Iterative Calculation of Sum Of Squares
Bruno Després, Maxime Herda

To cite this version:
Bruno Després, Maxime Herda. Iterative Calculation of Sum Of Squares. 2018. <hal-01946539v2>

HAL Id: hal-01946539
https://hal.archives-ouvertes.fr/hal-01946539v2
Submitted on 7 Dec 2018
Iterative Calculation of Sum Of Squares

Bruno Després∗and Maxime Herda†

Abstract

We propose an iterative algorithm for the calculations of sum of squares of polynomials, reformulated as positive interpolation. The method is based on the definition of a dual functional $G$. The domain of $G$, the boundary of the domain and the boundary at infinity are analyzed in details. In the general case, $G$ is closed convex. For univariate polynomials in the context of the Lukacs representation, $G$ is coercive and strictly convex which yields a unique critical point. Various descent algorithms are evoked. Numerical examples are provided, for univariate and bivariate polynomials.

Keywords: Positive polynomials, sum of squares, convex programming, nonlinear programming

AMS subject classifications: 90C30, 65K05, 90C25

1 Introduction

Let $P[X] := P[X_1, \ldots, X_d]$ be the set of real polynomials with $d$ variables. The subset of polynomials of total degree less than or equal to $n \geq 1$ is denoted by $P^n[X]$. Let $\mathbb{K} \subset \mathbb{R}^d$ be a closed and convex semi-algebraic set defined through a finite number $j_*$ of polynomial inequalities

$$\mathbb{K} = \{x \in \mathbb{R}^d \text{ such that } g_j(x) \geq 0 \text{ for } g_j \in P[X], 1 \leq j \leq j_*\}. \quad (1)$$

A fundamental question related to 17th Hilbert problem [8] concerns the characterization of the set of non-negative polynomials of maximal degree $n$ on $\mathbb{K}$

$$P^n_{\mathbb{K}, +}[X] = \{p \in P^n[X] \text{ such that } p(x) \geq 0 \text{ for any } x \in K\}, \quad r_* = \dim P^n[X]. \quad (2)$$

Famous examples of such characterizations as Sum Of Squares (SOS) are the Lukacs theorem [20] or Putinar’s Positivstellensatz [18]: a recent state of the art can be found in the books of Lasserre [12, 13]; some recent algorithmic issues in the context of optimal control can be found in [10] and
therein. We focus on a version where the maximal number of squares is equal to a predefined value
\(i_\ast \geq 1\) independent of \(j\)

\[
p = \sum_{j=1}^{j_\ast} g_j \left( \sum_{i=1}^{i_\ast} p_{ij}^2 \right) = \sum_{i=1}^{i_\ast} \left( \sum_{j=1}^{j_\ast} g_j p_{ij}^2 \right) = \sum_{j=1}^{j_\ast} \sum_{i=1}^{i_\ast} g_j p_{ij}^2.
\]

In this work, we are interested in the effective calculation of such representations by means of iterative methods. Our main motivation comes from scientific computing which, viewed as a community and as a practice, needs fast algorithms which can be implemented many times with reliable results, a general reference for the discretization of hyperbolic equations with high order methods is in [19]. Some preliminary tests in this direction are in [6], but the methods were inefficient in terms of the time of restitution. In a fully different direction, one can also mention the theory of numerical approximation with splines, see [14, 1]: splines are widely used in scientific computing and computer aided design (CAD) but often needs tensorisation in multi-dimension; this limitation is not encountered by our methods. These examples are the primary explanation of our interest in iterative calculation of SOS. To our knowledge, the approach developed in this work is, at the algorithmic level, very different from SOS algorithms based on SemiDefinite Programming (SPD) like SOLLYA [4], SOSTOOLS [17] and GLOBTIPOLY [12].

Our model problem below is an extension of the notion of positive interpolation which comes from the recent work [3], which is nevertheless restricted to univariate polynomials. We use the notion of unisolvence which comes from the Finite Element Method (FEM): a unisolvent set of points \((x_r)_{1 \leq r \leq r_\ast}\) is such that any polynomial \(p \in \mathbb{P}^n[X]\) is uniquely determined by its values \(p(x_r) = y_r\) for \(1 \leq r \leq r_\ast\). This notion is convenient for multivariate polynomials.

**Model problem 1.1** (Iterative positive interpolation on \(\mathbb{K}\)). Take a unisolvent set \((x_r)_{1 \leq r \leq r_\ast}\), \(p \in \mathbb{P}^n_\mathbb{K} + [X]\) and consider the interpolated values \(y_r = p(x_r)\). From the sole knowledge of \((x_r)_{1 \leq r \leq r_\ast}\) and \((y_r)_{1 \leq r \leq r_\ast}\), compute iteratively some polynomials \((p_{ij})_{ij}\) such that the SOS representation (3) holds at the limit.

Using duality techniques from convex optimization [9], our main result solves the model problem. It is based on the minimization of a dual function \(G(\lambda)\) defined for a Lagrange multiplier \(\lambda\).

**Theorem 1.2.** There exists an explicitly computable function \(G : \mathbb{R}^{r_\ast} \to \mathbb{R} \cup \{+\infty\}\) depending only on interpolation points and on the point values \(y_r = p(x_r)\) with the properties:

1. The function \(G\) is a proper closed convex function on \(\mathbb{R}^{r_\ast}\). The function \(G\) is \(C^\infty\) on its non-empty open convex domain \(D = \{\lambda \in \mathbb{R}^{r_\ast} \text{ such that } G(\lambda) < \infty\}\) which depends only on the interpolation points.

2. Each \(\lambda \in D\) explicitly defines computable polynomials \((p_{ij}[\lambda])_{1 \leq i \leq i_\ast, 1 \leq j \leq j_\ast}\) such that

\[
\frac{\partial G}{\partial \lambda_r}(\lambda) = y_r - \sum_{j=1}^{j_\ast} g_j(x_r) \sum_{i=1}^{i_\ast} p_{ij}[\lambda](x_r), \quad 1 \leq r \leq r_\ast.
\]

Therefore if \(\lambda_\ast \in D\) is a critical point of \(G\), that is \(\nabla G(\lambda_\ast) = 0\), then the family \((p_{ij}[\lambda_\ast])_{ij}\) is solution to (3), that is is a SOS.
3. Assume that \( d = 1 \), \( \mathbb{K} \) is a segment and \( p \) is positive on \( \mathbb{K} \). Then \( G \) is strictly convex, coercive and admits a unique critical point in \( \mathcal{D} \). By descent algorithms, it yields a constructive iterative method to represent \( p \) as a SOS.

Despite the fact that the existence of the critical point is proved only for \( d = 1 \) in the univariate setting, our numerical results illustrate the efficiency of the descent method for multi-variate polynomials as well. The proof of each claim in Theorem 1.2 can be found as follows: the function \( G \) and its domain \( \mathcal{D} \) are introduced in Definition 2.9; the identity (4) is a combination of (20), (21) and the unisolvence; in the univariate setting \( d = 1 \), coercivity and strict convexity of \( G \) are shown in Theorem 4.4.

Near the boundary of its domain, the function \( G \) behaves very much like a logarithmic barrier function [16, 2] for interior point methods, but it is not for two reasons. The first one is that it is a rational function not a logarithm function (one can call it a rational barrier, see (18)), the second one is this barrier does not introduce any kind of approximation as explained in [2][page 564]. In our case the barrier is an exact one. This property is a strong algorithmic asset of \( G \) with respect to more standard logarithmic barrier methods.

The outline of this paper is as follows. In Section 2, we propose a dual interpretation of Problem 1.1. This leads us to the introduction of the function \( G \). Then, in Section 3, we discuss necessary and sufficient conditions characterizing asymptotic properties and strict convexity of \( G \). These conditions show the important role played by the Lagrange polynomials associated to the interpolation points \( (x_r)_{1 \leq r \leq r_s} \). In Section 4, we show that for univariate positive polynomials on a segment, the former conditions are satisfied yielding strict convexity and coercivity of the associated function \( G \). Besides we provide a more precise description the structure of the domain \( \mathcal{D} \). In Section 5, we present the specific descent and Newton type methods we use to compute the critical points of \( G \). Finally in Section 6 we provide numerical illustrations of the efficiency of our new approach both for computing SOS decomposition of polynomials in one variables on segments and two variables on triangles.

Acknowledgements. Both authors are greatly indebted to Jean-Bernard Lasserre and Didier Henrion for their kind explanations on the theory and state of the art of semi-definite programming and sum of squares and would like to thank them for their invitation at LAAS and for their hospitality. The authors would also like to acknowledge Simon Foucart, Swann Marx, Frédérique Charles, Martin Campos-Pinto and Teddy Pichard for interesting discussions during the conception of this work.

2 Dual reformulation

We begin by recasting the model problem 1.1 as the convex dual of a Quadratically Constrained Quadratic Program (QCQP). This approach is classical [2], however we detail the notations adapted to the interpolation procedure. Original material begins at Definition 2.6. In any dimension, the notation \( \langle \cdot, \cdot \rangle \) will denote the Euclidean dot product and \( \| \cdot \| \) will denote the associated norm.

We consider \( p_{ij} \in \mathbb{P}^n[X] \) with \( n_j = \left\lfloor \frac{n - \text{deg}(g_j)}{2} \right\rfloor \),

where \( \lfloor \cdot \rfloor \) denotes the integer part of a real number. In order to parametrize any polynomials one can use the canonical basis made of monomials \( X_1^{\alpha_1} \ldots X_d^{\alpha_d} \) for \( \alpha_i \in \mathbb{N} \). With the standard multi-
index notation \( \alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d \), \( |\alpha| = \alpha_1 + \cdots + \alpha_d \) and \( X^\alpha = X_1^{\alpha_1} \cdots X_d^{\alpha_d} \), any polynomial \( p_{ij} \in \mathbb{P}^n_j[X] \) is expressed as a linear combination

\[
p_{ij}(X) = \sum_{|\alpha| \leq n_j} c^{ij}_{\alpha} X^\alpha,
\]

for some line vector of coefficients \( c^{ij}_{\alpha} = (c^{ij}_{\alpha})_\alpha \in \mathbb{R}^{r_j} \) where \( r_j = \dim(\mathbb{P}^n_j[X]) = \binom{d+n_j}{d} \).

**Remark 2.1.** The monomials are convenient for the simplicity of the mathematical presentation. But we immediately mention that we use other bases in numerical experiments in Section 6 in order to optimize the robustness and accuracy of the algorithms. It only changes the definition of the matrices introduced hereafter, and every result of this paper still hold.

**Remark 2.2.** Consider the formula (3). The dimension of the polynomial space for \( p \) is \( r_* \). The number of degrees of freedom involved in the right hand side is \( i_* r_{**} \) with \( r_{**} = \sum_{j=1}^{j_*} r_j \). Since the problem is to construct SOS which correspond to a given polynomial \( p \), a reasonable assumption is to find the solution in a space with more degrees of freedom than the number of constraints. It means \( i_* r_{**} \geq r_* \). The even more optimistic hypothesis \( i_* = 1 \) and \( r_{**} = r_* \) is actually true in dimension \( d = 1 \), see the Lukács Theorem 4.1 (also called the Markov-Lukács Theorem in [11]). So the minimal hypothesis is the equality

\[
r_{**} = r_* \tag{6}
\]

which will be considered as true throughout this work (it can be modified as well, but it leads to heavier notations with little gain in terms of generality). Comments about other possibility for implementation will be made in the numerical section.

**Remark 2.3.** An interesting consequence of the Caratheodory Theorem ([9, Theorem III.1.3.6 page 98]) is that if a formula like (3) holds for \( i_* > r_* \), then a similar one holds also for \( i_* \equiv r_* \) (but for different polynomials \( p_{ij} \)). Indeed the set \( W = \sum_{j=1}^{j_*} g_j(X)\mathbb{P}^n_j[X]^2 \) is a closed convex set embedded in \( p \in \mathbb{P}^n[X] \). Therefore any convex combination of \( i_* > r_* \) elements of \( W \) can be expressed as a convex combination of only \( r_* = \dim(\mathbb{P}^n[X]) \) elements of \( W \) (the coefficients of the convex combination can be set to 1 after proper rescaling). Some of our results below like Lemma 2.8 will explicitly use the minimal value \( i_* = r_* \), however we prefer to use \( i_* \) as a free parameter independent of \( r_* \). The two main reasons are that, on the one hand Lemma 2.11 will enlighten a sharper estimate, and on the other hand the Lukács-Markov Theorem is even better since \( i_* = 1 \).

Then one gathers the coefficients \( c^{i1}, c^{i2}, \ldots, c^{i_*} \) in a single column vector

\[
U_i = (c^{i1}, c^{i2}, \ldots, c^{i_*})^t \in \mathbb{R}^{r_*} \quad \text{where} \quad r_* = \sum_{j=1}^{j_*} r_j. \tag{7}
\]

Set \( D^{n_j}(X) \in \mathbb{R}^{r_j \times r_j} \) the matrix with polynomial coefficients

\[
D^{n_j\alpha,\beta}(X) = X^\alpha X^\beta, \quad |\alpha|, |\beta| \leq n_j.
\]

Define \( B(X) \in \mathbb{R}^{r* \times r*} \) which is a polynomial valued block matrix

\[
B(X) = \text{diag}(g_1(X)D^{n_1}(X), g_2(X)D^{n_2}(X), \ldots, g_{j_*}(X)D^{n_{j_*}}(X)) \tag{8}
\]

This notation means that the first block on the diagonal is square \( r_1 \times r_1 \), the second block is square \( r_2 \times r_2 \), \ldots, until the last block which is square \( r_{j_*} \times r_{j_*} \); all other terms are zero.
Lemma 2.4. Let \( x \in K \). Then \( B(x) \) is symmetric and non-negative, that is \( B(x) = B(x)^t \geq 0 \).

Proof. It is a consequence of the identity

\[
\sum_{j=1}^{n} g_j(X) \sum_{i=1}^{n} p_{ij}^2(X) = \sum_{i=1}^{n} \left( \sum_{j=1}^{n} g_j(X) p_{ij}^2(X) \right) = \sum_{i=1}^{n} \langle B(X) U_i, U_i \rangle \tag{9}
\]

and the fact that \( g_j(x) \geq 0 \) for \( x \in K \).

We denote the evaluation at interpolation points is denoted as

\[
B_r = B(x_r) \in \mathbb{R}^{r* \times r*}. \tag{10}
\]

If \( x_r \in K \), the matrices \( B_r \) are symmetric non-negative. Finally, we define the algebraic manifold

\[
\mathcal{U} = \{ U = (U_1, \ldots, U_{r^*}) \in (\mathbb{R}^{r*})^{i*} \text{ such that } \sum_{i=1}^{i*} \langle B_r U_i, U_i \rangle = y_r \text{ for all } 1 \leq r \leq r^* \}. \tag{11}
\]

With the unisolvence assumption, finding a SOS (3) holds amounts to finding one element \( U \in \mathcal{U} \).

2.1 Lagrangian duality and definition of \( G \)

In order to find a \( U \in \mathcal{U} \) in a constructive manner, our strategy is to start at a given \( V \) (probably outside \( \mathcal{U} \)) and to project on \( \mathcal{U} \) in the quadratic norm. It writes as follows.

Problem 2.5. Given \( V = (V_1, \ldots, V_{i*}) \in (\mathbb{R}^{r*})^{i*} \)

\[
\text{minimize} \quad \frac{1}{2} \sum_{i=1}^{i*} \| U_i - V_i \|^2. \tag{12}
\]

The vectors \( V = (V_i)_i \) may be thought of as a good initial guesses for the \( U = (U_i)_i \) (if they exist of course, namely if \( \mathcal{U} \neq \emptyset \) which we do not know yet). The optimal value of the cost in problem (12) does not matter. But of course, at first glance, the optimization problem (12) seems even harder to solve than the original problem we were concerned with. Our findings is that the Lagrangian dual problem of (12) is endowed with good properties provided \( V \) is conveniently chosen. In this case, the new problem 2.5 provides a way to determine an admissible \( U \in \mathcal{U} \).

Still for any \( V \), introduce the Lagrangian which is the sum of the functional (12) and of the dualization of the constraint (11) with a Lagrange multiplier \( \lambda \in \mathbb{R}^{r*}

\[
\mathcal{L}(U, \lambda) = \frac{1}{2} \sum_{i=1}^{i*} \left( \| U_i - V_i \|^2 + \sum_{r=1}^{r*} \lambda_r \langle B_r U_i, U_i \rangle \right) - \frac{1}{2} \langle \lambda, y \rangle \quad \text{where } y = (y_r)_{1 \leq r \leq r^*}. \]

Both the objective and the constraints are quadratic, so the optimality constraints are linear. Define the symmetric matrix \( M(\lambda) = M(\lambda)^t \in \mathbb{R}^{r* \times r*} \)

\[
M(\lambda) = I + \sum_{r=1}^{r*} \lambda_r B_r \tag{13}
\]
where $I$ is the identity matrix in $\mathbb{R}^{r^* \times r^*}$. The first order optimality condition in the $U$ variable write
\begin{equation}
M(\lambda)U_i = V_i \quad \text{for} \quad 1 \leq i \leq i_s \iff M(\lambda)U = V. \tag{14}
\end{equation}
If the multiplier $\lambda \in \mathbb{R}^{r^*}$ is such that the matrix $M(\lambda)$ is invertible, then the candidate solution $U$ can be computed explicitly in terms of $\lambda$ and $V$ as the solution of the linear system (14).

It is therefore natural to concentrate on a condition on $\lambda$ such that $M(\lambda)$ is invertible. To obtain convexity properties in the following we even restrict $\lambda$ to the set of positive-definiteness of $M(\lambda)$, that we eliminate non singular matrices which have negative eigenvalues. To our knowledge, this at this stage that our analysis departures from the standard expository of dual QCQP [2, 9].

**Definition 2.6.** The domain of positive definiteness of $M$ is $D \subset \mathbb{R}^{r^*}$
\begin{equation}
D = \{ \lambda \in \mathbb{R}^{r^*} \text{ such that } M(\lambda) > 0 \}. \tag{15}
\end{equation}

It is an open set and it is non empty since $0 \in D$.

For a Lagrange multiplier $\lambda \in D$, the inverse transformation of (14) is
\[ U(\lambda) = M(\lambda)^{-1}V. \]
Then, one can evaluate the Lagrangian at $U(\lambda)$. An elementary calculation yields
\[ \mathcal{L}(U(\lambda), \lambda) = \frac{1}{2} \sum_{i=1}^{i_s} \left( \|V_i\|^2 - \langle V_i, M(\lambda)^{-1}V_i \rangle \right) - \frac{1}{2} \langle \lambda, y \rangle. \]
This motivates the introduction of the dual objective function $G_V : D \to \mathbb{R}$ defined by
\begin{equation}
G_V(\lambda) = \sum_{i=1}^{i_s} \langle V_i, M(\lambda)^{-1}V_i \rangle + \langle \lambda, y \rangle \tag{16}
\end{equation}
Two basic properties are the following.

**Lemma 2.7.** The function $G_V$ is smooth on $D$. The first and second derivatives are
\begin{equation}
\frac{\partial G_V}{\partial \lambda_r}(\lambda) = y_r - \sum_{i=1}^{i_s} \langle U_i(\lambda), B_r U_i(\lambda) \rangle, \tag{17}
\frac{\partial^2 G_V}{\partial \lambda_r \partial \lambda_s}(\lambda) = 2 \sum_{i=1}^{i_s} \langle B_r U_i(\lambda), M(\lambda)^{-1} B_s U_i(\lambda) \rangle.
\end{equation}

In particular $G_V$ is convex on its domain $D$.

**Proof.** The proof stems from the identity $\partial_\lambda M(\lambda)^{-1} = -M(\lambda)^{-1}B_r M(\lambda)^{-1}$ and the symmetry of the various matrices involved. Convexity follows from the positivity of $M(\lambda)$ and the expression of second derivatives yielding $\langle \nabla^2 G_V(\lambda), \mu \rangle = 2 \sum_{i=1}^{i_s} \langle A_i(\mu, \lambda), M(\lambda)^{-1} A_i(\mu, \lambda) \rangle \geq 0$ where $A_i(\mu, \lambda) = \sum_{i=1}^{i_s} \mu_r B_r U_i(\lambda)$ for $1 \leq i \leq i_s$. \qed
Figure 1: Graph of a convex function $f$ with domain $(0, \infty)$. On the left, the function is not closed because it has a finite limit at the boundary. On the right, the function is closed because it has an asymptote.

Now let us address the behavior of $G_V$ near the boundary, as illustrated in Figure 1. A convex function $f : \mathbb{R}^r \to \mathbb{R} \cup \{+\infty\}$ is said to be closed over its domain $D_f = \{x, f(x) < \infty\}$ if and only if the level sets $\{x, f(x) \leq t\}$ are closed for $t < +\infty$: see [9] or [2, Appendix A.3.3.]. This property is extremely important in our approach because it yields a strong control of the objective function at finite distance and it rules out situations like the one on the left part of Figure 1.

For a good choice of $V$ one can show that $G_V$ is infinite at $\partial D$ and is indeed a closed convex function.

**Lemma 2.8.** Take $i_* = r_*$ and $V \in \mathbb{R}^{r* \times r*}$ an orthogonal matrix. Then one has the simpler expression

$$G_V(\lambda) = \text{tr}(M^{-1}(\lambda)) + \langle \lambda, y \rangle.$$  \hspace{1cm} (18)

where $\text{tr}(\cdot)$ denotes the trace of a square matrix. Moreover the extension of $G_V$ with value $+\infty$ outside of $D$ is a closed convex function.

*Proof.* The formula is a direct consequence of (16). Thanks to the continuity on $D$, the closedness of $G_V$ on $\mathbb{R}^{r*}$ amounts to showing that for any sequence $(\mu_k)_k$ in $D$ converging to a point of the boundary of the domain

$$\partial D = \left\{ \lambda \in \mathcal{D}, \det \left( I + \sum_{r=1}^{r_*} \lambda_r B_r \right) = 0 \right\},$$

then one has $G_V(\mu_k) \to +\infty$ as $k \to +\infty$. In the light of the representation formula (18) involving the trace of $M(\lambda)^{-1}$ it is the case since the minimal eigenvalue of $M(\mu_k)$ goes to 0 as $k \to +\infty$. \(\square\)

From now on we consider only the closed convex extension. Since it is independent of the orthonormal basis $V$, one can take $V = I$ the identity matrix. A special notation is used for the dual function.

**Definition 2.9.** Denote by $G = G_I$ the closed convex function with domain $D$

$$G(\lambda) = \begin{cases} \text{tr}(M(\lambda)^{-1}) + \langle \lambda, y \rangle, & \text{if } \lambda \in D, \\ +\infty, & \text{otherwise.} \end{cases}$$  \hspace{1cm} (19)
The only parameters are the vector $y$ and the matrices $(B_r)_{r \in \{1, \ldots, r_*\}}$ through (10)-(13).

### 2.2 Critical points of $G$

In this section, we formalize natural consequences of the formulas (17) for the derivatives of $G$. These first properties are essentially a reformulation of the previous material. For each Lagrange multiplier $\lambda \in D$ one defines the vectors $(e_i^j[\lambda])_{\alpha,j} \in \mathbb{R}^r$ which are the components of $U_i(\lambda)$ (see (7) for details on the notations), the latter being the $i$th column of $M(\lambda)^{-1}$. With (5) and (7), it defines the polynomials $p_{ij}[\lambda] \in P^{n_j}[X]$

$$p_{ij}[\lambda](X) = \sum_{|\alpha| \leq n_j} e_{ij}^\alpha[\lambda] X^\alpha$$

With (9), these polynomials define a sum of squares $p[\lambda] \in P^n_+[X]$

$$p[\lambda](X) = \sum_{j=1}^{j_*} g_j(X) \left( \sum_{i=1}^{i_*} p_{ij}[\lambda](X) \right).$$

Using (9), $p[\lambda](x_r) = \sum_{i=1}^{i_*} \langle B_r U_i, U_i \rangle$. So (17) is rewritten as

$$\frac{\partial G}{\partial \lambda_r}(\lambda) = y_r - p[\lambda](x_r).$$

Proposition below characterizes that Problem 1.1 is equivalent to finding critical points of $G$.

**Proposition 2.10.** Take $p \in P^n_+[X]$ and an unisolvent set of interpolation points $(x_r)_{1 \leq r \leq r_*}$ in $K$. Consider $y_r = p(x_r)$ for $1 \leq r \leq r_*$. The following properties are equivalent

- $\lambda^* \in D$ is a critical point of $G$, namely $\nabla G(\lambda^*) = 0$.
- $\lambda^* \in D$ minimizes $G$.
- $p(X) = p[\lambda^*](X)$.

**Proof.** Since $G$ is closed convex, local minima coincide exactly with critical points, so the two first points are equivalent. The equivalence between the first and third assertions follows from (21) and the unisolvence assumption. □

### 2.3 Number of squares

Let us now precise the number of squares involved in the SOS formula (20). At first sight it seems indeed that each $\sum_{i=1}^{i_*} p_{ij}^2[\lambda](X)$ might involve $i_* = r_* = r_1 + \cdots + r_{j_*}$ different polynomials. Actually this is not the case.

**Lemma 2.11.** The number of non zero polynomials in $\sum_{i=1}^{i_*} p_{ij}^2[\lambda](X)$ is less or equal to $r_j$.

**Proof.** By construction $(U_1(\lambda), \ldots, U_{i_*}(\lambda)) = U(\lambda) = M(\lambda)^{-1}$ is a block diagonal matrix. The blocks have size $r_1 \times r_1$ until $r_{j_*} \times r_{j_*}$. So, for a given $j$, the polynomials $p_{ij}[X]$ vanish for $1 \leq i \leq r_1 + \cdots + r_{j_*}$ and for $r_1 + \cdots + r_{j_*} - 1 + 1 \leq i \leq i_*$. □
Remark 2.12. The result of Lemma 2.11 is nevertheless non optimal in dimension \( d = 1 \). Indeed consider the Lukács Theorem (see Proposition 4.1) in the odd case \( n = 2k + 1 \) and take \( g_1(X) = X \) and \( g_2(X) = (1 - X) \) as in (31). So \( r_+ = n \) and \( r_1 = r_2 = k \). Assume that there exists a critical point \( \lambda_* \) to \( G \). Then (20) yields a representation
\[
p(X) = X \sum_{i=1}^{k} p_{11}[\lambda_*](X) + (1 - X) \sum_{i=k+1}^{2k} p_{12}[\lambda_*](X).
\]
In terms of the number of squares, here \( 2k \), it is clearly non optimal with respect to the result of the Lukács Theorem which involves only two polynomials whatever \( n \).

In greater dimensions \( d \geq 1 \), the non optimality with respect to the literature [12, 13] is less evident so far. However, by comparison with the exponential bound of Theorem 2.16 in [12], one can notice the SOS formula (20) is endowed with a strong control of the degree and therefore with a strong control of the number of terms.

3 Coercivity of \( G \)

Now that the function convex \( G \) is constructed with good properties at the boundary \( \partial D \), we study the existence and uniqueness of critical points of the function \( G \). On the one hand, since the latter function is closed convex (so is is infinite at \( \partial D \)), a sufficient condition for the existence of a critical point is coercivity, namely if \( G \) is infinite at infinity
\[
\lim_{|\lambda| \to +\infty} G(\lambda) = +\infty.
\]
On the other hand a sufficient condition for the uniqueness of the critical points is strict convexity.

In the following, we start in Section 3.1 by investigating the asymptotic behavior of \( G \) along rays starting at 0. From this knowledge we derive conditions characterizing coercivity in Section 3.2. Finally we shall characterize strict convexity in Section 3.3.

3.1 Lower boundedness in the asymptotic cone

There are two types of directions in \( D \). For \( d \in \mathbb{R}^* \) with \( |d| = 1 \), one defines the rays \( R_d := \{td, t \geq 0\} \) issued from the starting point 0 \( \in R_d \). Two possibility occur: either \( R_d \) intersects the boundary \( \partial D \) either it does not. In the first case \( \sup_{t \geq 0} \{ t : G(td) < +\infty \} < +\infty \), and we know already that the function \( G \) is bounded from below and coercive along \( R_d \): that is if one notes \( t_d > 0 \) the unique real number such that \( td \in \partial D \), then \( \lim_{t \to t_d^{-}} G(td) = +\infty \).

In this section one is interested in the rest of the directions. They generate the so-called asymptotic cone or recession cone of \( D \). The asymptotic cone is closed, independent of the starting point and is classically defined [9] by \( C_\infty = \{\lambda \in \mathbb{R}^* \text{ such that } \forall \mu \in D, t \geq 0, \mu + t\lambda \in D\} \).

Lemma 3.1. The asymptotic cone of \( D \) is given by
\[
C_\infty = \left\{ \lambda \in \mathbb{R}^*, \sum_{r=1}^{r_n} \lambda_r B_r \geq 0 \right\}.
\]
Proof. Let \( \lambda, \mu \) such that \( \sum_{r=1}^{r_{\ast}} \lambda_r B_r \geq 0 \) and \( I + \sum_{r=1}^{r_{\ast}} \mu_r B_r > 0 \). Then, \( I + \sum_{r=1}^{r_{\ast}} (\mu_r + t \lambda_r) B_r > 0 \) for all \( t \geq 0 \), so \( \lambda \) belongs to the asymptotic cone. Conversely let \( \lambda \) such that for all \( \mu \) and \( t \geq 0 \), \( \mu + t \lambda \in \mathcal{D} \). If \( \sum_{r=1}^{r_{\ast}} \lambda_r B_r \) had a negative eigenvalue then for \( t \) large enough \( I + t \sum_{r=1}^{r_{\ast}} \lambda_r B_r \) would also have a negative eigenvalue which would contradict the fact that \( t \lambda \in \mathcal{D} \).

The main question in the current and next section is the asymptotic behavior of \( G \) in directions in \( \mathcal{C}_{\mathcal{K}} \). Several situations are illustrated in Figure 2.

Figure 2: Graph of a closed convex function \( f \) which is convex over its domain \( (0, \infty) \) and asymptotically linear at infinity, that is \( C_{\mathcal{K}}^f = [0, \infty) \). On the left, the function is not lower bounded and not coercive. In the center the function is lower bounded but not coercive. On the right, the function is lower bounded and coercive.

Let us derive a condition which rules out the behavior on the left, namely unboundedness from below in the asymptotic cone. One introduces the polynomial valued vector \( L(X) \) with components being the Lagrange polynomials associated with the set of points \( (x_r)_{1 \leq r \leq r_{\ast}} \) evaluated at \( x \), namely

\[
L(X) = (l_r(X))_{1 \leq r \leq r_{\ast}} \in \mathbb{R}^{r_{\ast}},
\]

where the Lagrange interpolation polynomials \( l_r \in \mathbb{P}^n[X] \) are defined by \( l_r(x_s) = \delta_{rs} \) for \( 1 \leq r, s \leq r_{\ast} \), where \( \delta_{rs} \) denotes the Kronecker symbol. The vector \( L(X) \) will be called a Lagrange vector. The polynomial \( p \) which takes the value \( y_r \) at \( x_r \) satisfies the Lagrange interpolation formula

\[
p(X) = \sum_{r=1}^{r_{\ast}} y_r l_r(X) = \langle y, L(X) \rangle.
\]

One can show another interpolation property characteristics of our problem.

**Lemma 3.2.** One has

\[
B(X) = \sum_{r=1}^{r_{\ast}} l_r(X) B_r
\]

In particular, for all \( x \in \mathcal{K} \), \( B(x) \) is non-negative and

\[
L(x) \in C_{\mathcal{K}}.
\]
Proof. Let $W, Z \in \mathbb{R}^r*$ which are, as in (7), the coefficients of some polynomials $(p_j)_{1 \leq j \leq j_*}$ and $(q_j)_{1 \leq j \leq j_*}$. By definition (8-9) of $B(x)$ which is symmetric one knows that

$$\langle W, \left( B(x) - \sum_{r=1}^{r_*} l_r(x)B_r \right) Z \rangle = \sum_{j=1}^{j_*} \left( g_j(x)p_j(x)q_j(x) - \sum_{r=1}^{r_*} l_r(x)g_j(x)r_j(x)p_j(x)q_j(x) \right) = 0.$$ 

Since $W, Z$ are arbitrary, it yields (26). Also for $x \in \mathbb{R}$, one has that $g_j(x) \geq 0$. Therefore

$$\langle W, B(x)W \rangle = \sum_{j=1}^{j_*} g_j(x)p_j(x)^2 \geq 0$$

which yields that $B(x) \geq 0$. One gets the condition (23), so $L(x) \in C_{\infty}$.

One can state a first negative result on the lower boundedness of $G$.

**Lemma 3.3.** Assume there exists $z \in \mathbb{K}$ such that $p(z) < 0$. Then the corresponding function $G$ is not bounded from below in $C_{\infty}$ since

$$\lim_{t \to +\infty} G(tL(z)) = -\infty.$$ 

Proof. The half line generated by $L(z)$ is included in $D$ by Lemma 3.2 and so all for $t \geq 0$, one has $G(tL(z)) = \text{tr} (M(tL(z))^{-1}) + tp(z)$. Since $\lambda = tL(z) \in C_{\infty}$, one has $M(\lambda) \geq I$ so

$$G(t\lambda) \leq r_* + tp(z) \xrightarrow{t \to +\infty} -\infty$$

Proposition 3.4. Consider $p \in P^n_{\mathbb{K}, x}[X]$, a unisolvent set of interpolation points $(x_r)_{1 \leq r \leq r_*}$ in $\mathbb{K}$ and define $y_r = p(x_r)$ for $1 \leq r \leq r_*$. The following properties are equivalent.

- For any $\lambda \in C_{\infty}$, one has $\langle \lambda, y \rangle \geq 0$.
- There exists polynomials $p_{ij}$ for $1 \leq j \leq j_*$ and $1 \leq i \leq i_*$ such that

$$p(X) = \sum_{j=1}^{j_*} g_j(X) \sum_{i=1}^{i_*} p_{ij}^2(X).$$

Proof. For $W \in \mathbb{R}^{r*}$, define the vector $s_W = \langle (B_r W, W) \rangle_{1 \leq r \leq r_*} \in \mathbb{R}^{r*}$. A definition of $C_{\infty}$ equivalent to (23) is

$$C_{\infty} = \{ \lambda \in \mathbb{R}^{r*} \text{ such that } \langle s_W, \lambda \rangle \geq 0 \text{ for all } W \in \mathbb{R}^{r*} \}.$$ 

To prove the result, one can invoke the Generalized Farkas Theorem ([9, Theorem III.4.3.4 page 131] with the correspondence $y = b$). It already states that our first assertion is equivalent to $y$ being in the closed convex conical hull of the linear forms $s_W$, that is $y = \sum_{i=1}^{i_*} \alpha_i s_{W_i}$, where $\alpha_i \geq 0$ for all $i$, and $i_*$ is sufficiently large. It is rewritten as $y = \sum_{i=1}^{i_*} s_{Z_i}$ for $Z_i = (\alpha_i)^{\frac{1}{2}} W_i$. Following Remark 2.3, one can take $i_* = r_*$. Using (11)-(9), the latter rewrites as our second assertion. \qed
3.2 Coercivity

Now we investigate the conditions yielding coercivity \((22)\) of \(G\). It corresponds to \(G\) being infinite at infinity, as in the right part of Figure 2 and formally to \((22)\). A first negative result about coercivity is the following.

Lemma 3.5. Assume there exists \(z \in K\) such that \(p(z) = 0\). Then \(G\) is not coercive since \(G(tL(z))\) remains bounded as \(t \to +\infty\).

Proof. It can be easily adapted from the proof of Lemma 3.3.

Thus we can only hope for coercivity starting from strictly positive polynomials. Let us know define a specific useful polynomial denoted as \(p_B\).

Definition 3.6. Define the polynomial \(p_B(X) = \text{tr}(B(X)) \in P^n_{K,+}[X]\), where \(B(X)\) is the matrix defined in \((8)\).

A key property of this polynomial is the following.

Lemma 3.7. Assume that the matrices \(\{B_r\}_{1 \leq r \leq r_*}\) are linearly independent. Then there exists a constant \(c_* > 0\) such that

\[
c_*\|\lambda\| \leq \sum_{r=1}^{r_*} \lambda_r p_B(x_r), \quad \forall \lambda \in C_\infty.
\]

Proof. Let \(\lambda \in C_\infty\). The matrix \(\sum_{r=1}^{r_*} \lambda_r B_r\) is symmetric and non-negative. So its matrix norm can be controlled by its largest eigenvalue and thus by its trace, namely

\[
\left\| \sum_{r=1}^{r_*} \lambda_r B_r \right\| \leq \text{tr} \left( \sum_{r=1}^{r_*} \lambda_r B_r \right) = \sum_{r=1}^{r_*} \lambda_r p_B(x_r).
\]

Second we also know that \(\lambda \to \sum_{r=1}^{r_*} \lambda_r B_r\) is injective thanks to the linear independence assumption. Thus there a constant \(c_* > 0\) such that \(c_*\|\lambda\| \leq \|\sum_{r=1}^{r_*} \lambda_r B_r\|\). Combining both inequalities ends the proof.

The result of the next Proposition holds in any dimension. It shows that any SOS can be approximated with our approach, provided one shifts positively the SOS by \(\varepsilon p_B\) where \(\varepsilon > 0\) is a small as required.

Proposition 3.8. Let \(p \in P^n_{K,+}[X]\) which admits a SOS \((3)\). Take a unisolvent set of interpolation points \((x_r)_{1 \leq r \leq r_*}\) in \(K\) and assume that the corresponding matrices \(\{B_r\}_{1 \leq r \leq r_*}\) are linearly independent. Take \(\varepsilon > 0\) and set \(p^\varepsilon = p + \varepsilon p_B\). Then the function \(G^\varepsilon\) built from \(x_r\) and \(y_r^\varepsilon = p^\varepsilon(x_r) = y_r + \varepsilon p_B(x_r)\) for \(1 \leq r \leq r_*\) is coercive.

Proof. The asymptotic cone \(C_\infty\) does not depend on \(y\) or \(y^\varepsilon\) and we desire to show firstly that \(G^\varepsilon\) grows linearly to infinity for directions in \(C_\infty\). One has the identity

\[
\sum_{r=1}^{r_*} \lambda_r y_r^\varepsilon = \sum_{r=1}^{r_*} \lambda_r y_r + \varepsilon \sum_{r=1}^{r_*} \lambda_r p_B(x_r).
\]
Take $\lambda \in C_\infty$: proposition 3.4 yields $\sum_{r=1}^{r_*} \lambda_r y_r \geq 0$ because $p$ is a SOS by assumption; then Lemma 3.7 shows that for any $\lambda \in C_\infty$, $\sum_{r=1}^{r_*} \lambda_r y_r \geq 0 + \varepsilon c_4 |\lambda|$ which yields uniform coercivity in the directions in the asymptotic cone.

To show coercivity (22) which is a stronger statement, the proof is by contradiction. Assume it does not hold. Then there exists a constant $\lambda \in \mathbb{R}$ as well as a sequence $(t_m, d_m)_{m \in \mathbb{N}}$ such that $t_m \to +\infty$, $|d_m| = 1$ and $G_{\text{univ}}(t_m d_m) \leq K$. By convexity, and since $G(0) = r_*$, one has $G(t d_m) \leq \max(r_*, K)$ for $t \in (0, t_m]$. Up to the extraction of a sub-sequence there exists $d_* \in \mathbb{R}^n$ such that $G(t d_*) \leq \max(r_*, K)$ for $t \in \mathbb{R}^+$. In particular the ray with direction $d_*$ cannot intersect the boundary $\partial D$ so it belongs to the asymptotic cone $C_\infty$. By the first estimate $G(t d_*) \geq \varepsilon c_4 t$, so it cannot be bounded which yields the contradiction. □

**Remark 3.9.** With the same strategy of proof, it is possible to show that any polynomial in a neighborhood of $p_u$, that is, $\{ q \in \mathbb{P}^{n}[X], |q - p_u| < \varepsilon \}$, generates a function $G$ which is coercive. It yields that all polynomials in this neighborhood admits a representation as a SOS.

### 3.3 Strict convexity

Strict convexity, if it holds, yields uniqueness of the critical point (if it exists).

**Proposition 3.10.** Let $p \in \mathbb{P}_{\mathbb{K}, \infty}[X]$ be strictly positive on $\mathbb{K}$. Take a unisolvent set of interpolation points $(x_r)_{1 \leq r \leq r_*}$ in $\mathbb{K}$ and assume that the corresponding matrices $(B_r)_{1 \leq r \leq r_*}$ are linearly independent. Then $G$ is strictly convex.

**Proof.** From (17) the Hessian $\nabla^2 G$ of $G$ is such that for all $\mu \in \mathbb{R}^{r_*}$

$$\langle \nabla^2 G(\lambda) \mu, \mu \rangle = 2 \sum_{i=1}^{i_*} \langle A_i(\mu, \lambda), M(\lambda)^{-1} A_i(\mu, \lambda) \rangle \geq 0$$

where $A_i(\mu, \lambda) = (\sum_{r=1}^{r_*} \mu_r B_r) U_i(\lambda)$ for $1 \leq i \leq i_*$. Since $M(\lambda)^{-1}$ is positive definite, its columns $U_i(\lambda)$ form a basis.

By contradiction, assume $G$ is not strictly convex. There exists $\mu \neq 0$ such that $\langle \nabla^2 G(\lambda) \mu, \mu \rangle = 0$. So the vectors $A_i(\mu, \lambda)$ vanish for all $i$. So $\sum_{r=1}^{r_*} \mu_r B_r = 0$, and $\mu = 0$ by linear independence of the matrices $(B_r)_{r=1,...,r_*}$. This is a contradiction so $\nabla^2 G(\lambda) > 0$ and $G$ is strictly convex. □

The strict convexity of $G$ can be measured with the minimal eigenvalue of its Hessian

$$\alpha(\lambda) = \inf_{\mu \neq 0} \frac{\langle \nabla^2 G(\lambda) \mu, \mu \rangle}{\| \mu \|^2} > 0,$$

for any $\lambda \in D$. An important property which motivates the design of one of our numerical methods is the following.

**Lemma 3.11.** Under the assumptions of Proposition 3.10, then $\alpha$ has a cubic degeneracy at infinity in the interior of the asymptotic cone of $D$. For all $d \in \mathbb{R}^{r_*}$ such that $\|d\| = 1$ and $\sum_{r=1}^{r_*} d_r B_r > 0$, there is $C_d > 0$ such that for all $t \geq 0$

$$\alpha(t d) \leq C_d (1 + t)^{-3}.$$
Proof. Let $\lambda = td$. Then for some constant $C$ depending only on the data one has
\[ \langle \nabla^2 G(\lambda) \mu, \mu \rangle \leq C \| M(\lambda)^{-1} \| \| \mu \|^2. \]
Then just note that under the assumptions the minimal eigenvalue of $M(\lambda)$ is given by $1 + e_d t$ with $e_d$ the minimal eigenvalue of $\sum_{r=1}^{\mathfrak{r}_*} d_r B_r$. Hence $\| M(\lambda)^{-1} \|$ behaves like $O((1 + t)^{-1})$. \hfill \Box

4 Univariate polynomials on a segment

In this section, we focus on univariate polynomials, namely when $d = 1$, over the segment $\mathbb{K} = [0, 1]$. This case is interesting because one can easily prove the coercivity and the strict convexity. And also a full description of the asymptotic cone is available. The notation is simplified by using the real variable $x \in \mathbb{R}$, more adapted to the analytical methods and results in Section 4.2.

4.1 The function $G$ for univariate polynomials

We check that the various assumptions granting coercivity and strict convexity are satisfied. In view of Proposition 3.4, Proposition 3.8 and Proposition 3.10 of the previous section, it suffices to exhibit an appropriate choice of functions $(g_j)_r$ and of interpolation points such that

- Any non-negative polynomial admits a (possibly non-explicit) sum of squares decomposition;
- The matrices $(B_r)_r$ are linearly independent.

The first point follows from the Markov-Lukács Theorem, see [20, 6, 5, 11] for a proof.

Proposition 4.1 (Markov-Lukács). Let us consider $p \in \mathbb{P}^n[x]$ and $\mathbb{K} = [0, 1]$.

- **Even case:** If $n = 2k$, then $p$ is non-negative on $\mathbb{K}$ if and only if there are polynomials $a$ and $b$ with degree less or equal to $k$ and $k - 1$ respectively such that

  \[ p(x) = a^2(x) + x(1 - x)b^2(x). \]  

  \[ (29) \]

- **Odd case:** If $n = 2k + 1$, then $p$ is non-negative on $\mathbb{K}$ if and only if there are polynomials $a$ and $b$ with degree less or equal to $k$ such that

  \[ p(x) = xa^2(x) + (1 - x)b^2(x). \]  

  \[ (30) \]

Now let us precise the setting. One takes $j_* = 2$ and

\[ \left\{ \begin{array}{ll}
  \text{for } n \text{ is even : } & g_1(x) = 1 \quad \text{and} \quad g_2(x) = x(1 - x), \\
  \text{for } n \text{ is odd : } & g_1(x) = x \quad \text{and} \quad g_2(x) = 1 - x.
\end{array} \right. \]  

\[ (31) \]

Concerning the interpolation points, we choose any $r_* = n + 1$ distinct points $(x_r)_{r=1,...,n+1}$ on the segment $[0, 1]$. The polynomials are represented along monomials so that the matrices $B_r$ have the block structure

\[ B_r = \left( \begin{array}{cc}
  g_1(x_r) w_1^r \otimes w_1^r & 0 \\
  0 & g_2(x_r) w_2^r \otimes w_2^r
\end{array} \right) \in \mathbb{R}^{(n+1) \times (n+1)} \]  

\[ (32) \]
Proof. Assume the matrices Lemma 4.3. The consequence is the following.

For $n = 2k$, \( \mathbf{w}_1^* = (1, x_r, \ldots, x^k)^t \) and \( \mathbf{w}_2^* = (1, x_r, \ldots, x^{k-1})^t \).

For $n = 2k + 1$, \( \mathbf{w}_1^* = \mathbf{w}_2^* = (1, x_r, \ldots, x^k)^t \).

With these notations, the equalities (29) and (30) are equivalent to $y_r = \langle B_r, \mathbf{U}, \mathbf{U} \rangle$ for $1 \leq r \leq n+1$. In the odd case $n = 2k + 1$ one has \( \mathbf{U} = (a_0, \ldots, a_k, b_0, \ldots, b_k)^t \in \mathbb{R}^{n+1} \) with \( a(x) = \sum_{i=0}^{k} a_i x^i \) and \( b(x) = \sum_{i=0}^{k} b_i x^i \). In the even case $n = 2k$, \( \mathbf{U} = (a_0, \ldots, a_k, b_0, \ldots, b_{k-1})^t \in \mathbb{R}^{n+1} \).

**Corollary 4.2** (of Proposition 3.4). Take $p \in P^n_{[0,1]}$ and set $y_r = p(x_r)$. Then, for all $\lambda \in C_x$, one has that $\langle \lambda, y \rangle \geq 0$.

*Proof.* Indeed the second statement of Proposition 3.4 holds with $i_* = 1$ by taking $p_{11} = a$ and $p_{12} = b$ with $a, b$ provided by Proposition 4.1.

Let $\lambda \in \mathbb{R}^{n+1}$. Using the structure (32) of the matrices $B_r$, one has the Hankel matrices

\[
\sum_{r=1}^{n+1} \lambda_r B_r = \begin{pmatrix} H_1 & 0 \\ 0 & H_2 \end{pmatrix}
\]  

(33)

where

\[
\begin{align*}
\text{for } n = 2k & : & \langle H_1 \mathbf{v}, \mathbf{w} \rangle &= \sum_{i,j=0}^{k} s_{i+j+1} v_i w_j, & \langle H_2 \mathbf{v}, \mathbf{w} \rangle &= \sum_{i,j=0}^{k} (s_{i+j} - s_{i+j+1}) v_i w_j, \\
\text{for } n = 2k + 1 & : & \langle H_1 \mathbf{v}, \mathbf{w} \rangle &= \sum_{i,j=0}^{k} s_{i+j+1} v_i w_j, & \langle H_2 \mathbf{v}, \mathbf{w} \rangle &= \sum_{i,j=0}^{k-1} (s_{i+j+1} - s_{i+j+2}) v_i w_j.
\end{align*}
\]

The $s_i$'s are given by $s_i = \sum_{r=1}^{n+1} \lambda_r x^i_r$. The linear map $\lambda \mapsto (s_0, \ldots, s_n)$ is one to one, since $(s_0, \ldots, s_n)$ is obtained by multiplying $\lambda$ by a Vandermonde matrix, which is invertible. A direct consequence is the following.

**Lemma 4.3.** The matrices $\{B_r\}_{1 \leq r \leq r_*}$ are linearly independent.

*Proof.* Assume $\sum_{r=0}^{n} \lambda_r B_r = 0$. Then (33) and the definition of $H_1$ and $H_2$ yields that $s_0 = \cdots = s_n = 0$. It yields $\lambda = 0$. So the $\{B_r\}_{1 \leq r \leq r_*}$ are linearly independent.

**Theorem 4.4.** For any univariate polynomial $p$ that is strictly positive on $\mathbb{K} = [0, 1]$, the associated function $G$ is strictly convex and coercive. As a consequence, it has a unique critical point $\lambda^*$ which defines a sum of squares decomposition $p[\lambda^*] = p$.

*Proof.* Thanks to Corollary 4.2 and Lemma 4.3, the assumptions of Proposition 3.8 and Proposition 3.10 are satisfied which yields the result.

### 4.2 The asymptotic cone for univariate polynomials

Given a subset $S \subset \mathbb{R}^{n+1}$ we denote by $\text{con}(S)$ the *conical hull* of $S$ that is the set of linear combinations with non-negative coefficients of elements of $S$. In this section the asymptotic cone $\mathcal{D}$ (23) is constructed from the matrices (32) or (33) in the univariate case. The main result is the following.
Theorem 4.5. The asymptotic cone of $\mathcal{D}$ is generated by the Lagrange vectors $L(x)$ for $0 \leq x \leq 1$

$C_\infty = \text{coni} \{ L(x) \in \mathbb{R}^{n+1}, x \in [0, 1] \}$.

We need some intermediate results in order to prove Theorem 4.5. First, let us define

$$C_1^1 = \{ \lambda \in C_\infty, \sum_{r=1}^{n+1} \lambda_r = 1 \} \subset C_\infty.$$  \hspace{1cm} (34)

Since $\sum_{r=1}^{n+1} l_r(x) = 1$ for all $1 \leq r \leq r_*$, one has $\sum_{r=1}^{n+1} l_r(X) = 1$. Therefore, with Lemma 3.2, we know that $\{ L(x) \in \mathbb{R}^{n+1}, x \in [0, 1] \} \subset C_1^1$. The main point of the proof is to show that $C_1^1 \subset \{ L(x) \in \mathbb{R}^{n+1}, x \in [0, 1] \}$.

To do so we identify $C_1^1$ with a subset of Borel probability measures on $[0, 1]$ using the theory of the moment problem for which an comprehensive reference is [11]. The proof of the Theorem invoked below in the proof is strongly related to the Lukacs decomposition of Theorem 4.1.

Proposition 4.6. Let $\lambda \in \mathbb{R}^{n+1}$. The following are equivalents

- The vector $\lambda$ belongs to $C_1^1$.
- There is a Borel probability measure $\sigma$ on $[0, 1]$ such that

$$\sum_{r=1}^{n+1} \lambda_r B_r = \int_{[0,1]} B(x) d\sigma(x). \hspace{1cm} (35)$$

Proof. Using (33), one can say that $\lambda \in C_2^1 \iff (s_0, \ldots, s_n)$ are such that $H_1$ and $H_2$ are non-negative matrices and $s_0 = 1$. By [11, Theorem 2.3, Theorem 2.4], this is equivalent to the existence of a Borel probability measure $\sigma$ such that (35) holds.

Corollary 4.7. The set $C_1^1$ is compact.

Proof. Since, by Proposition 4.6, the $s_i$’s are moments of a Borel probability measure on $[0, 1]$, one has $(s_0, \ldots, s_n) \in [0, 1]^{n+1}$. Therefore, since $\lambda \mapsto (s_0, \ldots, s_n)$ is linear and invertible (see Lemma 4.3), $C_1^1$ is bounded.

We recall that a point $\lambda$ of a convex set $C$ is said to be an extreme point (see [9, III, Definition 2.3.1]) of $C$ if for any $\lambda_1, \lambda_2 \in C$ such that $\lambda = (\lambda_1 + \lambda_2)/2$, one has $\lambda_1 = \lambda_2$. We denote by $\text{ext}(C)$ the set of extreme points of $C$.

Proposition 4.8. The set of extreme points of $C_1^1$ is given by

$$\text{ext}(C_1^1) = \{ L(x), x \in [0, 1] \}.$$ 

Proof. Let $\lambda \in \text{ext}(C_2^1)$. Since extreme points of a convex set are located on its boundary there is a vector $V \neq 0$ such that $\langle \sum_{r=1}^{n+1} \lambda_r B_r V, V \rangle = 0$. Let $\sigma$ be a Borel measure satisfying (35) and define $q(X) = \langle B(X)V, V \rangle \geq 0$. One has

$$\int_{[0,1]} q(x) d\sigma(x) = 0.$$
Since \( q \) is not identically zero, the measure \( \sigma \) must be supported on a subset of the finite set of roots of \( q \) intersected with \([0,1]\). Since \( q \) has degree \( n \), \( \sigma \) has the form

\[
\sigma = \sum_{k=1}^{n} \alpha_k \delta_{x_k}, \quad \sum_{k=1}^{n} \alpha_k = 1, \quad 0 \leq \alpha_k \leq 1, \quad x_k \in [0,1],
\]

for some distinct \( x_1, \ldots, x_n \) and where \( \delta_{x_k} \) is the Dirac measure at \( x_k \). Now assume that for some index \( k \), \( \alpha_k \in (0,1) \). Then there is \( k' \neq k \) such that \( \alpha_{k'} \in (0,1) \). Then let \( 0 < \varepsilon < \min(\alpha_k, \alpha_{k'}, 1 - \alpha_k, 1 - \alpha_{k'}) \) and define \( \sigma_1 = \sigma - \varepsilon \delta_{x_k} + \varepsilon \delta_{x_{k'}} \) and \( \sigma_2 = \sigma + \varepsilon \delta_{x_k} - \varepsilon \delta_{x_{k'}} \). The measures \( \sigma_1 \) and \( \sigma_2 \) are two Borel probability measures generating different sets of moments for at least some \( \varepsilon \) in the range. Since \( \lambda \mapsto (s_0, \ldots, s_n) \) is linear and invertible there are distinct \( \lambda_1, \lambda_2 \in C^1_\mathbb{X} \) satisfying (35) for the respective measures \( \sigma_1 \) and \( \sigma_2 \) and one has \( \lambda = (\lambda_1 + \lambda_2)/2 \).

There is a contradiction. Therefore either \( \alpha_k = 0 \) or \( \alpha_k = 1 \) so \( \sigma \) must be a dirac measure at some point \( x_\ast \in [0,1] \). Hence \( \sum_{r=1}^{n+1} \lambda_r B(x_r) = B(x_\ast) \) so in particular

\[
\sum_{r=1}^{n+1} \lambda_r x_r^k = x_\ast^k \quad \text{for any } 0 \leq k \leq n
\]

which yields \( \lambda = L(x_\ast) \).

Conversely if \( \lambda = L(x_\ast) \) and \( \lambda = (\lambda_1 + \lambda_2)/2 \), then there are probability measures \( \sigma_1 \) and \( \sigma_2 \) such that \( \delta_{x_\ast} = (\sigma_1 + \sigma_2)/2 \). Therefore \( \sigma_1 \) and \( \sigma_2 \) are supported at \( x_\ast \) and since they have the same mass one has \( \delta_{x_\ast} = \sigma_1 = \sigma_2 \), so \( \lambda \in \text{ext}(C^1_\mathbb{X}) \).

Now we can prove the main result of the section.

**Proof of Theorem 4.5.** We denote by \( \text{co}(S) \) the convex hull of \( S \) that is the set of linear combinations of elements of \( S \) with non-negative coefficients whose sum equals 1. By the Minkowski (or Krein- Milman) theorem [9, III, Theorem 2.3.4] we know that any compact convex set is the convex hull of its extreme points, therefore \( C^1_\mathbb{X} = \text{co}(\text{ext}(C^1_\mathbb{X})) \). Then we remark that \( C_\mathbb{X} = \bigcup_{t \geq 0} t C^1_\mathbb{X} \) and the result follows.


## 5 Numerical algorithms

Now we introduce several numerical methods employed to compute sum of squares decompositions. They are based on the minimization of the dual function \( G \) either by a descent type algorithm, either by the direct search of a critical point with a Newton type methods. All the methods enter the generic iterative framework

\[
\lambda_{m+1} = \lambda_m - \tau_m H_m^{-1} \nabla G(\lambda_m), \quad \lambda_0 = 0,
\]

with \( H_m \) and \( \tau_m \) to be defined. The latter is an adaptive time step ensuring the decay of \( \|\nabla G(\lambda_m)\| \) at each step. We recall that this quantity actually measures the euclidean norm between the current sum of squares \( \langle p[\lambda_m](x_r), y \rangle \) and \( y \). The adaptive time step \( \tau_m \) is defined as follows. Let us define \( \lambda_m^{(k)} = \lambda_m - 2^{-k} \tau_m^{(k)} H_m^{-1} \nabla G(\lambda_m) \). Then we denote by \( k_m \) the smallest integer such that \( \|\nabla G(\lambda_m^{(k)})\| < \|\nabla G(\lambda_{m+1})\| \). From there we define

\[
\tau_{m+1} = \begin{cases} 
2^{-k_m} \tau_m & \text{if } k_m > 0, \\
2 \tau_m & \text{if } k_m = 0.
\end{cases}
\]

17
These algorithms can be understood as the discretization of a convenient gradient flow ordinary differential equation ODE.

5.1 An ODE

We exhibit an simple ODE which builds up a sum of squares decomposition of a positive polynomial \( p \) at the limit of large time.

**Proposition 5.1.** Let \( p \) be a positive polynomial on \([0, 1]\) and let \( G \) be the associated dual function. Consider the gradient flow

\[
\begin{align*}
\frac{d\lambda}{dt}(t) &= -\nabla G(\lambda(t)), \\
\lambda(0) &= 0 \in \mathcal{D}.
\end{align*}
\]

Then \( \lambda(t) \) converges to a critical point \( \lambda^* \in \mathcal{D} \), and \( \lim_{t \to \infty} p[\lambda(t)] = p[\lambda^*] = p \).

**Proof.** First \( G \) decays along the trajectory since

\[
\frac{d}{dt} G(\lambda(t)) = -\|\nabla G(\lambda(t))\|^2 \leq 0.
\]

Since \( G \) is coercive and smooth, (37) has a unique global solution \( \lambda(t) \in \mathcal{D} \) for all \( t \geq 0 \). Since \( G \) is bounded from below \( G(\lambda(t)) \) has a limit when \( t \to \infty \). Then \( t \to G(\lambda(t)) \) is convex along the flow since

\[
\frac{d^2}{dt^2} G(\lambda(t)) = 2 \left\langle \nabla G(\lambda(t)), \nabla^2 G(\lambda(t)) \nabla G(\lambda(t)) \right\rangle \geq 0.
\]

Therefore, by convexity, one has \( \frac{d}{dt} G(\lambda(t)) \to 0 \) which turns into \( \lim_{t \to \infty} \|\nabla G(\lambda(t))\| = 0 \). Since \( G \) has a unique critical point \( \lambda^* \), \( \lambda(t) \to \lambda^* \) when \( t \to \infty \).

5.2 Descent methods

Descent methods (36) can be seen as a discretization in time of the gradient flow (37).

**Forward descent method**

The first method we use is the classical descent method which consists in taking

\[
H_m = I,
\]

where \( I \) is the identity matrix.

**Backward descent method**

This second numerical method is based on the gradient flow defined in Proposition 5.1. Given a sequence of positive time steps \( \tau_m \), the following iterative scheme

\[
\tilde{\lambda}_{m+1} = \arg \min_{\lambda \in \mathcal{D}} G(\lambda) + \frac{1}{2\tau_m} \|\lambda - \tilde{\lambda}_m\|^2, \quad \tilde{\lambda}_0 = 0.
\]

is well defined since \( G \) is convex. It corresponds exactly to the implicit Euler discretization of the gradient flow with variable time steps. At step \( m \) we look for the critical point of the strictly convex objective function by making one step of a Newton method starting at \( \lambda_m \), yielding the scheme (36) with

\[
H_m = I + \tau_m \nabla^2 G(\lambda_m).
\]

The adaptive time step is chosen as in Section 5.2.
5.3 Newton-Raphson methods

Newton-Raphson methods can be understood as acceleration techniques for descent methods. We also threshold the maximal time $\tau_m \leq 1$, since it is the theoretical value of the Newton-Raphson method.

**Newton-Raphson method**

A straightforward method for a direct search of the critical point of $G$ is the classical Newton method

$$H_m = \nabla^2 G(\lambda_m),$$

with $\nabla^2 G$ the Hessian of $G$. The time step $\tau_m$ is computed as in Section 5.2.

**Modified Newton-Raphson method**

The Hessian of $G$ degenerates far from its minimum as showed in Lemma 3.11. In practice, a classical Newton-Raphson method for solving $\nabla G(\lambda) = 0$ can be inaccurate at the first iterations in some cases. Instead one may notice that $\lambda_\ast$ is a critical point of $G(\lambda)$ if and only if it is a critical point of $(G(\lambda) - C)^2$ where $C$ is a constant which is smaller than the infimum of $G$. One expect the latter function to grow quadratically at infinity thus improving the conditioning of the Hessian. This suggests the modified Newton method (36) with

$$H_m = \alpha_m \nabla G(\lambda_m) \otimes \nabla G(\lambda_m) + \nabla^2 G(\lambda_m).$$

The time step $\tau_m$ is chosen as in the previous sections. Several choices are possible for $\alpha_m$. Following the heuristic one could impose $\alpha_m = (G(\lambda_m) - K)^{-1}$. In practice, we found out that the (empirical) choice $\alpha_m = \|\nabla G(\lambda_m)\|/(\|\nabla G(\lambda_m)\| + \|\nabla G(0)\|)$ yields good results.

6 Numerical experiments

In this section, we perform various numerical experiments in order to illustrate the theoretical results and to explore the behavior of the numerical algorithms.

6.1 Univariate polynomials on a segment

We consider with univariate SOS polynomials. We proceed as explained in Section 4, except that the monomial basis is replaced here by the orthogonal basis of shifted Chebychev polynomials $(T_i(x))_{i=1,...,k}$ satisfying $T_i(\cos(\theta) + 1)/2 = \cos(i\theta)$, for all $\theta \in \mathbb{R}$. The only modification of the method presented earlier concerns the definition of the $D_r$ matrices which become $D_r = w_r^t w_r \in \mathbb{R}^{r_k \times r_k}$ with $w_r = (T_0(x_r), T_1(x_r), \ldots, T_k(x_r))^t \in \mathbb{R}^{r_k}$. The reason is that shifted Chebychev polynomials have much better behavior in terms of numerical approximation, since they produce "uniformly distributed" polynomials in $[0,1]$, see [7] for comprehensive mathematical treatment. Of course this is better than monomials $x^i$ which concentrate at $x = 1$ for $i \to +\infty$. One can refer to [6] for a comparison between the use of Chebychev polynomials and monomials. In the following we propose different test cases to illustrate the various properties of the various descent and Newton-Raphson type methods proposed in Section 5. For univariate polynomials, the tests 0-1-2-3 are performed with the odd order option (31) of the weights: similar results are observed.
with \( g_1(x) = 1 \) and \( g_2(x) = x(1 - x) \), and so are not reported. Test 5 is performed with both the odd and even options.

**Test case 0.** The starting point of the iterative algorithm, the Lagrange multiplier \( \lambda = 0 \), defines a sum of square polynomial \( p[0](x) \) which depends on the degree \( n \) \( (r_\ast = i_\ast = n + 1) \), the weights in the sum of square ansatz (here given in (31) following the Markov-Lukács theorem, for \( n \) odd) so \( j_\ast = 2 \) and the polynomial basis. These SOS are represented on Figure 3. One observes a concentration near the boundaries which is characteristic of the properties of the Chebychev polynomials.

![Figure 3: Sum of squares \( p[0](x) \) (corresponding to the Lagrange multiplier \( \lambda = 0 \)) for different degrees \( n \). They depend only the matrices \( B_r \) and thus on the interpolation points, weight functions and the polynomial basis (here shifted Chebychev).](image)

**Test case 1.** We compare the convergence of the methods for an easy objective polynomial, that is a polynomial with low degree and far above \( 0 \): we take \( n = 5 \), \( r_\ast = i_\ast = n + 1 = 6 \), \( p(x) = x^5 + 1 \) and the weights \( g_1(x) = x \) with \( g_2(x) = 1 - x \) (so \( j_\ast = 2 \)). We observe on Figure 4 that the Newton type methods both reach the threshold precision of \( 10^{-8} \) after only 6 iterations. The implicit Euler and gradient descent methods need respectively 573 and 2727 iterations to reach the same error: this low convergence has been observed for many other test problems. This is why we continue the tests with the Newton methods only.

**Test case 2.** In this second test case, we illustrate the better performance of the modified Newton-Raphson method compared to the standard Newton-Raphson method. We choose a highly oscillating objective polynomial with lower bound equal to \( 0 \). It is given by \( n = 21 \), \( r_\ast = i_\ast = n + 1 = 22 \), \( p(x) = T_{21}(x) + 1 \) and the weights \( g_1(x) = x \) with \( g_2(x) = 1 - x \) (so \( j_\ast = 2 \)). We observe on Figure 5 that the modified Newton-Raphson method reaches a precision of around \( 10^{-8} \) in 40 iterations. In the case of the standard Newton-Raphson method, the adaptive time step quickly reduces to a very small value in order to keep decreasing the error at each iteration. A similar phenomena happens near convergence for the modified Newton-Raphson method. These behaviors can be interpreted thanks to the evolution of the condition number of the matrix \( H_m \) also.
Test case 3. Now, we illustrate the influence of the lower bound of \( p \) on the convergence of the method. To proceed, we compute a sum of squares approximation of the polynomial \( p(x) = T_{11}(x) + 1 + \alpha \) for various lower bounds \( \alpha \) \((n = 5, r_a = i_a = n + 1 = 6, j_a = 2)\).

The results are displayed on Figure 6. We observe that the number of iterations required to reach a precision of \( 10^{-8} \) seems to increase proportionally with \( |\log(\alpha)| \). The condition number of \( H_m \) and the norm of \( \lambda_m \) at convergence decays like some negative power of \( \alpha \). Interestingly enough, one also sees that the quadratic convergence of the (modified) Newton method seems to degenerate to linear convergence when \( \alpha \) goes to 0. All these behaviors can be interpreted thanks to the results of Lemma 3.5 and Lemma 3.11. We know from Lemma 3.5 that for \( \alpha = 0 \), \( p \) has a root \( x_0 \) in \([0, 1]\), and thus the coercivity of \( G \) is lost in some direction of the asymptotic cone of \( D \) (that of the Lagrange vector \( L(x_0) \)). Thus as \( \alpha \to 0 \), the minimizer \( \lambda_a^* \) may go to \(+\infty\) in the asymptotic cone which would explain here the explosion of the norm of \( \lambda \) and of the condition number of \( H_m \) as predicted by Lemma 3.11 and shown on Figure 6.

Test case 4. In this fourth test case we illustrate the influence of the degree \( n \) of the objective polynomial \( p(x) = x^n + 1 \) on the convergence of our method, with \( g_1(x) = x \) and \( g_2(x) = 1 - x \) for
Figure 5: Test case 2. Sum of square interpolation of \( p(x) = T_{21}(x) + 1 \). (Top left) error \( \|\nabla \mathcal{G}(\lambda_m)\|_2 \) vs. iteration \( m \); (Top right) Step size \( \tau_m \) vs. iteration \( m \); (Bottom left) Condition number of \( H_m \) vs. iteration \( m \); (Bottom right) Data \( (y_r = p(x_r)) \) and sum of squares \( p[\lambda](x) \) satisfying \( \|\nabla \mathcal{G}(\lambda)\| < 10^{-6} \).

\( n \) odd and \( g_1(x) = 1 \) and \( g_2(x) = x(1 - x) \) for \( n \) even.

The result are displayed on Figure 7. We observe that the number of iterations required to reach an error of \( 10^{-8} \) increases with the degree, but weakly. We also observe that the condition number \( \text{cond}(H_m) = \|H_m\|\|H_m^{-1}\| \) near convergence deteriorates with \( n \), approximatively quadratically.

6.2 Bivariate polynomials on a triangle

In this part we use our algorithm for the computation of a sum of squares representation of some positive polynomial \( p \in P_n[X,Y] \) on the triangle.
Figure 6: Test case 3. Influence of the lower bound $\alpha$ in the sum of square interpolation of $p(x) = (T_{11}(x) + 1) + \alpha$. (Top left) Number of iterations to converge vs. $\alpha$; (Top right) Condition number of $H_m$ at the last iteration vs. $\alpha$; (Bottom) Error $\|\nabla G(\lambda_m)\|_2$ vs. iteration $m$ for different lower bounds $\alpha$.

Figure 7: Test case 4. Influence of the degree $\alpha$ in the sum of square interpolation of $p(x) = x^n + 1$. (Left) Number of iterations to converge vs. $n$; (Right) Condition number of $H_m$ at the last iteration vs. $n$. 

23
Numerical setting. The barycentric coordinates corresponding to the vertices $S_1$, $S_2$ and $S_3$ of the triangle are denoted as $\mu_j$ for $j = 1, 2, 3$

$$
\mu_1(x, y) = 1 - x - y, \quad \mu_2(x, y) = x, \quad \mu_3(x, y) = y,
$$
The triangle is

$$
K = \{x = (x, y) \in \mathbb{R}^2, \quad \mu_1(x) \geq 0, \quad \mu_2(x) \geq 0, \quad \mu_3(x) \geq 0\}.
$$
The interpolation points are $x_r = (x_r, y_r)$ for $1 \leq r \leq r_* = (n + 1)(n + 2)/2$ are the distinct points of a cartesian grid intersected with the triangle. For a given polynomial $p \in P^n[X]$ of a given degree, the data is $z \in \mathbb{R}^{r_*}$ which is the vector with components $z_r = p(x_r, y_r)$. An illustration of the geometry is provided in Figure 8 where the degree is $n = 4$.

![Domain and interpolation points](image)

Figure 8: The simplex $K$ and interpolation points for $n = 4$.

We consider the ansatz

$$
p[\lambda](x) = \sum_{i=1}^{r_*} g_1(x, y) p_{i1}[\lambda](x, y)^2 + g_2(x, y) p_{i2}[\lambda](x, y)^2 + g_3(x, y) p_{i3}[\lambda](x, y)^2 + g_4(x, y) p_{i4}[\lambda](x, y)^2,
$$
where, arbitrarily with respect to the literature [12], the weights are

$$
\begin{cases}
\text{for } n = 2k + 1, & g_i = \mu_i \text{ for } i = 1, 2, 3 \text{ and } g_4 = \mu_1 \mu_2 \mu_3, \\
\text{for } n = 2k, & g_1 = \mu_2 \mu_3, \quad g_2 = \mu_3 \mu_1, \quad g_3 = \mu_1 \mu_2 \text{ and } g_4 = 1.
\end{cases}
$$

With this choice we recover in every cases $r_* = r_1 + r_2 + r_3 + r_4$. All polynomials are parametrized on the basis of bivariate monomials since Chebychev polynomials are not available on the triangle.

Test case 5. We approach the polynomial $p(x, y) = (T_4(x) + 1)(T_4(y) + 1)/4 + 10^{-3}$ on the 2D simplex with the modified Newton method. The numerical parameter are $n = 8$, $r_* = i_* = 45$ and $j_* = 4$. 24
We observe on Figure 9 that our method converges in this multivariate setting and reaches a precision of less than $10^{-8}$ in 210 iterations. We observe that the error decays slowly during the first 200 iterations before reaching usual quadratic speed of convergence of the Newton method near the minimizer of $G$. This result illustrates the ability of our algorithms to provided a computational strategy for the calculation of positive polynomials on bi-dimensional sets.

![Figure 9: Test case 5. Bivariate sum of square interpolation of the degree 8 polynomial $p(x, y) = (T_4(x) + 1)(T_4(y) + 1)/4 + 10^{-3}$ on the 2D simplex. (Left) error $\|\nabla G(\lambda_m)\|_2$ vs. iteration $m$; (Right) surface plot of the converged sum of square.](image)

Test case 6. In this last test case we are interested in the SOS approximation of the Motzkin polynomial [15]

$$p(x, y) = x^2y^4 + y^2x^4 - 3x^2y^2 + 1.$$ 

This polynomial is non-negative over $\mathbb{R}^2$ and famous for not being a sum of square in the sense that it admits no decomposition (3) with weights $\tilde{g}_i = \cdots = \tilde{g}_{j_s} = 1$ (whatever the choice of $i_*$ or, equivalently in this particular case, $j_*$). The parameters are $n = 6$, $r_* = i_* = 28$ and $j_* = 4$.

We use our method to approach this polynomial with the sum of square ansatz (42) but with two different weights: on the one hand we use the weights $g_i$ (43) for which we expect some convergence of the algorithm; on the other other hand we use the weights $\tilde{g}_i = 1$ for $i = 1, 2, 3, 4$.

In the latter case the non-convergence of the method would not converge in coherence with the non-existence of a sum of square decomposition for the Motzkin polynomial. This is indeed confirmed by our experiment as shown on Figure 10. The algorithm with weights $g_i$ converges while the algorithm with weights $\tilde{g}_i$ does not converge (Bottom right illustration in the Figure).

6.3 Concluding algorithmic remarks

All our numerical results show that the modified Newton-Raphson algorithm is able to compute polynomials which respect a sign condition on a given simple semi-algebraic set. Much more needs to
be investigated to evaluate the full potential of such methods. Here we detail to possible domains of research which are consequences of the multiple connections of our methods with the ones of Scientific Computing.

- The first one concerns the acceleration of the different methods. We have in mind that a C++ implementation needs to be tested.
- On this basis it will be possible to couple with codes in scientific computing (such as the ones evoked in [19] and the references therein) to evaluate the gain in robustness provided by the new algorithms.
- Finally we mention that an implementation of the gradient algorithms with absolute guarantee of the condition \( \lambda^{m+1} \in D \) is possible. Indeed start from \( \lambda^m \in D \). Since \( \lambda^{m+1} = \lambda^m - \Delta t \mathbf{d}^m \) for a given direction \( \mathbf{d}^m = (d^m_1, d^m_2, d^m_3) \), the condition \( \lambda^{m+1} \in D \) is satisfied provided \( M(\lambda^m) - \Delta t \sum d^m_i B_i \geq 0 \).

It is satisfied under the sufficient condition \( \rho (\sum d^m_i B_i ) \Delta t < \rho (M(\lambda^m))^{-1} \): here \( \rho(A) \) denotes the spectral radius of a given square matrix \( A \). This condition is very much a CFL stability condition.

References


