sven Wagner.

Putinar's positivstellensatz is more efficient than Schmüdgen's because it requires only $m+1$ SOS certificates to certify that $f > 0$ on Kas opposed to the $2^m + 1$ SOS certificates required in Schmüdgen's positivstellensatz .

Now what is of interest to us is how to get these certificates. This is again a difficult problem in general, but if we wish to check whether there exists a certificate of bounded degree, that is, if we wish to check whether there exists a certificate of non negativity for Putinar's positivstellensatz in which the degree of each σ_i is less than d, then we can do this using SDP . The analogous statement also holds for Schmüdgen's positivstellensatz. Before we go on to describe how this can be done, let us look at how we can check whether a polynomial is a sum of squares, using SDP.

Theorem 2.8. *Let $f \in \mathbb{R}[X]$ of degree $2d$ for $X = (x_1, x_1, \ldots, x_n)$. Let z be the vector of all monomials of degree $\leq d$. f is SOS if and only if*

$$f = z^T Qz \text{ for some } Q \succeq 0$$

Observe that if $Q \succeq 0$, we can write $Q = U^T U$ for some matrix U. so we have $f = z^T U^T Uz = (UZ)^T (UZ) = \|UZ\|^2 = \sum_{i=1}^{r} (u_i z)^2$. Here u_i is the u^{th} row of U. This is the *SOS* representation of f if $Q \succeq 0$. This can easily be checked using SDP. If we let $Z = zz^T$, saying that $f = z^T Qz$ is the same as saying that $f = Z \bullet Q$. So we have an SDP with the variable Q in standard primal form to check whether a polynomial is a sum of squares.

2.2.2 SDP hierarchy

We now now all the tools necessary to define an SDP hierarchy to solve the polynomial optimization problem 2.7, using the positivstellensatz of Putinar and Schmüdgen. We describe the hierarchy for Putinar's positivstellensatz.

First let us see how we can check for certificate of positivity of bounded degree, given an $f \in \mathbb{R}[X]$ and K as defined earlier. Let $[x]_d$ be the vector of all monomials of degree $\leq d$. Since we are looking for a certificate of degree $\leq 2d$ of the form in 2.2.1 , we want that $\sigma_i(x) = [x]_d^T Q_i [x]_d$ for some $Q_i \succeq 0$ as we require σ_is to be SOS. If we define $g_0 = 1$, we require the following

$$\begin{aligned} f &= \sum_{i=0}^{m} g_i(x)\sigma_i(s) \\ &= \sum_{i=0}^{m} g_i(x)[x]_d^T Q_i [x]_d \\ &= [x]_d^T \left(\sum_{i=0}^{m} g_i(x) Q_i \right) [x]_d \end{aligned}$$

Now if we consider the entries of Q_is to be the variables, the constraint that

$$f = [x]_d^T \left(\sum_{i=0}^{m} g_i(x) Q_i \right) [x]_d$$

can be written linearly in the variables by comparing the coefficients of f and the polynomial obtained on the right. Additionally we have the constraints that $Q_i \succeq 0$. So checking the certificate of nonnegativity as stated in Putinar's positivstellensatz is the same as checking the feasibility of the SDP with the following constraints

$$\begin{aligned} f &= [x]_d^T \left(\sum_{i=0}^{m} g_i(x) Q_i \right) [x]_d \\ Q_i &\succeq 0 \quad \forall i = 1, 2 \ldots m \end{aligned} \tag{2.8}$$

Note that almost the same thing also works with minor modifications if we want that the degree of $\sigma_i(x) g_i(x) \leq d$ instead of the degree of $\sigma_i(x) \leq d \quad \forall i = 1, 2 \ldots m$.

We can use this to define a hierarchy of SDPs which converge to 2.7. We define the d^{th} level of the hierarchy as follows:

$$\begin{aligned} f_d^* = \quad & \max \quad \lambda \\ & \text{subject to} \quad f - \lambda = \sigma_0 + \sum_{i=1}^{m} \sigma_i(x) g_i(x) \\ & \qquad \{\sigma_i\} \text{ are SOS} \\ & \qquad \deg(\sigma_0), \deg(\sigma_i g_i) \leq d \forall i = 1, 2 \ldots m \end{aligned} \tag{2.9}$$

The SDP hierarchy defined above is known as the *Lasserre hierarchy*. The sequence $f_1^*, f_2^*, \ldots f_d^*$ will converge to f^*. In some cases, it might even converge finitely, or we might be able to show that just a few levels of the hierarchy give us a good approximation guarantee. In fact the max-cut problem described earlier also fits into the SOS framework. For a detailed analysis and comparision of of LP and SDP hierarchies, the reader is referred to [12, 13].

We can also analogously describe an SDP hierarchy for Schmüdgen's positivstellensatz.

2.3 The moment problem

We saw an SDP hierarchy for the dual SDP 2.7 of the problem of optimizing a polynomial over a semi-algebraic set. We can also define a hierarchy from the primal point of view of moment sequences. We do not define the hierarchy but just try to give a feeling for it.The idea is that instead of giving a value y_α to every monomial X^α to describe the measure, we only give a value of measure to monomials of degree $\leq d$(and therefore to all polynomials of degree $\leq d$). So we are approximation a distribution by its low degree moments in some sense. So now the measure has been relaxed to a pseudodistribution μ, where a degree d pseudodistribution is a function

$$\mu : \mathbb{R}[x]_d \to \mathbb{R}$$

that satisfies

1. μ is linear
2. $\mu(1) = 1$
3. $\mu(h^2) \geq 0 \quad \forall h \in \mathbb{R}[x]_{d/2}$

where $\mathbb{R}[x]_d$ denotes the set of all polynomials of degree $\leq d$. Given a pseudodistribution μ, we define a moment matrix as follows.

Definition 2.9 (Moment matrix $M_d(y)$)**.** $M_d(y)(\alpha, \beta) = \mu(X^{\alpha+\beta}) = y_{\alpha+\beta}, \quad \alpha, \beta \in \mathbb{N}^n \quad |\alpha|, |\beta| \leq d$ where α, β are indexed over $\{X^\alpha\}$

We must also ensure that the constraints $\{g_i \geq 0\}$ are satisfied. Towards this purpose, we define a 'Localizing matrix'. It finally turns out that all the conditions that we require for μ can be expressed in terms of positivesemidefiniteness appropriate moment matrices and 'localizing moment matrices'. For details, the reader is referred to [12].

3 Sum of Squares and the Unique Games Conjecture

A important part of theoretical computer science is to understand which computational problems can be solved efficiently, which ones cannot, and what it is about a problem that makes it easy or hard. We are quite good at coming up with new algorithms for problems, but once we have an algorithm, it seems much harder to prove that we cannot do better. It would seem therefore that we are not that good at proving hardness results and so we make conjectures that would be helpful in proving hardness results. For example most hardness results are proved modulo the conjecture that $\mathsf{NP} \neq \mathsf{P}$. Subhash Khot's *Unique Games Conjecture*(UGC) [14] is another attempt at understanding and tackling the problem of proving hardness results. UGC conjectures that a certain approximation problem is NP-hard.

Assuming the UGC conjecture has lead to proof of a large number of hardness results in the last decade, some of which are quite surprising. For example Raghavendra [15] showed that : If UGC is true, then for every constraint satisfaction problem the best approximation ratio is given by a certain simple SDP. But the question still remains whether UGC is actually true. The scientific community is divided in their opinion regarding the truth of UGC.

We have defined an SDP hierarchy in the previous chapter. It turns out that the most promising approach to refuting the UGC is through hierarchies of that form. This is because UGC predicts that for a wide variety of problems, low degree hierarchies are optimal. For example, the algorithm for the max-cut problem can also be stated as a degree 2 SDP hierarchy. It has been proved that assuming the UGC, this algorithm is optimal. But we do not know whether higher degree hierarchies could give us better approximation guarantees, but it certainly seems plausible that they should. The UGC predicts the degree 2 hierarchy to be optimal for a wide variety of other problems also. This means that if we can show that a higher degree hierarchy can do better for any of these problems, we will have refuted the UGC. Though note that the degree must not be two high, or the algorithm will not be polynomial time.

But it turns out that analyzing the guarantees of these algorithms is a very hard problem. Barak and Steurer [16] have a nice survey in which they discuss the approach towards trying to analyze the approximation guarantees of such algorithms.

Bibliography

[1] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521833787.

[2] Farid Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1993.

[3] Michel X. Goemans. Semidefinite programming and combinatorial optimization, 1997.

[4] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, November 1995. ISSN 0004-5411. doi: 10.1145/227683.227684. URL http://doi.acm.org/10.1145/227683.227684.

[5] Sanjeev Mahajan and H. Ramesh. Derandomizing approximation algorithms based on semidefinite programming. *SIAM Journal on Computing*, 28(5):1641–1663, 1999. doi: 10.1137/S0097539796309326. URL http://dx.doi.org/10.1137/S0097539796309326.

[6] David Hilbert. Ueber die darstellung definiter formen als summe von formenquadraten. *Mathematische Annalen*, 32(3):342–350, 1888. ISSN 0025-5831. doi: 10.1007/BF01443605. URL http://dx.doi.org/10.1007/BF01443605.

[7] Konrad Schmdgen. Around hilberts 17th problem, 2012.

[8] David Hilbert. ber ternre definite formen. *Acta Mathematica*, 17(1):169–197, 1893. ISSN 0001-5962. doi: 10.1007/BF02391990. URL http://dx.doi.org/10.1007/BF02391990.

[9] Emil Artin. ber die zerlegung definiter funktionen in quadrate. *Abhandlungen aus dem Mathematischen Seminar der Universitt Hamburg*, 5(1):100–115, 1927. ISSN 0025-5858. doi: 10.1007/BF02952513. URL http://dx.doi.org/10.1007/BF02952513.

[10] Gilbert Stengle. A nullstellensatz and a positivstellensatz in semialgebraic geometry. *Mathematische Annalen*, 207:87–98, 1974. URL http://eudml.org/doc/162533.

[11] Konrad Schmdgen. The k-moment problem for compact semi-algebraic sets. *Mathematische Annalen*, 289(2):203–206, 1991. URL http://eudml.org/doc/164777.

[12] Monique Laurent. A comparison of the sherali-adams, lovsz-schrijver and lasserre relaxations for 0-1 programming. *MATHEMATICS OF OPERATIONS RESEARCH*, 28:470–496, 2001.

[13] Eden Chlamtac and Madhur Tulsiani. Convex relaxations and integrality gaps. In Miguel F. Anjos and Jean B. Lasserre, editors, *Handbook on Semidefinite, Conic and Polynomial Optimization*, volume 166 of *International Series in Operations Research and Management Science*, pages 139–169. Springer US, 2012. ISBN 978-1-4614-0768-3. doi: 10.1007/978-1-4614-0769-0_6. URL http://dx.doi.org/10.1007/978-1-4614-0769-0_6.

[14] Subhash Khot and Nisheeth K. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1. *CoRR*, abs/1305.4581, 2013. URL http://arxiv.org/abs/1305.4581.

[15] Prasad Raghavendra. Optimal algorithms and inapproximability results for every csp? In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 245–254, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-047-0. doi: 10.1145/1374376.1374414. URL http://doi.acm.org/10.1145/1374376.1374414.

[16] Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. *CoRR*, abs/1404.5236, 2014. URL http://arxiv.org/abs/1404.5236.