Border Basis relaxation for polynomial optimization

Marta Abril Bucero, Bernard Mourrain

Border basis relaxation for polynomial optimization

Marta Abril Bucero [a] , Bernard Mourrain [a]

[a] *EPI GALAAD, INRIA Méditerranée, 2004 Route des Lucioles, BP 93, 06902 Valbonne, France*

Abstract

A relaxation method based on border basis reduction which improves the efficiency of Lasserre's approach is proposed to compute the infimum of a polynomial function on a basic closed semi-algebraic set. A new stopping criterion is given to detect when the relaxation sequence reaches the infimum, using a sparse flat extension criterion. We also provide a new algorithm to reconstruct a finite sum of weighted Dirac measures from a truncated sequence of moments, which can be applied to other sparse reconstruction problems. As an application, we obtain a new algorithm to compute zero-dimensional minimizer ideals and the minimizer points or zero-dimensional G-radical ideals. Experiments show the impact of this new method on significant benchmarks.

Key words: Polynomial optimization, moment matrices, flat extension, border basis

1. Introduction

Computing the global infimum of a polynomial function f on a semi-algebraic set is a difficult but important problem, with many applications. A relaxation approach was proposed in (Lasserre, 2001) (see also (Parrilo, 2003), (Shor, 1987)) which approximates this problem by a sequence of finite dimensional convex optimization problems. These optimization problems can be formulated in terms of linear matrix inequalities on moment matrices associated to the set of monomials of degree $\leq t \in \mathbb{N}$ for increasing values of t. They can be solved by Semi-Definite Programming (SDP) techniques. The sequence of minima converges to the actual infimum f^* of the function under some hypotheses (Lasserre, 2001). In some cases, the sequence even reaches the infimum in a finite number of steps (Laurent, 2007; Nie et al., 2006; Marshall, 2009; Demmel et al., 2007; Ha and Pham, 2010; Nie, 2011). This approach has proved to be particularly fruitful in many problems (Lasserre, 2009). In contrast with numerical methods such as gradient descent

Email addresses: `Marta.Abril_Bucero@inria.fr` (Marta Abril Bucero), `Bernard.Mourrain@inria.fr` (Bernard Mourrain).

Preprint submitted to Linear Algebra and its Applications *August 22, 2015*

methods, which converge to a local extremum but with no guaranty for the global solution, this relaxation approach can provide certificates for the infimum value f^* in terms of sums of squares representations.

From an algorithmic and computational perspective, however some issues need to be considered.

The size of the SDP problems to be solved is a bottleneck of the method. This size is related to the number of monomials of degree $\leq t$ and increases exponentially with the number of variables and the degree t. Many SDP solvers are based on interior point methods which provide an approximation of the optimal moment sequence within a given precision in a polynomial time: namely $\mathcal{O}((p\,s^{3.5} + c\,p^2 s^{2.5} + c\,p^3 s^{0.5})\log(\epsilon^{-1}))$ arithmetic operations where $\epsilon > 0$ is the precision of the approximation, s is the size of the moment matrices, p is the number of parameters (usually of the order s^2) and c is the number of constraints (Nesterov and Nemirovski, 1994). Thus reducing the size s or the number of parameters p can significantly improve the performance of these relaxation methods. Some recent works address this issue, using symmetries (see e.g. (Riener et al., 2013)) or polynomial reduction (see e.g. (Lasserre et al., 2012)). In this paper, we extend this latter approach.

While determining the infimum value of a polynomial function on a semi-algebraic set is important, computing the minimizer points, is also critical in many applications. Determining when and how these minimizer points can be computed from the relaxation sequence is a problem that has been addressed, for instance in (Henrion and Lasserre., 2005; Nie, 2012) using full moment matrices. This approach has been used for solving polynomial equations (Laurent, 2007; Lasserre et al., 2008, 2009; Lasserre, 2009).

The optimization problem can be reformulated as solving polynomial equations related to the (minimal) critical value of the polynomial f on a semi-algebraic set. Polynomial solvers based, for instance, on Gröbner basis or border basis computation can then be used to recover the real critical points from the complex solutions of (zero-dimensional) polynomial systems (see e.g. (Parrilo and Sturmfels, 2003; Safey El Din, 2008; Greuet and Safey El Din, 2011)). This type of methods relies entirely on polynomial algebra and univariate root finding. So far, there is no clear comparison of these elimination methods and the relaxation approaches.

Contributions. We propose a new method which combines Lasserre's SDP relaxation approach with polynomial algebra, in order to increase the efficiency of the optimization algorithm. Border basis computations are considered for their numerical stability (Mourrain and Trébuchet, 2005; Mourrain and Trébuchet, 2008). In principle, any graded normal form technique could be used here.

A new stopping criterion is given to detect when the relaxation sequence reaches the infimum, using a flat extension criterion from (Laurent and Mourrain, 2009). We also provide a new algorithm to reconstruct a finite sum of weighted Dirac measures from a truncated sequence of moments. This reconstruction method can be used in other problems such as tensor decomposition (Brachat et al., 2010) and multivariate sparse interpolation (Giesbrecht et al., 2009).

As shown in (Abril Bucero and Mourrain, 2013; Nie et al., 2006; Demmel et al., 2007; Marshall, 2009; Nie, 2011; Ha and Pham, 2010), an exact SDP relaxation can be constructed for "well-posed" optimization problems. As an application, we obtain a new algorithm to compute zero-dimensional minimizer ideals and the minimizer points, or zero-dimensional G-radicals. Experiments show the impact of this new method compared to the previous relaxation constructions.

Content. The paper is organized as follows. Section 2 describes the minimization problem and includes a running example to explain the different steps of our method. In Section 3-5, we describe the ingredients of the main algorithm, which is described in Section 7. In section 3, we describe the SDP relaxation hierarchies (full moment matrices and border basis). In Section 4, we tackle the sub-problem of how to compute the optimal linear form through the solution of a SDP problem. In Section 5, we tackle the sub-problem of how to verify that we have found the infimum, checking the flat extension property using orthogonal polynomials. In Section 6, we tackle the sub-problem of how to compute the minimizer points using multiplication matrices. Section 7 gives a description of the complete minimization algorithm. Section 8 analyses cases for which an exact relaxation can be constructed. Section 9 concludes experimentation.

2. Minimization problem

Let $f \in \mathbb{R}[\mathbf{x}]$ be a polynomial function with real coefficients and let $\mathbf{g} = \{g_1^0, \ldots, g_{n_1}^0; g_1^+, \ldots, g_{n_2}^+\} \in \mathbb{R}[\mathbf{x}]$ be a set of constraints which is the union of a finite subset $\mathbf{g}^0 = \{g_1^0, \ldots, g_{n_1}^0\}$ of polynomials corresponding to the equality constraints and a finite subset $\mathbf{g}^+ = \{g_1^+, \ldots, g_{n_2}^+\}$ corresponding to the non-negativity constraints. The basic semi-algebraic set defined by the constraints $\mathbf{g}$ will be denoted $S := \mathcal{S}(\mathbf{g}) = \{\mathbf{x} \in \mathbb{R}^n \mid g_1^0(\mathbf{x}) = \cdots = g_{n_1}^0(\mathbf{x}) = 0, g_1^+(\mathbf{x}) \geq 0, ..., g_{n_2}^+(\mathbf{x}) \geq 0\}$. We assume that $S \neq \emptyset$ and that f is bounded by below on S (i.e. $\inf_{\mathbf{x}\in S} f(\mathbf{x}) > -\infty$). The minimization problem that we consider throughout the paper is the following: compute

$$\begin{aligned} &\inf_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \\ s.t.\ & g_1^0(\mathbf{x}) = \cdots = g_{n_1}^0(\mathbf{x}) = 0 \\ & g_1^+(\mathbf{x}) \geq 0, ..., g_{n_2}^+(\mathbf{x}) \geq 0 \end{aligned} \tag{1}$$

When $n_1 = n_2 = 0$, there is no constraint and $S = \mathbb{R}^n$. In this case, we are considering a global unconstrained minimization problem.

The points $\mathbf{x}^* \in \mathbb{R}^n$ which satisfy $f(\mathbf{x}^*) = \inf_{\mathbf{x}\in S} f(\mathbf{x})$ are called the *minimizer points* of f on S. The set of minimizer points is denoted $V_{min} = \{\mathbf{x}^* \in S\ s.t\ f(\mathbf{x}^*) = \inf_{\mathbf{x}\in S} f(\mathbf{x})\}$. The ideal of $\mathbb{R}[\mathbf{x}]$ defining the set V_{min} is denoted I_{min}. The value $f^* = \inf_{\mathbf{x}\in S} f(\mathbf{x})$ is called the *minimum* of f on S, when the set of minimizers is not empty.

If the set of minimizer points is not empty, we say that the *minimization problem is feasible*. The minimization problem is not feasible means that $V_{min} = \emptyset$ and $I_{min} = \mathbb{R}[\mathbf{x}]$.

We present a running problem to explain the different steps of our method to solve Problem (1).

Example 2.1.

$$\inf_{\mathbf{x}\in\mathbb{R}^2} f(x, y) = (x-1)^2(x-2)^2(x^2+1) + (y-1)^2(y^2+1)$$

This example is a global unconstrained minimization problem. We take its gradient ideal and hereafter we consider the problem of minimizing the aforementioned function over its gradient ideal.

$$\begin{aligned} \inf_{\mathbf{x}\in\mathbb{R}^2} \; f(x,y) &= (x-1)^2(x-2)^2(x^2+1)+(y-1)^2(y^2+1) \\ s.t.\; 6x^5-30x^4+56x^3-54x^2+34x-12 &= 0 \\ 4y^3-6y^2+4y-2 &= 0 \end{aligned}$$

The minimizer points are $(1,1)$ and $(2,1)$. The minimum is $f^*=0$.

3. Convex relaxations

In this section, we describe the finite dimensional convex optimization problems that we consider to solve the polynomial optimization problem (1). We recall the well-known full moment matrix relaxation and then we explain the border basis relaxation that we use. At the end of the section we compute the border basis for our running example.

But first, we introduce the notation we are going to use. Let $\mathbb{R}[\mathbf{x}]$ be the set of the polynomials in the variables $\mathbf{x}=(x_1,\ldots,x_n)$, with real coefficients in $\mathbb{R}$. For $\alpha\in\mathbb{N}^n$, $\mathbf{x}^\alpha=x_1^{\alpha_1}\cdots x_n^{\alpha_n}$ is the monomial with exponent α and degree $|\alpha|=\sum_i\alpha_i$. The set of all monomials in $\mathbf{x}$ is denoted $\mathcal{M}=\mathcal{M}(\mathbf{x})$. For a polynomial $f=\sum_\alpha f_\alpha\mathbf{x}^\alpha$, its support is $supp(f):=\{\mathbf{x}^\alpha \mid f_\alpha\neq 0\}$, the set of monomials occurring with a nonzero coefficient in f.

For $t\in\mathbb{N}$ and $F\subseteq\mathbb{R}[\mathbf{x}]$, we introduce the following sets: F_t is the set of elements of F of degree $\leq t$; $\langle F\rangle=\big\{\sum_{f\in F}\lambda_f f \mid f\in F,\lambda_f\in\mathbb{R}\big\}$ is the linear span of F; if F is a vector space, F^* is the dual space of linear forms from F to $\mathbb{R}$; $(F)=\big\{\sum_{f\in F}p_f f \mid p_f\in\mathbb{R}[\mathbf{x}], f\in F\big\}$ is the ideal in $\mathbb{R}[\mathbf{x}]$ generated by F; $\langle F\,|\,t\rangle$ is the vector space spanned by $\{\mathbf{x}^\alpha f \mid f\in F_t, |\alpha|\leq t-\deg(f)\}$; $F\cdot F:=\{p\,q \mid p,q\in F\}$; $\Sigma^2(F)=\{\sum_{i=1}^s f_i^2 \mid f_i\in F\}$ is the set of finite sums of squares of elements of F; for $F=\{f_1,\ldots,f_m\}\subset\mathbb{R}[\mathbf{x}]$, $\prod(F)=\{\prod_{i=1}^m f_i^{\epsilon_i} \mid \epsilon_i\in\{0,1\}\}$.

3.1. Hierarchies of relaxation problems

Definition 3.1. Given a finite dimensional vector space $E\subset\mathbb{R}[\mathbf{x}]$ and a set of constraints $G\subset\mathbb{R}[\mathbf{x}]$, we define the quadratic module of G on E as

$$\begin{aligned}\mathcal{Q}_{E,G}=\{&\textstyle\sum_{g\in G^0} g\,h+\sum_{g'\in G^+} g'\,h' \\ &\mid h\in E, g\,h\in\langle E\cdot E\rangle, h'\in\Sigma^2(E), g'h'\in\langle E\cdot E\rangle\}.\end{aligned}$$

If $G^\star\subset\mathbb{R}[\mathbf{x}]$ is the set of constraints such that $G^{\star 0}=G^0$ and $G^{\star +}=\prod(G^+)$, the (truncated) quadratic module $\mathcal{Q}_{E,G^\star}$ is called the (truncated) preordering of G and denoted $\mathcal{Q}^\star_{E,G}$ or $\mathcal{P}_{E,G}$.

By construction, $\mathcal{Q}_{E,G}\subset\langle E\cdot E\rangle$ is a cone of polynomials which are non-negative on the semi-algebraic set S.

We consider now its dual cone.

Definition 3.2. Given a finite dimensional vector space $E \subset \mathbb{R}[\mathbf{x}]$ which contains 1 and a set of constraints $G \subset \mathbb{R}[\mathbf{x}]$, we define

$$\mathcal{L}_{E,G} := \{\Lambda \in \langle E \cdot E\rangle^* \mid \Lambda(p) \geq 0,\ \forall p \in \mathcal{Q}_{E,G}, \Lambda(1) = 1\}.$$

The convex set associated to the preordering $\mathcal{Q}^\star_{E,G} = \mathcal{Q}_{E,G^\star}$ is denoted $\mathcal{L}^\star_{E,G}$.

By this definition, for any element $\Lambda \in \mathcal{L}_{E,G}$ and any $g \in \langle G^0\rangle \cap E$, we have $\Lambda(g) = 0$.

We introduce now truncated Hankel operators, which will play a central role in the construction of the minimizer ideal of f on S.

Definition 3.3. For a linear form $\Lambda \in \langle E \cdot E\rangle^*$, we define the map $H^E_\Lambda : E \to E^*$ by $H^E_\Lambda(p)(q) = \Lambda(p\,q)$ for $p, q \in E$. It is called the truncated Hankel operator of Λ on the subspace E.

Its matrix in the monomial and dual bases of E and E^* is usually called the moment matrix of Λ. The kernel of this truncated Hankel operator will be used to compute generators of the minimizer ideal, as we will see.

Definition 3.4. Let $E \subset \mathbb{R}[\mathbf{x}]$ such that $1 \in E$ and a set of constraints $G \subset \mathbb{R}[\mathbf{x}]$. We define the following extrema:
- $f^\mu_{E,G} = \inf\ \{\Lambda(f) \text{ s.t. } \Lambda \in \mathcal{L}_{E,G}\}$,
- $f^{sos}_{E,G} = \sup\ \{\gamma \in \mathbb{R} \text{ s.t. } f - \gamma \in \mathcal{Q}_{E,G}\}$.

By convention if the sets are empty, $f^{sos}_{E,G} = -\infty$ and $f^\mu_{E,G} = +\infty$.

If $E = \mathbb{R}[\mathbf{x}]_t$ and $G^0 = \langle \mathbf{g}^0 \mid 2t\rangle$, we also denote $f^\mu_{E,G}$ by $f^\mu_{t,G}$ and $f^{sos}_{E,G}$ by $f^{sos}_{t,G}$.

We easily check that $f^{sos}_{E,G} \leq f^\mu_{E,G}$, since if there exists $\gamma \in \mathbb{R}$ such that $f - \gamma = q \in \mathcal{Q}_{E,G}$ then $\forall \Lambda \in \mathcal{L}_{E,G}$, $\Lambda(f - \gamma) = \Lambda(f) - \gamma = \Lambda(q) \geq 0$.

If $\mathcal{S}(G) \subset S$, we also have $f^\mu_{E,G} \leq f^*$ since for any $\mathbf{s} \in S$, the evaluation $\mathbf{1_s} : p \in \mathbb{R}[\mathbf{x}] \mapsto p(\mathbf{s})$ is in $\mathcal{L}_{E,G}$.

Notice that if $E \subset E'$, $G \subset G'$ then $\mathcal{Q}_{E,G} \subset \mathcal{Q}_{E',G'}$, $\mathcal{L}_{E',G'} \subset \mathcal{L}_{E,G}$, $f^\mu_{E,G} \leq f^\mu_{E',G'}$ and $f^{sos}_{E,G} \leq f^{sos}_{E',G'}$.

3.2. Full moment matrix relaxation hierarchy

The relaxation hierarchies introduced in (Lasserre, 2001) correspond to the case where $E = \mathbb{R}[\mathbf{x}]_t$, $G^0 = \langle \mathbf{g}^0 | 2t\rangle$ and $G^+ = \mathbf{g}^+$.

The quadratic module $\mathcal{Q}_{\mathbb{R}[\mathbf{x}]_t,G}$ is denoted $\mathcal{Q}_{t,\mathbf{g}}$ and $\mathcal{L}_{\mathbb{R}[\mathbf{x}]_t,G}$ is denoted $\mathcal{L}_{t,\mathbf{g}}$. Hereafter, we will also call the Lasserre hierarchy, the full moment matrix relaxation hierarchy. It corresponds to the sequences

$$\cdots \subset \mathcal{L}_{t+1,\mathbf{g}} \subset \mathcal{L}_{t,\mathbf{g}} \subset \cdots \text{ and } \cdots \subset \mathcal{Q}_{t,\mathbf{g}} \subset \mathcal{Q}_{t+1,\mathbf{g}} \subset \cdots$$

which yield the following increasing sequences for $t \in \mathbb{N}$:

$$\cdots f^\mu_{t,\mathbf{g}} \leq f^\mu_{t+1,\mathbf{g}} \leq \cdots \leq f^* \text{ and } \cdots f^{sos}_{t,\mathbf{g}} \leq f^{sos}_{t+1,\mathbf{g}} \leq \cdots \leq f^*.$$

The foundation of Lasserre's method is to show that these sequences converge to f^*. This is proved under some conditions in (Lasserre, 2001). It has also been shown that the limit can even be reached in a finite number of steps in some cases, see e.g. (Lasserre et al., 2009; Nie et al., 2006; Marshall, 2009; Ha and Pham, 2010; Nie, 2011; Abril Bucero and Mourrain, 2013). In this case, the relaxation is said to be *exact*.

3.3. Border basis relaxation hierarchy

In the following we are going to use another type of relaxation hierarchy, which involves border basis computation. Its aim is to reduce the size of the convex optimization problems solved at each level of the relaxation hierarchy. As we will see in Section 9, the impact on the performance of the relaxation approach is significant. We briefly recall the properties of border basis that we need and describe how they are used in the construction of this relaxation hierarchy.

Given a vector space $E \subseteq \mathbb{R}[\mathbf{x}]$, its prolongation $E^+ := E + x_1 E + \ldots + x_n E$ is again a vector space.

The vector space E is said to be *connected to 1* if $1 \in E$ and there exists a finite increasing sequence of vector spaces $E_0 \subset E_1 \subset \cdots \subset E$ such that $E_0 = \langle 1 \rangle$, $E_{i+1} \subset E_i^+$. For a monomial set $B \subseteq \mathcal{M}$, $B^+ = B \cup x_1 B \cup \cdots \cup x_n B$ and $\partial B = B^+ \setminus B$. We easily check that $\langle B \rangle^+ = \langle B^+ \rangle$ and $\langle B \rangle$ is connected to 1 iff $1 \in B$ and for every monomial $m \neq 1$ in B, $m = x_{i_0} m'$ for some $i_0 \in [1, n]$ and some monomial $m' \in B$. In this case, we will say that the monomial set B is connected to 1.

Definition 3.5. Let $B \subset \mathcal{M}$ be connected to 1. A family $F \subset \mathbb{R}[\mathbf{x}] = R$ is a border basis for B in degree $t \in \mathbb{N}$, if $\forall f, f' \in F_t$,
- $supp(f) \subseteq B^+ \cap R_t$,
- f has exactly **one** monomial in ∂B, denoted $\gamma(f)$ and called the leading monomial of f.
- $\gamma(f) = \gamma(f')$ implies $f = f'$,
- $\forall m \in \partial B \cap R_t$, $\exists f \in F$ s.t. $\gamma(f) = m$,
- $R_t = \langle B \rangle_t \oplus \langle F|t \rangle$.

A border basis F for B in all degrees t is called a border basis for B. F is *graded* if moreover $\deg(\gamma(f)) = \deg(f)\ \forall f \in F$.

There are efficient algorithms to check that a given family F is a border basis for B in degree t and to construct such family from a set of polynomials. We refer to (Mourrain, 1999; Mourrain and Trébuchet, 2005; Mourrain and Trébuchet, 2008, 2012) for more details. We will use these tools as "black boxes" in the following.

For a border basis F for B in degree t, we denote by $\pi_{F,B}$ the projection of R_t on $\langle B_t \rangle$ along $\langle F|t \rangle$. We easily check that
- $\forall m \in B_t$, $\pi_{B,F}(m) = m$,
- $\forall m \in \partial B \cap R_t$, $\pi_{B,F}(m) = m - f$, where f is the (unique) polynomial in F for which $\gamma(f) = m$, assuming the polynomials $f \in F$ are normalized so that the coefficient of $\gamma(f)$ is 1.

If F is a graded border basis in degree t, one easily verifies that $\deg(\pi_{F,\mathcal{B}}(m)) \leq \deg(m)$ for $m \in \mathcal{M}_t$.

Border basis hierarchy. The sequence of relaxation problems that we will use hereafter is defined as follows. For each $t \in \mathbb{N}$, we construct the graded border basis F_{2t} of $\mathbf{g}^0$ in degree $2t$. Let B be the set of monomials (connected to 1) for which F is a border basis in degree $2t$. We define $E_t := \langle B_t \rangle$, G_t is the set of constraints such that $G_t^0 = \{m - \pi_{B_t,F_{2t}}(m),\ m \in B_t \cdot B_t\}$ and $G_t^+ = \pi_{B_t,F_{2t}}(\mathbf{g}^+)$, and consider the relaxation sequence

$$\mathcal{Q}_{E_t,G_t} \subset \langle B_t \cdot B_t \rangle \text{ and } \mathcal{L}_{E_t,G_t} \subset \langle B_t \cdot B_t \rangle^* \tag{2}$$

for $t \in \mathbb{N}$. Since the subsets B_t are not necessarily nested, these convex sets are not necessarily included in each other. However, by construction of the graded border basis of $\mathbf{g}$, we have the following inclusions

$$\cdots \subset \langle F_{2t}|2t\rangle \subset \langle F_{2t+2}|2t+2\rangle \subset \cdots (\mathbf{g}^0),$$

and we can relate the border basis relaxation sequences with the corresponding full moment matrix relaxation hierarchy, using the following proposition:

Proposition 3.6. *Let $t \in \mathbb{N}$, $B \subset \mathbb{R}[\mathbf{x}]_{2t}$ be a monomial set connected to 1, $F \subset \mathbb{R}[\mathbf{x}]$ be a border basis for B in degree $2t$, $E := \langle B_t\rangle$, $E' := \mathbb{R}[\mathbf{x}]_t$, G, G' be sets of constraints such that $G^0 = \{m - \pi_{B,F}(m),\ m \in B_t \cdot B_t\}$, $G'^0 = \langle F|2\,t\rangle$, $G^+ = G'^+$. Then for all $\Lambda \in \mathcal{L}_{E,G}$, there exists a unique $\Lambda' \in \mathcal{L}_{E',G'}$ which extends Λ. Moreover, Λ' satisfies $\operatorname{rank} H_{\Lambda'}^{E'} = \operatorname{rank} H_{\Lambda}^{E}$ and $\ker H_{\Lambda'}^{E'} = \ker H_{\Lambda}^{E} + \langle F\,|\,t\rangle$.*

Proof. As $F \subset \mathbb{R}[\mathbf{x}]$ is a border basis for B in degree $2t$, we have $\mathbb{R}[\mathbf{x}]_{2\,t} = \langle B\rangle_{2\,t} \oplus \langle F\,|\,2\,t\rangle$. As $\langle B_t \cdot B_t\rangle \subset \langle B\rangle_{2\,t} \oplus \langle G^0\rangle$, $\langle G^0\rangle \subset \langle G'^0\rangle = \langle F\,|\,2t\rangle$ and $\mathbb{R}[\mathbf{x}]_{2\,t} = \langle B\rangle_{2\,t} \oplus \langle F\,|\,2t\rangle$, we deduce that for all $\Lambda \in \mathcal{L}_{E,G}$, there exists a unique $\Lambda' \in \mathbb{R}[\mathbf{x}]_{2\,t}^*$ s.t. $\Lambda'_{|\langle B\rangle_{2t}} = \Lambda$ and $\Lambda'(\langle F\,|\,2t\rangle) = 0$.

Let us first prove that $\Lambda' \in \mathcal{L}_{E',G'} = \mathcal{L}_{t,G'}$. As any element q' of $\mathcal{Q}_{E',G'}$ can be decomposed as a sum of an element q of $\mathcal{Q}_{E,G}$ and an element $p \in \langle F|2t\rangle$, we have $\Lambda'(q') = \Lambda'(q) + \Lambda'(p) = \Lambda(q) \geq 0$. This shows that $\Lambda' \in \mathcal{L}_{E',G'}$.

Let us prove now that $\ker H_{\Lambda'}^{E'} = \ker H_{\Lambda}^{E} + \langle F\,|\,t\rangle$ where $E := \langle B_t\rangle$, $E' := \mathbb{R}[\mathbf{x}]_t$. As $E \cdot \langle F\,|\,t\rangle \subset \langle F\,|\,2t\rangle = G'^0$, we have $\Lambda'(E \cdot \langle F\,|\,t\rangle) = 0$ so that

$$\langle F\,|\,t\rangle \subset \ker H_{\Lambda'}^{E'}. \tag{3}$$

For any element $b \in \ker H_{\Lambda}^{E}$ we have $\forall b' \in E$, $\Lambda(b\,b') = \Lambda'(b\,b') = 0$. As $\Lambda'(E \cdot \langle F\,|\,t\rangle) = 0$ and $E' = E \oplus \langle F\,|\,t\rangle$, for any element $e \in E$, $\Lambda'(b\,e) = 0$. This proves that

$$\ker H_{\Lambda}^{E} \subset \ker H_{\Lambda'}^{E'}. \tag{4}$$

Conversely as $E' = E \oplus \langle F\,|\,t\rangle$, any element of E' can be reduced modulo $\langle F\,|\,t\rangle$ to an element of E, which shows that

$$\ker H_{\Lambda'}^{E'} \subset \ker H_{\Lambda}^{E} + \langle F\,|\,t\rangle. \tag{5}$$

From the inclusions (3), (4) and (5), we deduce that $\ker H_{\Lambda'}^{E'} = \ker H_{\Lambda}^{E} + \langle F\,|\,t\rangle$ and that $\operatorname{rank} H_{\Lambda'}^{E'} = \operatorname{rank} H_{\Lambda}^{E}$. □

We deduce from this proposition that $f_{E_t,G_t}^{\mu} = f_{t,\langle F_{2t}|2t\rangle}^{\mu}$. The sequence of convex sets $\mathcal{L}_{E_t,G_t}$ can be seen as the projections of nested convex sets

$$\cdots \supset \mathcal{L}_{t,\mathbf{g}} \supset \mathcal{L}_{t+1,\mathbf{g}} \supset \cdots$$

so that we have $\cdots \leq f_{E_t,G_t}^{\mu} \leq f_{E_{t+1},G_{t+1}}^{\mu} \leq \cdots \leq f^*$. We check that similar properties hold for $\mathcal{Q}_{E_t,G_t}$, $\mathcal{Q}_{t,\mathbf{g}}$ and $f_{E_t,G_t}^{sos} = f_{t,\mathbf{g}}^{sos}$, taking the quotient modulo $\langle F_{2t}|2t\rangle$.

Now we compute the border basis for our running example 2.1 and the monomials that we can reduce by using this border basis.

Example 3.7. We take the set of constraint $g^0 = \{6x^5 - 30x^4 + 56x^3 - 54x^2 + 34x - 12, 4y^3 - 6y^2 + 4y - 2\}$ and $t = 3$.

- The border basis is $F_3 = \{x^5 - 5x^4 + 9.333x^3 - 9x^2 + 5.66x - 2, y^3 - 1.5y^2 + y - 0.5\}$
- The monomial basis in degree ≤ 3 is:

$$B_3 = \{1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2\}$$

  The monomial y^3 is the leading term of an element of F_3.
- The border basis SDP relaxation is constructed from the reduction of the monomials in $B_3 \cdot B_3$. The following monomials are reduced by the border basis

$$\{y^3, xy^3, y^4, x^5, x^2y^3, xy^4, x^6, x^5y, x^3y^3, x^2y^4\}.$$

  This yields the following constraints:

$$\begin{aligned}
y^3 &\equiv 0.5 - \ y + 1.5\, y^2 \\
xy^3 &\equiv 0.5\, x - \ xy + 1.5\, xy^2 \\
y^4 &\equiv 0.75 - \ y + 1.25\, y^2 \\
x^2y^3 &\equiv 0.5\, x^2 - \ x^2y + 1.5\, x^2y^2 \\
xy^4 &\equiv 0.75\, x - \ xy + 1.25\, xy^2 \\
x^5 &\equiv 2 - 5.666\, x + 9\, x^2 - 9.333\, x^3 + 5\, x^4 \\
x^6 &\equiv 10 - 26.333\, x + 39.333\, x^2 - 37.666\, x^3 + 15.666\, x^4 \\
x^5y &\equiv 2\, y - 5.666\, xy + 9\, x^2y - 9.333\, x^3y + 5\, x^4y \\
x^3y^3 &\equiv 0.5\, x^3 - \ x^3y + 1.5\, x^3y^2 \\
x^2y^4 &\equiv 0.75\, x^2 - \ x^2y + 1.25\, x^2y^2
\end{aligned}$$

4. Optimal linear form

In this section we introduce the notion of *optimal linear form for* f, involved in the computation of I_{min} (also called generic linear form when $f = 0$ in (Lasserre et al., 2009, 2012)). In order to find this optimal linear form we solve a Semi-Definite Programming (SDP) problem, which involves truncated Hankel matrices associated with the monomial basis and the reduction of their products by the border basis, as described in the previous section. This allows us to reduce the size of the matrix and the number of parameters. At the end of this section we compute the optimal linear form for our running example.

Definition 4.1. $\Lambda^* \in \mathcal{L}_{E,G}$ is optimal for f if

$$\text{rank } H^E_{\Lambda^*} = \max_{\Lambda \in \mathcal{L}_{E,G}, \Lambda(f) = f^\mu_{E,G}} \text{rank } H^E_{\Lambda}.$$

The next result shows that only elements in I_{min} are involved in the kernel of a truncated Hankel operator associated with an optimal linear form for f.

Theorem 4.2. *Let $E \subset \mathbb{R}[\mathbf{x}]$ such that $1 \in E$ and $f \in \langle E \cdot E \rangle$ and let $G \subset \mathbb{R}[\mathbf{x}]$ be a set of constraints with $V_{min} \subset \mathcal{S}(G)$. If $\Lambda^* \in \mathcal{L}_{E,G}$ is optimal for f and such that $\Lambda^*(f) = f^*$, then* $\ker H^E_{\Lambda^*} \subset I_{min}$.

Proof. The proof is similar e.g. to (Lasserre et al., 2012)[Theorem 4.9]. $\square$

Let us describe how optimal linear forms are computed by solving convex optimization problems:

Algorithm 4.1: Optimal Linear Form

Input: $f \in \mathbb{R}[\mathbf{x}]$, $B_t = (\mathbf{x}^\alpha)_{\alpha \in A}$ a monomial set of degree $\leq t$ containing 1 with $f = \sum_{\alpha \in A+A} f_\alpha \mathbf{x}^\alpha \in \langle B_t \cdot B_t \rangle$, $G \subset \mathbb{R}[\mathbf{x}]$.
Output: the infimum $f^{\mu}_{t,G}$ of $\sum_{\alpha \in A+A} \lambda_\alpha f_\alpha$ subject to:
– $H^{B_t}_{\Lambda^*} = (h_{\alpha,\beta})_{\alpha,\beta \in A} \succcurlyeq 0$,
– $H^{B_t}_{\Lambda^*}$ satisfies the Hankel constraints
  $h_{0,0} = 1$, and $h_{\alpha,\beta} = h_{\alpha',\beta'}$ if $\alpha + \beta = \alpha' + \beta'$,
– $\Lambda^*(g^0) = \sum_{\alpha \in A+A} g^0_\alpha \lambda_\alpha = 0$ for all $g^0 = \sum_{\alpha \in A+A} g^0_\alpha \mathbf{x}^\alpha \in G^0 \cap \langle B_t \cdot B_t \rangle$.
– $H^{B_{t-w}}_{g^+ \cdot \Lambda^*} \succcurlyeq 0$ for all $g^+ \in G^+$ where $w = \lceil \frac{deg(g^+)}{2} \rceil$.
and $\Lambda^* \in \langle B_t \cdot B_t \rangle^*$ represented by the vector $[\lambda_\alpha]_{\alpha \in A+A}$.

This optimization algorithm involves a Semidefinite programming problem, corresponding to the optimization of a linear functional on the intersection of a linear subspace with the convex set of positive semidefinite matrices. It is a convex optimization problem, which can be solved efficiently by SDP solvers. If an interior point method is used, the solution Λ^* is in the interior of a face on which the infimum $\Lambda^*(f)$ is reached so that Λ^* is optimal for f. This is the case for tools such as CSDP, SDPA, SDPA-GMP, and MOSEK that we will use in the experiments.

Example 4.3. For the running example 2.1 and the relaxation order $t = 3$, we solve the following SDP problem:

$$
\begin{aligned}
&\inf\ \Lambda(f) = 2.75 - \Lambda(y) - 4.333\Lambda(x) + 0.25\Lambda(y^2) + 2.333\Lambda(x^2) + 0.333\Lambda(x^3) - 0.333\Lambda(x^4)\\
&\textit{with } \Lambda \textit{ s.t.}\\
&\Lambda(y^3) = 0.5 - \Lambda(y) + 1.5\Lambda(y^2)\\
&\Lambda(xy^3) = 0.5\Lambda(x) - \Lambda(xy) + 1.5\Lambda(xy^2)\\
&\Lambda(y^4) = 0.75 - \Lambda(y) + 1.25\Lambda(y^2)\\
&\Lambda(x^5) = 2 - 5.666\Lambda(x) + 9\Lambda(x^2) - 9.333\Lambda(x^3) + 5\Lambda(x^4)\\
&\Lambda(x^2y^3) = 0.5\Lambda(x^2) - \Lambda(x^2y) + 1.5\Lambda(x^2y^2)\\
&\Lambda(xy^4) = 0.75\Lambda(x) - \Lambda(xy) + 1.25\Lambda(xy^2)\\
&\Lambda(x^6) = 10 - 26.333\Lambda(x) + 39.333\Lambda(x^2) - 37.666\Lambda(x^3) + 15.666\Lambda(x^4)\\
&\Lambda(x^5y) = 2\Lambda(y) - 5.666\Lambda(xy) + 9\Lambda(x^2y) - 9.333\Lambda(x^3y) + 5\Lambda(x^4y)\\
&\Lambda(x^3y^3) = 0.5\Lambda(x^3) - \Lambda(x^3y) + 1.5\Lambda(x^3y^2)\\
&\Lambda(x^2y^4) = 0.75\Lambda(x^2) - \Lambda(x^2y) + 1.25\Lambda(x^2y^2)\\
&\Lambda(1) = 1
\end{aligned}
$$

and

$$H_\Lambda^{B_3} := \begin{pmatrix} 1 & a & b & c & d & e & f & g & h \\ a & c & d & f & g & h & i & j & k \\ b & d & e & g & h & \Lambda(y^3) & j & k & \Lambda(xy^3) \\ c & f & g & i & j & k & \Lambda(x^5) & l & m \\ d & g & h & j & k & \Lambda(xy^3) & l & m & \Lambda(x^2y^3) \\ e & h & \Lambda(y^3) & k & \Lambda(xy^3) & \Lambda(y^4) & m & \Lambda(x^2y^3) & \Lambda(xy^4) \\ f & i & j & \Lambda(x^5) & l & m & \Lambda(x^6) & \Lambda(x^5y) & n \\ g & j & k & l & m & \Lambda(x^2y^3) & \Lambda(x^5y) & n & \Lambda(x^3y^3) \\ h & k & \Lambda(xy^3) & m & \Lambda(x^2y^3) & \Lambda(xy^4) & n & \Lambda(x^3y^3) & \Lambda(x^2y^4) \end{pmatrix} \succcurlyeq 0$$

where $a = \Lambda(x)$, $b = \Lambda(y)$, $c = \Lambda(x^2)$, $d = \Lambda(xy)$, $e = \Lambda(y^2)$, $f = \Lambda(x^3)$, $g = \Lambda(x^2y)$, $h = \Lambda(xy^2)$, $i = \Lambda(x^4)$,$j = \Lambda(x^3y)$, $k = \Lambda(x^2y^2)$, $l = \Lambda(x^4y)$, $m = \Lambda(x^3y^2)$, $n = \Lambda(x^4y^2)$ and $\Lambda(y^3) = 0.5 - b + 1.5e, \Lambda(y^4) = 0.75 - b + 1.25e$, $\Lambda(x^2y^3) = 0.5c - g + 1.5k, \Lambda(xy^3) = 0.5a - d + 1.5h$, $\Lambda(x^5) = 2 - 5.666a + 9c - 9.333f + 5i, \Lambda(x^5y) = 2b - 5.666d + 9g - 9.333j + 5l$, $\Lambda(x^6) = 10 - 26.333a + 39.333c - 37.666f + 15.666i, \Lambda(xy^4) = 0.75a - d + 1.25h$, $\Lambda(x^3y^3) = 0.5f - j + 1.5m, \Lambda(x^2y^4) = 0.75c - g + 1.25k$.

A **solution** is: $\Lambda^*(1) = 1, \Lambda^*(x) = 1.5, \Lambda^*(y) = 1, \Lambda^*(x^2) = 2.5, \Lambda^*(xy) = 1.5, \Lambda^*(y^2) = 1, \Lambda^*(xy^2) = 1.5, \Lambda^*(x^2y) = 2.5, \Lambda^*(x^3) = 4.5, \Lambda^*(x^2y^2) = 2.5, \Lambda^*(x^3y) = 4.5, \Lambda^*(x^4) = 8.5, \Lambda^*(x^4y) = 4.5, \Lambda^*(x^3y^2) = 8.5, \Lambda^*(x^4y^2) = 8.5$.
The minimum is $\Lambda^*(f) = 0$.

5. Decomposition

To be able to compute the minimizer points from an optimal linear form, we need to detect when the infimum is reached. In this section, we describe new criterion to check when the kernel of a truncated Hankel operator associated to an optimal linear form for f yields the generators of the minimizer ideal. It involves the flat extension theorem of (Laurent and Mourrain, 2009) and applies to polynomial optimization problems where the minimizer ideal I_{min} is zero-dimensional. At the end of this section we verify the flat extension property in our running example.

5.1. Flat extension criterion

Definition 5.1. Given vector subspaces $E_0 \subset E \subset \mathbb{R}[\mathbf{x}]$ and $\Lambda \in \langle E \cdot E \rangle^*$, H_Λ^E is said to be a *flat extension* of its restriction $H_\Lambda^{E_0}$ if $\operatorname{rank} H_\Lambda^E = \operatorname{rank} H_\Lambda^{E_0}$.

We recall here a result from (Laurent and Mourrain, 2009), which gives a rank condition for the existence of a flat extension of a truncated Hankel operator [1].

[1] In (Laurent and Mourrain, 2009), it is stated with a vector space spanned by a monomial set connected to 1, but its extension to vector spaces connected to 1 is straightforward.

Theorem 5.2. *Let $V \subset E \subset \mathbb{R}[\mathbf{x}]$ be vector spaces connected to 1 with $V^+ \subset E$ and let $\Lambda \in \langle E \cdot E \rangle^*$. Assume that* $\operatorname{rank} H_\Lambda^E = \operatorname{rank} H_\Lambda^V = \dim V$. *Then there exists a (unique) linear form $\tilde{\Lambda} \in \mathbb{R}[\mathbf{x}]^*$ which extends Λ, i.e., $\tilde{\Lambda}(p) = \Lambda(p)$ for all $p \in \langle E \cdot E \rangle$, satisfying* $\operatorname{rank} H_{\tilde{\Lambda}} = \operatorname{rank} H_\Lambda^E$. *Moreover, we have* $\ker H_{\tilde{\Lambda}} = (\ker H_\Lambda^E)$.

In other words, the condition $\operatorname{rank} H_\Lambda^E = \operatorname{rank} H_\Lambda^V = \dim V$ implies that the truncated Hankel operator H_Λ^E has a (unique) flat extension to a (full) Hankel operator $H_{\tilde{\Lambda}}$ defined on $\mathbb{R}[\mathbf{x}]$.

Theorem 5.3. *Let $V \subset E \subset \mathbb{R}[\mathbf{x}]$ be finite dimensional vector spaces connected to* 1 *with $V^+ \subset E$, $G^0 \cdot V \subset \langle E \cdot E \rangle$, $G^+ \cdot V \cdot V \subset \langle E \cdot E \rangle$.*

Let $\Lambda \in \mathcal{L}_{E,G}$ such that $\operatorname{rank} H_\Lambda^E = \operatorname{rank} H_\Lambda^V = \dim V$. *Then there exists a linear form $\tilde{\Lambda} \in \mathbb{R}[\mathbf{x}]^*$ which extends Λ and is supported on points of $\mathcal{S}(G)$ with positive weights:*

$$\tilde{\Lambda} = \sum_{i=1}^{r} \omega_i \mathbf{1}_{\xi_i} \text{ with } \omega_i > 0, \xi_i \in \mathcal{S}(G).$$

Moreover, $(\ker H_\Lambda^E) = \mathcal{I}(\xi_1, \ldots, \xi_r)$.

Proof. As $\operatorname{rank} H_\Lambda^E = \operatorname{rank} H_\Lambda^V = \dim V$, Theorem 5.2 implies that there exists a (unique) linear function $\tilde{\Lambda} \in \mathbb{R}[\mathbf{x}]^*$ which extends Λ. As $\operatorname{rank} H_{\tilde{\Lambda}} = \operatorname{rank} H_\Lambda^V = |V|$ and $\ker H_{\tilde{\Lambda}} = (\ker H_\Lambda^E)$, any polynomial $p \in \mathbb{R}[\mathbf{x}]$ can be reduced modulo $\ker H_{\tilde{\Lambda}}$ to a polynomial $b \in V$ so that $p - b \in \ker H_{\tilde{\Lambda}}$. Then $\tilde{\Lambda}(p^2) = \tilde{\Lambda}(b^2) = \Lambda(b^2) \geq 0$ since $\Lambda \in \mathcal{L}_{E,G}$. By Theorem 3.14 of (Lasserre et al., 2012), $\tilde{\Lambda}$ has a decomposition of the form $\tilde{\Lambda} = \sum_{i=1}^{r} \omega_i \mathbf{1}_{\xi_i}$ with $\omega_i > 0$ and $\xi_i \in \mathbb{R}^n$.

By Lemma 3.5 of (Lasserre et al., 2012), V is isomorphic to $\mathbb{R}[\mathbf{x}]/\mathcal{I}(\xi_1, \ldots, \xi_r)$ and there exist (interpolation) polynomials $b_1, \ldots, b_r \in V$ satisfying $b_i(\xi_j) = 1$ if $i = j$ and $b_i(\xi_j) = 0$ otherwise. We deduce that for $i = 1, \ldots, r$ and for all elements $g \in G^0$,

$$\Lambda(b_i g) = 0 = \tilde{\Lambda}(b_i g) = \omega_i g(\xi_i).$$

As $\omega_i > 0$ then $g(\xi_i) = 0$. Similarly, for all $h \in G^+$,

$$\Lambda(b_i^2 h) = \tilde{\Lambda}(b_i^2 h) = \omega_i h(\xi_i) \geq 0$$

and $h(\xi_i) \geq 0$, hence $\xi_i \in \mathcal{S}(G)$.

By Theorem 3.14 of (Lasserre et al., 2012) and Theorem 5.2, we also have $\ker H_{\tilde{\Lambda}} = \mathcal{I}(\xi_1, \ldots, \xi_r) = (\ker H_\Lambda^E)$. □

This theorem applied to an optimal linear form Λ^* for f gives a convergence certificate to check when the infimum f^* is reached and when a generating family of the minimizer ideal is obtained. It generalizes the flat truncation certificate given in (Nie, 2012). As we will see in the experiments, it allows to detect more efficiently when the infimum is reached. Notice that if the test is satisfied, necessarily I_{min} is zero-dimensional.

5.2. Flat extension algorithm

In this section, we describe a new algorithm to check the flat extension property for a linear form for which some moments are known.

Let E be a finite dimensional subspace of $\mathbb{R}[\mathbf{x}]$ connected to 1 and let Λ^* be a linear form defined on $\langle E \cdot E \rangle$ given by its "moments" $\Lambda^*(e_i) := \Lambda_i^*$, where $e_1, \ldots, e_s$ is a basis of

$\langle E \cdot E \rangle$ (for instance a monomial basis). In the context of global polynomial optimization that we consider here, this linear form is an optimal linear form for f (see Section 4) computed by SDP.

We define the linear functional Λ^* from its moments as $\Lambda^* : p = \sum_{i=1}^{s} p_i e_i \in \langle E \cdot E \rangle \mapsto \sum_{i=1}^{s} p_i \Lambda_i$ and the corresponding inner product:

$$\begin{aligned} & E \times E \to \mathbb{R} \\ & (p, q) \mapsto \langle p, q \rangle_* := \Lambda^*(p\, q) \end{aligned} \tag{6}$$

To check the flat extension property, we are going to inductively define vector spaces V_i as follows. Start with $V_0 = \langle 1 \rangle$. Suppose V_i is known and compute a vector space L_i of maximal dimension in V_i^+ such that L_i is orthogonal to V_i: $\langle L_i, V_i \rangle_* = 0$ and $L_i \cap \ker H_{\Lambda^*}^{V_i^+} = \{0\}$. Then we define $V_{i+1} = V_i + L_i$.

Suppose that $b_1, \ldots, b_{r_i}$ is an orthogonal basis of V_i: $\langle b_i, b_j \rangle_* = 0$ if $i \neq j$ and $\langle b_i, b_i \rangle_* \neq 0$. Then L_i can be constructed as follows: Compute the vectors

$$b_{i,j} = x_j b_i - \sum_{k=1}^{r_i} \frac{\langle x_j b_i, b_k \rangle_*}{\langle b_k, b_k \rangle_*} b_k,$$

generating $V_i^{\perp}$ in V_i^+ and extract a maximal orthogonal family $b_{r_i+1}, \ldots, b_{r_i+s}$ for the inner product $\langle ., . \rangle_*$, that form a basis of L_i. This can be done for instance by computing a QR decomposition of the matrix $[\langle b_{i,j}, b_{i',j'} \rangle_*]_{1 \leqslant i,i' \leqslant r_i, 1 \leqslant j,j' \leqslant n}$. The process can be repeated until either

- $V_i^+ \not\subset E$ and the algorithm will stop and return `failed`,
- or $L_i = \{0\}$ and $V_i^+ = V_i \oplus \ker H_{\Lambda^*}^{V_i^+}$. In this case, the algorithm stops with `success`.

Here is the complete description of the algorithm:

Algorithm 5.1: DECOMPOSITION

Input: a vector space E connected to 1 and a linear form $\Lambda^* \in \langle E \cdot E \rangle^*$.
Output: `failed` or `success` with
- a basis $B = \{b_1, \ldots, b_r\} \subset \mathbb{R}[\mathbf{x}]$;
- the relations $x_k b_j - \sum_{i=1}^{r} \frac{\langle x_k b_j, b_i \rangle_*}{\langle b_i, b_i \rangle_*} b_i$, $j = 1 \ldots r$ $k = 1 \ldots n$.

Begin

(1) Take $B := \{1\}$; $s := 1$; $r := 1$;
(2) While $s > 0$ and $B^+ \subset E$ do
  (a) compute $b_{j,k} := x_k b_j - \sum_{i=1}^{r} \frac{\langle x_k b_j, b_i \rangle_*}{\langle b_i, b_i \rangle_*} b_i$ for $j = 1, \ldots, r$, $k = 1, \ldots, n$;
  (b) compute a maximal subset $B' = \{b'_1, \ldots, b'_s\}$ of $\langle b_{j,k} \rangle$ of orthogonal vectors for the inner product $\langle ., . \rangle_*$ and let $B := B \cup B'$, $s = \mid B' \mid$ and $r += s$;
(3) If $B^+ \not\subset E$ then return `failed`
  else $(s = 0)$ return `success`.

End

Let us describe the computation performed on the moment matrix, during the main loop of the algorithm. At each step, the moment matrix of Λ^* on V_i^+ is of the form

$$H_{\Lambda^*}^{V_i^+} = \left[\begin{array}{c|c} H_{\Lambda^*}^{B_i, B_i} & H_{\Lambda^*}^{B_i, \partial B_i} \\ \hline H_{\Lambda^*}^{\partial B_i, B_i} & H_{\Lambda^*}^{\partial B_i, \partial B_i} \end{array} \right]$$

where ∂B_i is a subset of $\{b_{i,j}\}$ such that $B_i \cup \partial B_i$ is a basis of $\langle B_i^+\rangle$. By construction, the matrix $H_{\Lambda^*}^{B_i,B_i}$ is diagonal since B_i is orthogonal for $\langle\cdot,\cdot\rangle_*$. As the polynomials $b_{i,j}$ are orthogonal to B_i, we have $H_{\Lambda^*}^{B_i,\partial B_i} = H_{\Lambda^*}^{\partial B_i,B_i} = 0$. If $H_{\Lambda^*}^{\partial B_i,\partial B_i} = 0$ then the algorithm stops with `success` and all the elements $b_{i,j}$ are in the kernel of $H_{\Lambda^*}^{B_i,B_i}$. Otherwise an orthogonal basis $b'_1,\dots,b'_s$ is extracted. It can then be completed in a basis of $\langle b_{i,j}\rangle$ so that the matrix $H_{\Lambda^*}^{\partial B_i,\partial B_i}$ in this basis is diagonal with zero entries after the $(s+1)^{th}$ index. In the next loop of the algorithm, the basis B_{i+1} contains the maximal orthogonal family $b'_1,\dots,b'_s$ so that the matrix $H_{\Lambda^*}^{B_{i+1},B_{i+1}}$ remains diagonal and invertible.

Proposition 5.4. *Let $\Lambda^* \in \mathcal{L}_{E,G}$ be optimal for f. If Algorithm 5.1 applied to Λ^* and E stops with* `success`*, then*

(1) there exists a linear form $\tilde\Lambda \in \mathbb{R}[\mathbf{x}]^*$ which extends Λ^* and is supported on points in $\mathcal{S}(G)$ with positive weights:

$$\tilde\Lambda = \sum_{i=1}^{r} \omega_i \mathbf{1}_{\xi_i} \text{ with } \omega_i > 0, \xi_i \in \mathbb{R}^n.$$

(2) $B = \{b_1,\dots,b_r\}$ is a basis of $\mathcal{A}_{\tilde\Lambda} = \mathbb{R}[\mathbf{x}]/I_{\tilde\Lambda}$ where $I_{\tilde\Lambda} = \ker H_{\tilde\Lambda}$,

(3) $x_k b_j - \sum_{i=1}^r \frac{\langle x_k b_j, b_i\rangle_*}{\langle b_i,b_i\rangle_*} b_i$, $j=1,\dots,r$, $k=1,\dots,n$ are generators of $I_{\tilde\Lambda} = \mathcal{I}(\xi_1,\dots,\xi_r)$,

(4) $f^{\mu}_{E,G} = f^*$,

(5) $V_{min} = \{\xi_1,\dots,\xi_r\}$.

Proof. When the algorithm terminates with `success`, the set B is such that $\operatorname{rank} H_{\Lambda^*}^{B^+} = \operatorname{rank} H_{\Lambda^*}^{B} = |B|$. By Theorem 5.3, there exists a linear form $\tilde\Lambda \in \mathbb{R}[\mathbf{x}]^*$ extends Λ^* and is supported on points in $\mathcal{S}(G)$ with positive weights:

$$\tilde\Lambda = \sum_{i=1}^{r} \omega_i \mathbf{1}_{\xi_i} \text{ with } \omega_i > 0, \xi_i \in \mathcal{S}(G).$$

This implies that $\mathcal{A}_{\tilde\Lambda}$ is of dimension r and that $I_{\tilde\Lambda} = \mathcal{I}(\xi_1,\dots,\xi_r)$. As $H_{\Lambda^*}^{B}$ is invertible, B is a basis of $\mathcal{A}_{\tilde\Lambda}$ which proves the second point.

Let K be the set of polynomials $x_j b_i - \sum_{k=1}^r \frac{\langle x_j b_i, b_k\rangle_*}{\langle b_k,b_k\rangle_*} b_k$. If the algorithm terminates with `success`, we have $\ker H_{\Lambda^*}^{B^+} = \langle K\rangle$ and by Theorem 5.3, we deduce that $(K) = (\ker H_{\Lambda^*}^{B^+}) = I_{\tilde\Lambda}$, which proves the third point.

As $\tilde\Lambda(1) = 1$, we have $\sum_{i=1}^r w_i = 1$ and

$$\tilde\Lambda(f) = \sum_{i=1}^{r} \omega_i f(\xi_i) \geq f^*$$

since $\xi_i \in \mathcal{S}(G)$ and $f(\xi_i) \geq f^*$. The relation $f^{\mu}_{E,G} \leq f^*$ implies that $f(\xi_i) = f^*$ for $i = 1,\dots,r$ and the fourth point is true: $f^{\mu}_{E,G} = f^*$.

As $f(\xi_i) = f^*$ for $i=1,\dots,r$, we have $\{\xi_1,\dots,\xi_r\} \subset V_{min}$. By Theorem 4.2, the polynomials of K are in I_{min} so that $V_{min} \subset \mathcal{V}(K) = \{\xi_1,\dots,\xi_r\}$. This shows that $V_{min} = \{\xi_1,\dots,\xi_r\}$ and concludes the proof of this proposition $\square$

Example 5.5. We apply Algorithm 5.1 to our running example. A solution of the SDP problem output by Algorithm 4.1 is:

$\Lambda^*(1) = 1, \Lambda^*(x) = 1.5, \Lambda^*(y) = 1, \Lambda^*(x^2) = 2.5, \Lambda^*(xy) = 1.5, \Lambda^*(y^2) = 1, \Lambda^*(xy^2) = 1.5, \Lambda^*(x^2y) = 2.5, \Lambda^*(x^3) = 4.5, \Lambda^*(x^2y^2) = 2.5, \Lambda^*(x^3y) = 4.5, \Lambda^*(x^4) = 8.5, \Lambda^*(x^4y) = 4.5, \Lambda^*(x^3y^2) = 8.5, \Lambda^*(x^4y^2) = 8.5$

We verify the flat extension criterion for $\mathbb{R}[\mathbf{x}]_3$.

- $B_0 = \{1\},\ \partial B_0 = \{x, y\},\ B_0^+ = \{1, x, y\}$

$$H_{\Lambda^*}^{B_0^+} = \begin{pmatrix} 1 & 1.5 & 1 \\ 1.5 & 2.5 & 1.5 \\ 1 & 1.5 & 1 \end{pmatrix} \longrightarrow H_{\Lambda^*}^{\{1, x-1.5, y-1\}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0.25 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$\text{rank } H_{\Lambda^*}^{B_0^+} = 2,$$

$$\{y-1\} \in \ker H_{\Lambda^*}^{B_0^+},\ \{x-1.5\} \perp B_0 \textit{ and } \{x-1.5\} \notin \ker H_{\Lambda^*}^{B_0^+},$$

$$L_0 = \{x - 1.5\}.$$

- $B_1 = B_0 \cup L_0 = \{1, x-1.5\},\ \partial B_1 = \{y, x^2 - 1.5x, xy - 1.5y\}$,
  $B_1^+ = \{1, x - 1.5, y, x^2 - 1.5x, xy - 1.5y\}$

$$H_{\Lambda^*}^{B_1^+} = \begin{pmatrix} 1 & 0 & 1 & 0.25 & 0 \\ 0 & 0.25 & 0 & 0.375 & 0.25 \\ 1 & 0 & 1 & 0.25 & 0 \\ 0.25 & 0.375 & 0.25 & 0.625 & 0.375 \\ 0 & 0.25 & 0 & 0.375 & 0.25 \end{pmatrix} \longrightarrow H_{\Lambda^*}^{\tilde{B}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

where $\tilde{B} = \{1, x - 1.5, y - 1, x^2 - 3x + 2, xy - 1.5y - x + 1.5\}$

rank $H_{\Lambda^*}^{B_1^+} = 2,\ \textit{and } L_1 = \{0\}.$

The algorithm stops with success, the flat extension property is satisfied,

$$\ker\ H_{\Lambda^*}^{B_1^+} = \{y - 1, x^2 - 3x + 2, xy - 1.5y - x + 1.5\}$$

and

$$B_1 = \{1, x - 1.5\}.$$

6. Minimizers

In this section we tackle the computation of the minimizer points, once Algorithm 5.1 stops with `success` for $\Lambda^* \in \mathcal{L}_{E,G}$ optimal for f. The minimizer points can be computed from the eigenvalues of the multiplication operators $M_k : a \in \mathcal{A}_{min} \mapsto x_k a \in \mathcal{A}_{min}$ for $k = 1, \ldots, n$ where $\mathcal{A}_{min} = \mathbb{R}[\mathbf{x}]/I_{min}$ and $I_{min} = I_{\tilde{\Lambda}} = \mathcal{I}(\xi_1, \ldots, \xi_r)$. At the end of this section we compute the minimizers for our running example.

Proposition 6.1. *The matrix of M_k in the basis B of $\mathcal{A}_{min}$ is $[M_k] = (\frac{\Lambda^*(x_k\, b_i\, b_j)}{\Lambda^*(b_i\, b_i)})_{1\le i,j\le r}$. The operators M_k, $k = 1 \ldots n$ have r common eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$ which satisfy $M_k \mathbf{u}_i = \xi_{i,k}\mathbf{u}_i$, with $\xi_{i,k}$ the k^{th} coordinate of the minimizer point $\xi_i = (\xi_{i,1}, \ldots, \xi_{i,n}) \in S$.*

Proof. By Proposition 5.4 and by definition of the inner-product (6), $B = \{b_1, \ldots, b_r\}$ is a basis of $\mathcal{A}_{\tilde{\Lambda}}$ and

$$x_k b_j \equiv \sum_{i=1}^{r} \frac{\Lambda^*(x_k\, b_i\, b_j)}{\Lambda^*(b_i\, b_i)} b_i \mod I_{min},$$

for $j = 1 \ldots r$, $k = 1 \ldots n$.
This yields the matrix of the operator M_k in the basis B: $[M_k] = (\frac{\Lambda^*(x_k\, b_i\, b_j)}{\Lambda^*(b_i\, b_i)})_{1\le i,j\le r}$.

As the roots of I_{min} are simple, by (Elkadi and Mourrain, 2007)[Theorem 4.23] the eigenvectors of all M_k, $k = 1 \ldots n$ are the so-called idempotents $\mathbf{u}_1, \ldots, \mathbf{u}_r$ of $\mathcal{A}_{min}$ and the corresponding eigenvalues are $\xi_{1,k}, \ldots, \xi_{r,k}$. □

Algorithm 6.1: Minimizer points

Input: B and the output relations from Algorithm 5.1.
Output: the minimizer points $\xi_i = (\xi_{i,1}, \ldots, \xi_{i,n})$, $i = 1 \ldots r$.
Begin

(1) Compute the matrices $[M_k] = (\frac{\Lambda^*(x_k\, b_i\, b_j)}{\Lambda^*(b_i\, b_i)})_{1\le i,j\le r}$.
(2) For a generic choice of $l_1, \ldots, l_n \in \mathbb{R}$, compute the eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$ of $l_1[M_1] + \cdots + l_n[M_n]$.
(3) Compute $\xi_{i,k} \in \mathbb{R}$ such that $M_k \mathbf{u}_i = \xi_{i,k}\mathbf{u}_i$.

End

Now we compute the minimizer points of our running example 2.1.

Example 6.2. With the basis $B = \{1, x - 1.5\}$ and the kernel ker $H_{\Lambda}^{B^+} = \langle y - 1, x^2 - 3x + 2, xy - 1.5y - x + 1.5\rangle$ produced by Algorithm 5.5, we can compute the multiplication matrices.

$$M_x^{B=\{1,x-1.5\}} = \begin{pmatrix} 1.5 & 0.25 \\ 1 & 1.5 \end{pmatrix} \longrightarrow \begin{cases} x \times 1 = 1.5 \cdot 1 + 1 \cdot (x - 1.5) \\ x \times (x - 1.5) = 0.25 \cdot 1 + 1.5 \cdot (x - 1.5) \end{cases}$$

$$M_y^{B=\{1,x-1.5\}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \longrightarrow \begin{cases} y \times 1 = 1 \cdot 1 + 0 \cdot (x - 1.5) \\ y \times (x - 1.5) = 0 \cdot 1 + 1 \cdot (x - 1.5) \end{cases}$$

We take a linear combination of these matrices

$$M = M_x^B + M_y^B = \begin{pmatrix} 2.5 & 0.25 \\ 1 & 2.5 \end{pmatrix}$$

and compute its eigenvalues $\lambda_1 = 2, \lambda_2 = 3$ and its eigenvectors:

$$M \cdot u_1 = \lambda_1 \cdot u_1 \to u_1^T = (-0.5, 1);\ M \cdot u_2 = \lambda_2 \cdot u_2 \to u_2^T = (0.5, 1)$$

From these eigenvectors, we compute the eigenvalues associated to each multiplication matrix M_x^B, M_y^B. Each computed eigenvalue corresponds to a coordinate of the corresponding minimizer point as we have seen in Proposition 6.1:

$$M_x^B \cdot u_1^T = x_1 \cdot u_1^T \to x_1 = 1;\ M_x^B \cdot u_2^T = x_2 \cdot u_2^T \to x_2 = 2$$

$$M_y^B \cdot u_1^T = y_1 \cdot u_1^T \to y_1 = 1;\ M_y^B \cdot u_2^T = y_2 \cdot u_2^T \to y_2 = 1$$

We recover the minimizer points $(1, 1)$ and $(2, 1)$.

7. Main algorithm

In this section we describe the algorithm to compute the infimum of a polynomial on S and the minimizer points when the minimizer ideal is zero-dimensional. It can be seen as a type of border basis algorithm, in which in the main loop we compute the optimal linear form (section 4), we then check when the minimun is reached (section 5) and finally we compute the minimizer points (section 6). This algorithm is closely connected to the real radical border basis algorithm presented in (Lasserre et al., 2012).

Algorithm 7.1: MINIMIZATION OF f ON S

Input: A real polynomial function f and a set of constraints $\mathbf{g} \subset \mathbb{R}[\mathbf{x}]$ with V_{min} non-empty finite.

Output: the minimum $f^* = f^*_{G_t,B_t}$, the minimizer points $V_{min} = V$, $I_{min} = (K)$ and B' such that K is a border basis for B'.

Begin

(1) Take $t = max(\lceil \frac{deg(f)}{2} \rceil, d^0, d^+)$ where $d^0 = max_{g^0 \in \mathbf{g}^0}(\lceil \frac{deg(g^0)}{2} \rceil), d^+ = max_{g^+ \in \mathbf{g}^+}(\lceil \frac{deg(g^+)}{2} \rceil)$

(2) Compute the graded border basis F_{2t} of $\mathbf{g}^0$ for B in degree $2t$.

(3) Let B_t be the set of monomials in B of degree $\leq t$.

(4) Let G_t be the set of constraints such that $G_t^0 = \{m - \pi_{B_t,F_{2t}}(m),\ m \in B_t \cdot B_t\}$ and $G^+ = \pi_{B_t,F_{2t}}(g^+)$

(5) $[f^*_{G_t,B_t}, \Lambda^*] :=$ OPTIMAL LINEAR FORM(f, B_t, G_t).

(6) $[c, B', K] :=$ DECOMPOSITION(Λ^*, B_t) where $c =$`failed`, $B' = \emptyset, K = \emptyset$ or $c =$`success`, B' is the basis and K is the set of the relations.

(7) if $c =$`success` then V=MINIMIZER POINTS(B', K) else go to step 2 with $t := t + 1$.

End

---

8. Finite convergence

In this section we analyse cases for which an exact relaxation can be constructed.

Our approach to compute the minimizer points relies on the fact that the border basis relaxation is exact.

By Proposition 3.6, the reduced border basis relaxation is exact if and only if the corresponding full moment matrix relaxation is exact.

Despite the full moment matrix relaxation is not always exact, it is possible to add constraints so that the relaxation becomes exact.

In (Abril Bucero and Mourrain, 2013), a general strategy to construct exact SDP relaxation hierarchies and to compute the minimizer ideal is described. It applies to the following problems:

Global optimization. Consider the case $n_1 = n_2 = 0$ with $f^* = \inf_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$ reached at a point of $\mathbb{R}^n$. Taking G such that $G^0 = \{\frac{\partial f}{\partial_{x_1}}, \ldots, \frac{\partial f}{\partial_{x_n}}\}$ and $G^+ = \emptyset$, the relaxation associated to the sequence $\mathcal{L}_{t,G}$ is exact and yields I_{min} (see (Nie et al., 2006; Abril Bucero and Mourrain, 2013)). If I_{min} is finite then the border basis relaxation yields the minimizer points and the corresponding border basis.

Regular case.

We say that $\mathbf{g} = (g_1^0, \ldots g_{n_1}^0;\, g_1^+, \ldots, g_{n_2}^+)$ is regular if for all points $\mathbf{x} \in \mathcal{S}(\mathbf{g})$ with $\{j_1, \ldots, j_k\} = \{j \in [1, n_2] \mid g_j^+(\mathbf{x}) = 0\}$, the vectors $\nabla g_1^0(\mathbf{x}), \ldots, \nabla g_{n_1}^0(\mathbf{x})$, $\nabla g_{j_1}^+(\mathbf{x}), \ldots$, $\nabla g_{j_k}^+(\mathbf{x})$ are linearly independent.

For $\nu = \{j_1, \ldots, j_k\} \subset [0, n_2]$ with $|\nu| \le n - n_1$, let

$$\begin{aligned} A_\nu &= [\nabla f, \nabla g_1^0, \ldots, \nabla g_{n_1}^0, \nabla g_{j_1}^+, \ldots, \nabla g_{j_k}^+] \\ \Delta_\nu &= \det(A_\nu A_\nu^T) \\ g_\nu &= \Delta_\nu \prod_{j\notin\nu} g_j^+. \end{aligned}$$

Let $G \subset \mathbb{R}[\mathbf{x}]$ be the set of constraints such that $G^0 = \mathbf{g}^0 \cup \{g_\nu \mid \nu \subset [0, n_2], |\nu| \le n - n_1\}$. Then the relaxation associated to the preordering sequence $\mathcal{L}_{t,G}^\star$ is exact and yields I_{min} (see (Ha and Pham, 2010; Abril Bucero and Mourrain, 2013) or (Nie, 2011) for $\mathbb{C}$-regularity and constraints G^0 that involve minors of A_ν).

If I_{min} is non-empty and finite then the border basis relaxation (2) yields the points V_{min} and the border basis of I_{min}.

Boundary Hessian Conditions. If f and $\mathbf{g}$ satisfies the so-called Boundary Hessian Conditions then $f - f^* \in \mathcal{Q}_{t,g}$ and the relaxation associated to $\mathcal{L}_{t,\mathbf{g}}$ is exact and yields I_{min} (see (Marshall, 2009)). If moreover I_{min} is finite then the border basis relaxation yields the points V_{min} and the corresponding border basis of I_{min}.

$\mathbf{g}^+$-radical computation. If we optimize $f = 0$ on the set $S = \mathcal{S}(\mathbf{g})$, then all the points of S are minimizer points, $V_{min} = S$ and by the Positivstellensatz, I_{min} is equal to

$$\sqrt[\mathbf{g}^+]{\mathbf{g}^0} = \{p \in \mathbb{R}[\mathbf{x}] \mid \exists m \in \mathbb{N}\ s.t.\ p^{2m} + q = 0, q \in \mathcal{P}_{\mathbb{R}[\mathbf{x}],\mathbf{g}}\}.$$

Here again, the preordering sequence $\mathcal{L}_{t,\mathbf{g}}^\star$ is exact. If we assume that $S = \mathcal{S}(\mathbf{g})$ is finite, then the corresponding border basis relaxation yields the points of S and the generators of $\sqrt[\mathbf{g}^+]{\mathbf{g}^0}$. See also (Lasserre et al., 2009, 2012) for zero dimensional real radical computation and (Ma et al., 2013).

9. Performance

In this section, we analyse the practical behavior of Algorithm 7.1. In all the examples the minimizer ideal is zero-dimensional hence our algorithm stops in a finite number of steps and yields the minimizer points and generators of the minimizer ideal.

The implementation of the previous algorithm has been performed using the BORDERBASIX[2] package of the MATHEMAGIX[3] software, which provides a C++ implementation of the border basis algorithm of (Mourrain and Trébuchet, 2012).

For the computation of border basis, we use a choice function that is tolerant to numerical inestability i.e. a choice function that chooses as leading monomial a monomial whose coefficient is maximal among the choosable monomials as described in (Mourrain and Trébuchet, 2008).

The Semi-Definite Programming problems are solved using SDPA, SDPA-GMP[4], CSDP and MOSEK[5] software. For the link with SDPA,CSDP and SDPA-GMP we use a file interface. In the case of MOSEK, we use the distributed library.

Once we have computed the moment matrix, we call the Decomposition Algorithm which is available in the BORDERBASIX package.

The minimizer points are computed from the eigenvalues of the multiplication matrices. This is performed using Lapack routines.

Experiments are made on an Intel Core i5 2.40GHz.

In Table 1, we compare our algorithm 7.1 (*bbr*) with the full moment matrix relaxation algorithm (*fmr*) inside the same environment. This latter (implemented by ourselves in C++ inside the BORDERBASIX package) reproduces the algorithm described in (Lasserre, 2009), which is also implemented in the package GLOPTIPOLY of MATLAB developed by D. Henrion and J.B. Lasserre. In this table, we record the problem name or the source of the problem, the number of decision variables (v), the number of inequality and equality constraints, we mark in parenthesis the number of equality constraints (c), the maximum degree of the constraints and of the polynomial to minimize (d), the number of minimizer points (sol). For the two algorithms *bbr* and *fmr* we report the total CPU time in seconds using MOSEK (t), the order of the relaxation (o), the number of parameters of the SDP problem (p) and the size of the moment matrices (s). The first part of the table contains examples of positive polynomials, which are not sums of squares. New equality constraints are added following (Abril Bucero and Mourrain, 2013) to compute the minimizer points in the examples marked with $\diamond$. When there are equality constraints, the border basis computation reduces the size of the moment matrices, as well as the localization matrices associated to the inequalities. This speeds up the SDP computation as shown the examples Ex 1.4.8, Ex 2.1.8, Ex 2.1.9 and simplex. In the case where there are only inequalities, the size of the moment matrices and number of parameters do not change but once the optimal linear form is computed using the SDP solver MOSEK, the DECOMPOSITION algorithm which computes the minimizers is more efficient and quicker than the reconstruction algorithm used in the full moment matrix relaxation approach. The performance is not the only issue: numerical problems can also occur due to the bigger size of the moment matrices in the flat extension test and the reconstruction of minimizer points. Such examples where the *fmr* algorithm fails are marked with *. In these three problems, there is not a big enough gap between the singular values to determine correctly the numerical rank and the flat extension property cannot be verified. The examples that GLOPTIPOLY cannot treat due to the high number

[2] http://www-sop.inria.fr/teams/galaad/software/bbx/

[3] www.mathemagix.org

[4] http://sdpa.sourceforge.net

[5] http://www.mosek.com

of variables (Lasserre, 2009) are marked with **. We can treat three of this examples (with *fmr*) because as we said *fmr* is implemented in C++ so it is more efficient than GLOTIPOLY, which is implemented inside MATLAB. We cannot treat the example 2.1.8 with the *fmr* algorithm due to the large number of parameters.

These experiments show that when the size of the SDP problems becomes significant, most of the time is spent during the SDP computation and the border basis time and reconstruction time are negligible. The use of MOSEK software provides a speed-up factor of 1.5 to 5 compared to the SDPA software for small examples (such as Robinson, Moztkin, Ex 3, Ex 5, Ex 2.1.1, Ex 2.1.2, Ex 2.1.4 and Ex 2.1.6). For large examples (such as Ex 2.1.3, Ex 2.1.7, Ex 2.1.8 and simplex) the improvement factors are between 10-30 times. These improvements are due to the new fast Cholesky decomposition inside of MOSEK software [6]. In all the examples, the new border basis relaxation algorithm outperforms the full moment matrix relaxation method.

In Table 2, we apply our algorithm *bbr* to find the best rank-1 and rank-2 tensor approximation for symmetric and non symmetric tensors on examples from (Nie and Wang, 2013) and (Ottaviani et al., 2013). For best rank-1 approximation problems with several minimizers (which is the case when there are symmetries), the method proposed in (Nie and Wang, 2013) cannot certify the result and uses a local method to converge to a local extrema. We apply the global border basis relaxation algorithm to find all the minimizers for the best rank 1 approximation problem.

The last example in Table 2 is a best rank-2 tensor approximation example from the paper (Ottaviani et al., 2013). The eight solutions come from the symmetries due to the invariance of the solution set by permutation and negation of the factors.

Acknowledgements

We would like to thank Philippe Trebuchet and Matthieu Dien for their development in the BORDERBASIX package.

References

Abril Bucero, M., Mourrain, B., 2013. Exact relaxation for polynomial optimization on semi-algebraic sets, http://hal.inria.fr/hal-00846977.

Brachat, J., Comon, P., Mourrain, B., Tsigaridas, E., 2010. Symmetric tensor decomposition. Linear Algebra and Applications 433, 1851–1872.

Demmel, J., Nie, J., Powers, V., 2007. Representations of positive polynomials on non-compact semialgebraic sets via kkt ideals. Journal of Pure and Applied Algebra 209 (1), 189 – 200.

Elkadi, M., Mourrain, B., 2007. Introduction à la résolution des systèmes d'équations algébriques. Vol. 59 of Mathématiques et Applications. Springer-Verlag.

Floudas, C. A., Pardalos, P. M., Adjiman, C. S., Esposito, W. R., Gumus, Z. H., Harding, S. T., Klepeis, J. L., Meyer, C. A., Schweiger, C. A., 1999. Handbook of Test Problems in Local and Global Optimization. Kluwer Academic Publishers.

Giesbrecht, M., Labahn, G., Lee, W.-S., Aug. 2009. Symbolic-numeric sparse interpolation of multivariate polynomials. J. Symb. Comput. 44 (8), 943–959.

[6] http://www.mosek.com

problem	v	c	d	sol	o_{bbr}	p_{bbr}	s_{bbr}	t_{bbr}	o_{fmr}	p_{fmr}	s_{fmr}	t_{fmr}
⋄ Robinson	2	0	6	8	4	20	15	0.07	7	119	36	*
⋄ Motzkin	2	0	6	4	4	25	15	0.060	9	189	55	*
⋄ Motzkin perturbed	3	1	6	1	5	127	35	0.18	5	286	56	8.01
⋄ L'01, Ex. 1	2	0	4	1	2	8	6	0.020	2	14	6	0.035
⋄ L'01, Ex. 2	2	0	4	1	2	8	6	0.020	2	14	6	0.026
⋄ L'01, Ex. 3	2	0	6	4	4	25	15	0.057	8	152	45	*
L'01, Ex. 5	2	3	2	3	2	14	6	0.032	2	14	6	0.045
F, Ex. 4.1.4	1	2	4	2	2	4	3	0.016	2	4	3	0.023
F, Ex. 4.1.6	1	2	6	2	3	6	4	0.018	3	6	4	0.020
F, Ex. 4.1.7	1	2	4	1	2	4	3	0.017	2	4	3	0.020
F, Ex. 4.1.8	2	5(1)	4	1	2	13	6	0.021	2	14	6	0.12
F, Ex. 2.1.1	5	11	2	1	3	461	56	3.10	3	461	56	3.12
F, Ex. 2.1.2	6	13	2	1	2	209	26	0.32	2	209	26	0.36
F, Ex. 2.1.3	13	35	2	1	2	2379	78	19.68	2	2379	78	25.60
F, Ex. 2.1.4	6	15	2	1	2	209	26	0.30	2	209	26	0.33
F, Ex. 2.1.5	10	31	2	1	2	1000	66	9.15	2	1000	66	9.7
F, Ex. 2.1.6	10	25	2	1	2	1000	66	3.6	2	1000	66	4.17
**F, Ex. 2.1.7(1)	20	30	2	1	2	10625	231	730.24	2	10625	231	1089.31
** F, Ex. 2.1.7(5)	20	30	2	1	2	10625	231	747.94	2	10625	231	1125.27
** F, Ex. 2.1.8	24	58(10)	2	1	2	3875	136	311.54	2	20474	325	>14h
F, Ex. 2.1.9	10	11(1)	2	1	2	714	44	0.62	2	1000	55	1.67
** simplex	15	16(1)	2	1	2	3059	120	15.30	2	3875	136	47.50

Table 1. Examples from F-(Floudas et al., 1999)), L'01-(Lasserre, 2001).

Greuet, A., Safey El Din, M., 2011. Deciding reachability of the infimum of a multivariate polynomial. In: Proceedings of the 36th international symposium on Symbolic and algebraic computation. ISSAC '11. ACM, New York, NY, USA, pp. 131–138.

Ha, H. V., Pham, T., 2010. Representation of positive polynomials and optimization on noncompact semialgebraic sets. SIAM Journal on Optimization 20 (6), 3082–3103.

Henrion, D., Lasserre., J., 2005. Positive Polynomials in Control. Lectures Notes in Control and Information Sciences. Springer, Ch. Detecting Global Optimality and Extracting Solutions in GloptiPoly., pp. 293–310.

Lasserre., J., 2001. Global optimization with polynomials and the problem of moments. SIAM J. Optim. 11, 796–817.

Lasserre, J., 2009. Moments, positive polynomials and their applications. Imperial College Press.

problem	v	c	d	sol	o_{bbr}	p_{bbr}	s_{bbr}	$t_{bbr+msk}$
(Nie and Wang, 2013) Ex. 3.1	2	1	3	1	2	8	5	0.028
(Nie and Wang, 2013) Ex. 3.2	3	1	3	1	2	24	9	0.025
(Nie and Wang, 2013) Ex. 3.3	3	1	3	1	2	24	9	0.035
(Nie and Wang, 2013) Ex. 3.4	4	1	4	2	2	24	9	0.097
(Nie and Wang, 2013) Ex. 3.5	5	1	3	1	2	104	20	0.078
(Nie and Wang, 2013) Ex. 3.6	5	1	4	2	4	824	105	15.39
(Nie and Wang, 2013) Ex. 3.8	3	1	6	4	3	48	16	1.14
(Nie and Wang, 2013) Ex. 3.11	8	4	4	8	3	84	25	0.17
(Nie and Wang, 2013) Ex. 3.12	9	3	3	4	2	552	52	1.55
(Nie and Wang, 2013) Ex. 3.13	9	3	3	12	3	3023	190	223.27
(Ottaviani et al., 2013) Ex. 4.2	6	0	8	4	8	2340	210	59.38

Table 2. Best rank-1 and rank-2 approximation tensors

Lasserre, J., Laurent, M., Rostalski., P., 2008. Semidefinite characterization and computation of real radical ideals. Foundations of Computational Mathematics 8 (5), 607–647.

Lasserre, J., Laurent, M., Rostalski, P., 2009. A unified approach for real and complex zeros of zero-dimensional ideals. In: Putinar, M., Sullivant, S. (Eds.), Emerging Applications of Algebraic Geometry. Vol. 149. Springer, pp. 125–156.

Lasserre, J.-B., Laurent, M., Mourrain, B., Rostalski, P., Trébuchet, P., 2012. Moment matrices, border bases and real radical computation. Journal of Symbolic Computation.

Laurent, M., 2007. Semidefinite representations for finite varieties. Math. Progr 109, 1–26.

Laurent, M., Mourrain, B., July 2009. A generalized flat extension theorem for moment matrices. Arch. Math. (Basel) 93 (1), 87–98.

Ma, Y., Wang, C., L., Z., 2013. A certificate for semidefinite relaxations in computing positive dimensional real varieties, http://arxiv.org/abs/1212.4924.

Marshall, M., 2009. Representations of non-negative polynomials, degree bounds and applications to optimization. Can. J. Math. 61 (1), 205–221.

Mourrain, B., 1999. A new criterion for normal form algorithms. In: Fossorier, M., Imai, H., Lin, S., Poli, A. (Eds.), Proc. AAECC. Vol. 1719 of LNCS. Springer, Berlin, pp. 430–443.

Mourrain, B., Trébuchet, P., 2005. Generalized normal forms and polynomials system solving. In: Kauers, M. (Ed.), ISSAC: Proceedings of the ACM SIGSAM International Symposium on Symbolic and Algebraic Computation. pp. 253–260.

Mourrain, B., Trébuchet, P., 2008. Stable normal forms for polynomial system solving. Theoretical Computer Science 409 (2), 229–240.

Mourrain, B., Trébuchet, P., Jul. 2012. Border basis representation of a general quotient algebra. In: van der Hoeven, J. (Ed.), ISSAC 2012. pp. 265–272.

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

Nie, J., 2011. An exact jacobian SDP relaxation for polynomial optimization. Mathematical Programming, 1–31.

Nie, J., 2012. Certifying convergence of Lasserre's hierarchy via flat truncation. Mathematical Programming, 1–26.

Nie, J., Demmel, J., Sturmfels, B., 2006. Minimizing polynomials via sum of squares over gradient ideal. Math. Program. 106 (3), 587–606.

Nie, J., Wang, L., 2013. Semidefinite relaxations for best rank-1 tensor approximations, http://arxiv.org/abs/1308.6562v2.

Ottaviani, G., Spaenlehauer, P.-J., Sturmfels, B., 2013. Exact solutions in structured low-rank approximation, http://arxiv.org/abs/1311.2376v2.

Parrilo, P., 2003. Semidefinite programming relaxations for semialgebraic problems. Mathematical Programming Ser. B 96 (2), 293–320.

Parrilo, P., Sturmfels, B., 2003. Minimizing polynomial functions. In: Proceedings of the DIMACS Workshop on Algorithmic and Quantitative Aspects of Real Algebraic Geometry in Mathematics and Computer Science. American Mathematical Society, pp. 83–100.

Riener, C., Theobald, T., Andrén, L. J., Lasserre, J. B., Feb. 2013. Exploiting symmetries in SDP-relaxations for polynomial optimization. Math. Oper. Res. 38 (1), 122–141.

Safey El Din, M., 2008. Computing the global optimum of a multivariate polynomial over the reals. In: Proceedings of the twenty-first international symposium on Symbolic and algebraic computation. ISSAC '08. ACM, New York, NY, USA, pp. 71–78.

Shor, N., 1987. Class of global minimum bounds of polynomial functions. Cybernetics 23, 731–734.

A. Results of Best rank-1 approximation tensors

Example 3.1: Consider the tensor $\mathcal{F} \in S^3(\mathbb{R}^2)$ with entries
$\mathcal{F}_{111} = 1.5578, \mathcal{F}_{222} = 1.1226, \mathcal{F}_{112} = -2.443, \mathcal{F}_{221} = -1.0982$
We get the rank-1 tensor $\lambda \cdot u^{\otimes 3}$ with:
$\lambda = 3.11551,\ u = (0.926433, -0.376457)$ and $\| \mathcal{F} - \lambda \cdot u^{\otimes 3} \| = 3.9333$.

Example 3.2: Consider the tensor $\mathcal{F} \in S^3(\mathbb{R}^3)$ with entries
$\mathcal{F}_{111} = -0.1281, \mathcal{F}_{112} = 0.0516, \mathcal{F}_{113} = -0.0954, \mathcal{F}_{122} = -0.1958, \mathcal{F}_{123} = -0.1790,$
$\mathcal{F}_{133} = -0.2676, \mathcal{F}_{222} = 0.3251, \mathcal{F}_{223} = 0.2513, \mathcal{F}_{233} = 0.1773, \mathcal{F}_{333} = 0.0338$
We get the rank-1 tensor $\lambda \cdot u^{\otimes 3}$ with:
$\lambda = 0.87298,\ u = (-0.392192, 0.7248, 0.5664)$ and $\| \mathcal{F} - \lambda \cdot u^{\otimes 3} \| = 0.4498$.

Example 3.3: Consider the tensor $\mathcal{F} \in S^3(\mathbb{R}^3)$ with entries
$\mathcal{F}_{111} = 0.0517, \mathcal{F}_{112} = 0.3579, \mathcal{F}_{113} = 0.5298, \mathcal{F}_{122} = 0.7544, \mathcal{F}_{123} = 0.2156,$
$\mathcal{F}_{133} = 0.3612, \mathcal{F}_{222} = 0.3943, \mathcal{F}_{223} = 0.0146, \mathcal{F}_{233} = 0.6718, \mathcal{F}_{333} = 0.9723$
We get the rank-1 tensor $\lambda \cdot u^{\otimes 3}$ with:
$\lambda = 2.11102,\ u = (0.52048, 0.511264, 0.683891)$ and $\| \mathcal{F} - \lambda \cdot u^{\otimes 3} \| = 1.2672$.

Example 3.4: Consider the tensor $\mathcal{F} \in S^4(\mathbb{R}^3)$ with entries
$\mathcal{F}_{1111} = 0.2883, \mathcal{F}_{1112} = -0.0031, \mathcal{F}_{1113} = 0.1973, \mathcal{F}_{1122} = -0.2458, \mathcal{F}_{1123} = -0.2939,$
$\mathcal{F}_{1133} = 0.3847, \mathcal{F}_{1222} = 0.2972, \mathcal{F}_{1223} = 0.1862, \mathcal{F}_{1233} = 0.0919, \mathcal{F}_{1333} = -0.3619$
$\mathcal{F}_{2222} = 0.1241, \mathcal{F}_{2223} = -0.3420, \mathcal{F}_{2233} = 0.2127, \mathcal{F}_{2333} = 0.2727, \mathcal{F}_{3333} = -0.3054$
We get the rank-1 tensor $\lambda \cdot u_i^{\otimes 3}$ with:
$\lambda = -1.0960,\ u_1 = (-0.59148, 0.7467, 0.3042); u_2 = (0.59148, -0.7467, -0.3042)$ and

$\| \mathcal{F} - \lambda \cdot u_i^{\otimes 4} \| = 1.9683$.

Example 3.5: Consider the tensor $\mathcal{F} \in S^3(\mathbb{R}^5)$ with entries
$\mathcal{F}_{i_1,i_2,i_3} = \frac{(-1)^{i_1}}{i_1} + \frac{(-1)^{i_2}}{i_2} + \frac{(-1)^{i_3}}{i_3}$
We get the rank-1 tensor $\lambda \cdot u^{\otimes 3}$ with:
$\lambda = 9.9776,\ u = (-0.7313, -0.1375, -0.46737, -0.23649, -0.4146)$ and
$\| \mathcal{F} - \lambda \cdot u^{\otimes 3} \| = 5.3498$.

Example 3.6: Consider the tensor $\mathcal{F} \in S^4(\mathbb{R}^5)$ with entries
$\mathcal{F}_{i_1,i_2,i_3,i_4} = arctan((-1)^{i_1} \frac{i_1}{5}) + arctan((-1)^{i_2} \frac{i_2}{5}) + arctan((-1)^{i_3} \frac{i_3}{5}) + arctan((-1)^{i_4} \frac{i_4}{5})$
We get the rank-1 tensor $\lambda \cdot u^{\otimes 4}$ with:
$\lambda = -23.56525,\ u_1 = (0.4398, 0.2383, 0.5604, 0.1354, 0.6459)$;
$u_2 = (-0.4398, -0.2383, -0.5604, -0.1354, -0.6459)$ and
$\| \mathcal{F} - \lambda \cdot u_i^{\otimes 4} \| = 16.8501$.

Example 3.8: Consider the tensor $\mathcal{F} \in S^6(\mathbb{R}^3)$ with entries
$\mathcal{F}_{111111} = 2, \mathcal{F}_{111122} = 1/3, \mathcal{F}_{111133} = 2/5, \mathcal{F}_{112222} = 1/3, \mathcal{F}_{112233} = 1/6,$
$\mathcal{F}_{113333} = 2/5, \mathcal{F}_{222222} = 2, \mathcal{F}_{222233} = 2/5, \mathcal{F}_{223333} = 2/5, \mathcal{F}_{333333} = 1$
We get the rank-1 tensor $\lambda \cdot u_i^{\otimes 6}$ with:
$\lambda = 2,\ u_1 = (1, 0, 0);\ u_2 = (-1, 0, 0);\ u_3 = (0, 1, 0);\ u_4 = (0, -1, 0)$ and
$\| \mathcal{F} - \lambda \cdot u_i^{\otimes 6} \| = 20.59$.

Example 3.11: Consider the tensor $\mathcal{F} \in \mathbb{R}^{2\times2\times2\times2}$ with entries
$\mathcal{F}_{1111} = 25.1, \mathcal{F}_{1212} = 25.6, \mathcal{F}_{2121} = 24.8, \mathcal{F}_{2222} = 23$
We get the rank-1 tensor $\lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3 \otimes u_i^4$ with:
$\lambda = 25.6,\ u_1^1 = (1, 0), u_1^2 = (0, 1), u_1^3 = (1, 0), u_1^4 = (0, 1)$;
$u_2^1 = (-1, 0), u_2^2 = (0, -1), u_2^3 = (-1, 0), u_2^4 = (0, -1)$;
$u_3^1 = (-1, 0), u_3^2 = (0, -1), u_3^3 = (1, 0), u_3^4 = (0, 1)$;
$u_4^1 = (1, 0), u_4^2 = (0, 1), u_4^3 = (-1, 0), u_4^4 = (0, -1)$;
$u_5^1 = (-1, 0), u_5^2 = (0, 1), u_5^3 = (-1, 0), u_5^4 = (0, 1)$;
$u_6^1 = (1, 0), u_6^2 = (0, -1), u_6^3 = (1, 0), u_6^4 = (0, -1)$;
$u_7^1 = (1, 0), u_7^2 = (0, -1), u_7^3 = (-1, 0), u_7^4 = (0, 1)$;
$u_8^1 = (-1, 0), u_8^2 = (0, 1), u_8^3 = (1, 0), u_8^4 = (0, -1)$.
The distance between $\mathcal{F}$ and one of these solutions is $\| \mathcal{F} - \lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3 \otimes u_i^4 \| = 42.1195$.

Example 3.12: Consider the tensor $\mathcal{F} \in \mathbb{R}^{3\times3\times3}$ with entries
$\mathcal{F}_{111} = 0.4333, \mathcal{F}_{121} = 0.4278, \mathcal{F}_{131} = 0.4140, \mathcal{F}_{211} = 0.8154, \mathcal{F}_{221} = 0.0199,$
$\mathcal{F}_{231} = 0.5598, \mathcal{F}_{311} = 0.0643, \mathcal{F}_{321} = 0.3815, \mathcal{F}_{331} = 0.8834, \mathcal{F}_{112} = 0.4866,$
$\mathcal{F}_{122} = 0.8087, \mathcal{F}_{132} = 0.2073, \mathcal{F}_{212} = 0.7641, \mathcal{F}_{222} = 0.9924, \mathcal{F}_{232} = 0.8752,$
$\mathcal{F}_{312} = 0.6708, \mathcal{F}_{322} = 0.8296, \mathcal{F}_{332} = 0.1325, \mathcal{F}_{113} = 0.3871, \mathcal{F}_{123} = 0.0769,$
$\mathcal{F}_{133} = 0.3151, \mathcal{F}_{213} = 0.1355, \mathcal{F}_{223} = 0.7727, \mathcal{F}_{233} = 0.4089, \mathcal{F}_{313} = 0.9715,$
$\mathcal{F}_{323} = 0.7726, \mathcal{F}_{333} = 0.5526$
We get the rank-1 tensor $\lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3$ with:
$\lambda = 2.8166,\ u_1^1 = (0.4279, 0.6556, 0.62209), u_1^2 = (0.5705, 0.6466, 0.5063), u_1^3 = (0.4500, 0.7093, 0.5425)$;
$u_2^1 = (0.4279, 0.6556, 0.62209), u_2^2 = (-0.5705, -0.6466, -0.5063), u_2^3 = (-0.4500, -0.7093, -0.5425)$;
$u_3^1 = (-0.4279, -0.6556, -0.62209), u_3^2 = (0.5705, 0.6466, 0.5063), u_3^3 = (-0.4500, -0.7093, -0.5425)$;
$u_4^1 = (-0.4279, -0.6556, -0.62209), u_4^2 = (-0.5705, -0.6466, -0.5063), u_4^3 = (0.4500, 0.7093, 0.5425)$,

The distance between $\mathcal{F}$ and one of these solutions is $|| \mathcal{F} - \lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3 ||= 1.3510$.

Example 3.13: Consider the tensor $\mathcal{F} \in \mathbb{R}^{3\times3\times3}$ with entries
$\mathcal{F}_{111} = 0.0072, \mathcal{F}_{121} = -0.4413, \mathcal{F}_{131} = 0.1941, \mathcal{F}_{211} = -04413, \mathcal{F}_{221} = 0.0940,$
$\mathcal{F}_{231} = 0.5901, \mathcal{F}_{311} = 0.1941, \mathcal{F}_{321} = -0.4099, \mathcal{F}_{331} = -0.1012, \mathcal{F}_{112} = -0.4413,$
$\mathcal{F}_{122} = 0.0940, \mathcal{F}_{132} = -0.4099, \mathcal{F}_{212} = 0.0940, \mathcal{F}_{222} = 0.2183, \mathcal{F}_{232} = 0.2950,$
$\mathcal{F}_{312} = 0.5901, \mathcal{F}_{322} = 0.2950, \mathcal{F}_{332} = 0.2229, \mathcal{F}_{113} = 0.1941, \mathcal{F}_{123} = 0.5901,$
$\mathcal{F}_{133} = -01012, \mathcal{F}_{213} = -0.4099, \mathcal{F}_{223} = 0.2950, \mathcal{F}_{233} = 0.2229, \mathcal{F}_{313} = -0.1012,$
$\mathcal{F}_{323} = 0.2229, \mathcal{F}_{333} = -0.4891$
We get the rank-1 tensor $\lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3$ with $\lambda = 1.000$ and the 12 solutions
$u_1^1 = (0.7955, 0.2491, 0.5524), u_1^2 = (-0.0050, 0.9142, -0.4051), u_1^3 = (-0.6060, 0.3195, 0.7285);$
$u_2^1 = (-0.0050, 0.9142, -0.4051), u_2^2 = (-0.6060, 0.3195, 0.7285), u_2^3 = (0.7955, 0.2491, 0.5524);$
$u_3^1 = (-0.6060, 0.3195, 0.7285), u_3^2 = (0.7955, 0.2491, 0.5524), u_3^2 = (-0.0050, 0.9142, -0.4051);$
$u_4^1 = (0.7955, 0.2491, 0.5524), u_4^2 = (0.0050, -0.9142, 0.4051), u_4^3 = (0.6060, -0.3195, -0.7285);$
$u_5^1 = (0.6060, -0.3195, -0.7285), u_5^2 = (0.7955, 0.2491, 0.5524), u_5^2 = (0.0050, -0.9142, 0.4051);$
$u_6^1 = (-0.6060, 0.3195, 0.7285), u_6^2 = (-0.7955, -0.2491, -0.5524), u_6^2 = (0.0050, -0.9142, 0.4051);$
$u_7^1 = (0.6060, -0.3195, -0.7285), u_7^2 = (-0.7955, -0.2491, -0.5524), u_7^2 = (-0.0050, 0.9142, -0.4051);$
$u_8^1 = (-0.7955, -0.2491, -0.5524), u_8^2 = (-0.0050, 0.9142, -0.4051), u_8^3 = (0.6060, -0.3195, -0.7285);$
$u_9^1 = (-0.7955, -0.2491, -0.5524), u_9^2 = (0.0050, -0.9142, 0.4051), u_9^3 = (-0.6060, 0.3195, 0.7285);$
$u_{10}^1 = (-0.0050, 0.9142, -0.4051), u_{10}^2 = (0.6060, -0.3195, -0.7285), u_{10}^3 = (-0.7955, -0.2491, -0.5524);$
$u_{11}^1 = (0.0050, -0.9142, 0.4051), u_{11}^2 = (0.6060, -0.3195, -0.7285), u_{11}^3 = (0.7955, 0.2491, 0.5524);$
$u_{12}^1 = (0.0050, -0.9142, 0.4051), u_{12}^2 = (-0.6060, 0.3195, 0.7285), u_{12}^3 = (-0.7955, -0.2491, -0.5524).$
The distance between $\mathcal{F}$ and one of these solutions is $|| \mathcal{F} - \lambda \cdot u_i^1 \otimes u_i^2 \otimes u_i^3 ||= 1.4143$.

Example 4.2: Consider the tensor $\mathcal{F} \in S^4(\mathbb{R}^3)$ with entries
$\mathcal{F}_{1111} = 0.1023, \mathcal{F}_{1112} = -0.002, \mathcal{F}_{1113} = 0.0581, \mathcal{F}_{1122} = 0.0039, \mathcal{F}_{1123} = -0.00032569,$
$\mathcal{F}_{1133} = 0.0407, \mathcal{F}_{1222} = 0.0107, \mathcal{F}_{1223} = -0.0012, \mathcal{F}_{1233} = -0.0011, \mathcal{F}_{1333} = 0.0196,$
$\mathcal{F}_{2222} = 0.0197, \mathcal{F}_{2223} = -0.0029, \mathcal{F}_{2233} = -0.00017418, \mathcal{F}_{2333} = -0.0021,$
$\mathcal{F}_{3333} = 0.1869$
We get the rank-2 tensor $\tilde{\mathcal{F}}(s,t,u) = (as+bt+cu)^4 + (ds+et+fu)^4$ with the 8 solutions:
$s_1 = (a, b, c, d, e, f) = (0.01877, 0.006239, -0.6434, -0.5592, 0.008797, -0.3522);$
$s_2 = (-0.01877, -0.006239, 0.6434, 0.5592, -0.008797, 0.3522);$
$s_3 = (0.01877, 0.006239, -0.6434, 0.5592, -0.008797, 0.3522);$
$s_4 = (-0.01877, -0.006239, 0.6434, -0.5592, 0.008797, -0.3522);$
$s_5 = (-0.5592, 0.008797, -0.3522, 0.01877, 0.006239, -0.6434);$
$s_6 = (0.5592, -0.008797, 0.3522, -0.01877, -0.006239, 0.6434);$
$s_7 = (-0.5592, 0.008797, -0.3522, -0.01877, -0.006239, 0.6434);$
$s_8 = (0.5592, -0.008797, 0.3522, 0.01877, 0.006239, -0.6434);.$
The distance between $\mathcal{F}$ and one of these solutions is $|| \mathcal{F} - \tilde{\mathcal{F}} ||= 0.00108483$.
The other possible real rank-2 approximations $\tilde{\mathcal{F}}(s,t,u) = \pm(as+bt+cu)^4 \pm (ds+et+fu)^4$ yield solutions which are not as close to $\mathcal{F}$ as these solutions.