



HAL
open science

Minimizing the sum of many rational functions

Florian Bugarin, Didier Henrion, Jean-Bernard Lasserre

► **To cite this version:**

Florian Bugarin, Didier Henrion, Jean-Bernard Lasserre. Minimizing the sum of many rational functions. 2011. hal-00569067v1

HAL Id: hal-00569067

<https://hal.science/hal-00569067v1>

Submitted on 24 Feb 2011 (v1), last revised 4 Dec 2014 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Minimizing the sum of many rational functions*

Florian Bugarin,^{1,2} Didier Henrion,^{2,3} Jean-Bernard Lasserre^{2,4}

February 24, 2011

Abstract

We consider the problem of globally minimizing the sum of many rational functions over a given compact semialgebraic set. The number of terms can be large (10 to 100), the degree of each term should be small (up to 10), and the number of variables can be large (10 to 100) provided some kind of sparsity is present. We describe a formulation of the rational optimization problem as a generalized moment problem and its hierarchy of convex semidefinite relaxations. Under some conditions we prove that the sequence of optimal values converges to the globally optimal value. We show how public-domain software can be used to model and solve such problems.

Keywords: rational optimization; global optimization; semidefinite relaxations; sparsity.
AMS MSC 2010: 46N10, 65K05, 90C22, 90C26.

1 Introduction

Consider the optimization problem

$$f^* := \inf_{\mathbf{x} \in \mathbf{K}} \sum_{i=1}^N f_i(\mathbf{x}) \quad (1)$$

over the basic semi-algebraic set

$$\mathbf{K} := \{\mathbf{x} \in \mathbb{R}^n : g_j(\mathbf{x}) \geq 0, j = 1, \dots, m\}, \quad (2)$$

for given polynomials $g_j \in \mathbb{R}[\mathbf{x}]$, $j = 1, \dots, m$, and where each term $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ is a rational function

$$\mathbf{x} \mapsto f_i(\mathbf{x}) := \frac{p_i(\mathbf{x})}{q_i(\mathbf{x})},$$

¹Université de Toulouse; Mines Albi; Institut Clément Ader (ICA); Campus Jarlard, F-81013 Albi, France

²CNRS; LAAS; 7 avenue du colonel Roche, F-31077 Toulouse, France; Université de Toulouse; UPS, INSA, INP, ISAE; LAAS; F-31077 Toulouse, France

³Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 4, CZ-16607 Prague, Czech Republic

⁴Institut de Mathématiques de Toulouse (IMT), Université de Toulouse, France

*The first author acknowledges support by the French National Research Agency (ANR) through COSINUS program (project ID4CS ANR-09-COSI-005). The second author acknowledges support by Research Program MSM6840770038 of the Czech Ministry of Education and Project 103/10/0628 of the Grant Agency of the Czech Republic.

with $p_i, q_i \in \mathbb{R}[\mathbf{x}]$ and $q_i > 0$ on \mathbf{K} , for each $i = 1, \dots, N$.

Problem (1) is a *fractional programming* problem of a rather general form. Nevertheless, we assume that the degree of each f_i and g_j is relatively small (up to 10), but the number of terms N can be quite large (10 to 100). For dense data the number of variables n should also be small (up to 10). However, this number can be also quite large (10 to 100) provided that the problem data feature some kind of sparsity (to be specified later). Even though problem (1) is of self-interest, our initial motivation came from some applications in computer vision, where such problems are typical. These applications will be described elsewhere.

In such a situation, fractional programming problem (1) is quite challenging. Indeed, we make *no* assumption on the polynomials p_i, q_i whereas even with a relatively small number of fractions and under convexity (resp. concavity) assumptions on p_i (resp. q_i), problem (1) is hard to solve (especially if one wants to compute the global minimum); see for example the survey [10] and references therein.

We are interested in solving problem (1) globally, in the sense that we do not content ourselves with a local optimum satisfying first order optimality conditions, as typically obtained with standard local optimization algorithms such as Newton's method or its variants. If problem (1) is too difficult to solve globally (because of ill-conditioning and/or too large a number of variables or terms in the objective function), we would like to have at least a valid lower bound on the global minimum, since upper bounds can be obtained with local optimization algorithms.

One possible approach is to reduce all fractions p_i/q_i to same denominator and obtain a single rational fraction to minimize. Then one may try to apply the hierarchy of semidefinite programming (SDP) relaxations defined in [6], see also [9, Section 5.8]. But such a strategy is not appropriate because the degree of the common denominator is potentially large and even if n is small, one may not even implement the first relaxation of the hierarchy, due to the present limitations of SDP solvers. Moreover, in general this strategy also destroys potential sparsity patterns present in the original formulation (1), and so precludes from using an appropriate version (for the rational fraction case) of the sparse semidefinite relaxations introduced in [11] whose convergence was proved in [7] under some conditions on the sparsity pattern, see also [9, Sections 4.6 and 5.3.4].

Another possibility is to introduce additional variables r_i (that we may call *liftings*) with associated constraints

$$\frac{p_i(\mathbf{x})}{q_i(\mathbf{x})} \leq r_i, \quad i = 1, \dots, N,$$

and solve the equivalent problem:

$$f^* := \inf_{(\mathbf{x}, \mathbf{r}) \in \widehat{\mathbf{K}}} \sum_{i=1}^N r_i \quad (3)$$

which is now a *polynomial* optimization problem in the new variables $(\mathbf{x}, \mathbf{r}) \in \mathbb{R}^n \times \mathbb{R}^N$, and where the new feasible set $\widehat{\mathbf{K}} = \mathbf{K} \times \{(\mathbf{x}, \mathbf{r}) \in \mathbb{R}^{n+N} : r_i q_i(\mathbf{x}) - p_i(\mathbf{x}) \geq 0\}$ is modeling the epigraphs of the rational terms. The sparsity pattern is preserved and if \mathbf{K} is compact one may in general obtain upper and lower bounds $\bar{r}_i, \underline{r}_i$ on the r_i so as to make $\widehat{\mathbf{K}}$ compact by adding the quadratic (redundant) constraints $(r_i - \underline{r}_i)(\bar{r}_i - r_i) \geq 0$, $i = 1, \dots, N$, and apply the sparse semidefinite relaxations. However, in doing so one introduces N additional variables, and this may have an impact on the overall performance, especially if N is large. In the sequel this approach is referred to as the *epigraph approach*.

The goal of the present paper is to circumvent all above difficulties in the following two situations: either n is relatively small, or n is potentially large but some sparsity is present, i.e., each f_i and each g_j in (1) is concerned with only a small subset of variables. In the approach that we propose, we do not need the epigraph liftings. The idea is to formulate (1) as an equivalent infinite-dimensional linear problem which a particular instance of the generalized moment problem (GMP) as defined in [8], with N unknown measures (where each measure is associated with a fraction p_i/q_i). In turn this problem can be easily modeled and solved with our public-domain software GloptiPoly 3 [5], a significant update of GloptiPoly 2 [4]. In the sequel this approach is referred to as the *GMP approach*.

The outline of the paper is as follows. In Section 2 we introduce the SDP relaxations first in the case that n is small and the data are dense polynomials. Then in Section 3 we extend the SDP relaxations to the case that n is large but sparsity is present. In Section 4 we show how the GMP formulation can be exploited to model the SDP relaxations of problem (1) easily with GloptiPoly 3. We also provide a collection of numerical experiments showing the relevance of our GMP approach, especially in comparison with the epigraph approach.

2 Dense SDP relaxations

In this section we assume that n , the number of variables in problem (1), is small, say up to 10.

2.1 GMP formulation

Consider the infinite dimensional linear problem

$$\begin{aligned} \hat{f} := \inf_{\mu_i \in \mathcal{M}(\mathbf{K})} & \sum_{i=1}^N \int_{\mathbf{K}} p_i d\mu_i \\ \text{s.t.} & \int_{\mathbf{K}} q_1 d\mu_1 = 1 \\ & \int_{\mathbf{K}} \mathbf{x}^\alpha q_i d\mu_i = \int_{\mathbf{K}} \mathbf{x}^\alpha q_1 d\mu_1, \quad \forall \alpha \in \mathbb{N}^n, i = 2, \dots, N, \end{aligned} \tag{4}$$

where $\mathcal{M}(\mathbf{K})$ is the space of finite Borel measures supported on \mathbf{K} .

Theorem 2.1 *Let $\mathbf{K} \subset \mathbb{R}^n$ in (2) be compact, and assume that $q_i > 0$ on \mathbf{K} , $i = 1, \dots, N$. Then $\hat{f} = f^*$.*

Proof: We first prove that $f^* \geq \hat{f}$. As $f = \sum_i p_i/q_i$ is continuous on \mathbf{K} , there exists a global minimizer $\mathbf{x}^* \in \mathbf{K}$ with $f(\mathbf{x}^*) = f^*$. Define $\mu_i := q_i(\mathbf{x}^*)^{-1} \delta_{\mathbf{x}^*}$, $i = 1, \dots, N$, where $\delta_{\mathbf{x}^*}$ is the Dirac measure at \mathbf{x}^* . Then obviously, the measures (μ_i) , $i = 1, \dots, N$, are feasible for (4) with associated value

$$\sum_{i=1}^N \int_{\mathbf{K}} p_i d\mu_i = \sum_{i=1}^N p_i(\mathbf{x}^*)/q_i(\mathbf{x}^*) = f(\mathbf{x}^*) = f^*.$$

Conversely, let (μ_i) be a feasible solution of (4). For every $i = 1, \dots, N$, let $d\nu_i$ be the measure $q_i d\mu_i$, i.e.

$$\nu_i(B) := \int_{\mathbf{K} \cap B} q_i(\mathbf{x}) d\mu_i(\mathbf{x})$$

for all sets B in the Borel σ -algebra of \mathbb{R}^n , and so the support of ν_i is \mathbf{K} . As measures on compact sets are moment determinate, the moments constraints of (4) imply that $\nu_i = \nu_1$, for every $i = 2, \dots, N$, and from $\int_{\mathbf{K}} q_1 d\mu_1 = 1$ we also deduce that ν_1 is a probability measure on \mathbf{K} . But then

$$\begin{aligned} \sum_{i=1}^N \int_{\mathbf{K}} p_i d\mu_i &= \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} q_i d\mu_i = \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} d\nu_1 \\ &= \int_{\mathbf{K}} \left(\sum_{i=1}^N \frac{p_i}{q_i} \right) d\nu_1 = \int_{\mathbf{K}} f d\nu_1 \geq \int_{\mathbf{K}} f^* d\nu_1 = f^*, \end{aligned}$$

where we have used that $f \geq f^*$ on \mathbf{K} and ν_1 is a probability measure on \mathbf{K} . \square

We next make the following assumption meaning that set \mathbf{K} admits an algebraic certificate of compactness.

Assumption 2.1 *The set $\mathbf{K} \subset \mathbb{R}^n$ in (2) is compact and the quadratic polynomial $\mathbf{x} \mapsto M - \|\mathbf{x}\|^2$ can be written as*

$$M - \|\mathbf{x}\|^2 = \sigma_0 + \sum_{j=1}^m \sigma_j g_j,$$

for some polynomials $\sigma_j \in \mathbb{R}[\mathbf{x}]$, all sums of squares of polynomials.

2.2 A hierarchy of dense SDP relaxations

Let $\mathbf{y}_i = (y_{i\alpha})$ be a real sequence indexed in the canonical basis (\mathbf{x}^α) of $\mathbb{R}[\mathbf{x}]$, $i = 1, \dots, N$, and for every $k \in \mathbb{N}$, let $\mathbb{N}_k^n := \{\alpha \in \mathbb{N}^n : \sum_j \alpha_j \leq k\}$.

Define the *moment* matrix $M_k(\mathbf{y}_i)$ of order k , associated with \mathbf{y}_i , whose entries indexed by multi-indices β (rows) and γ (columns) read

$$[M_k(\mathbf{y}_i)]_{\beta, \gamma} := y_{i(\beta+\gamma)}, \quad \forall \beta, \gamma \in \mathbb{N}_k^n,$$

and so are linear in \mathbf{y}_i . Similarly, given a polynomial $g(\mathbf{x}) = \sum_{\alpha} g_{\alpha} \mathbf{x}^{\alpha}$, define the *localising* matrix $M_k(g\mathbf{y}_i)$ of order k , associated with \mathbf{y}_i and g , whose entries read

$$[M_k(g\mathbf{y}_i)]_{\beta, \gamma} := \sum_{\alpha} g_{\alpha} y_{i(\alpha+\beta+\gamma)}, \quad \forall \beta, \gamma \in \mathbb{N}_k^n.$$

In particular, matrix $M_0(g\mathbf{y}_i)$ is identical to $L_{\mathbf{y}_i}(g)$ where for every i , $L_{\mathbf{y}_i} : \mathbb{R}[\mathbf{x}] \rightarrow \mathbb{R}$ is the linear functional defined by:

$$g \mapsto L_{\mathbf{y}_i}(g) := \sum_{\alpha \in \mathbb{N}^n} g_{\alpha} y_{i\alpha}, \quad \forall g \in \mathbb{R}[\mathbf{x}].$$

Let $u_i := \lceil (\deg q_i)/2 \rceil$, $i = 1, \dots, N$, $r_j := \lceil (\deg g_j)/2 \rceil$, $j = 1, \dots, m$, and with no loss of generality assume that $u_1 \leq u_2 \leq \dots \leq u_N$. Consider the hierarchy of semidefinite programming (SDP) relaxations:

$$\begin{aligned} f_k^* &= \inf_{\mathbf{y}_i} \sum_{i=1}^N L_{\mathbf{y}_i}(p_i) \\ \text{s.t. } & M_k(\mathbf{y}_i) \succeq 0, & i = 1, \dots, N \\ & M_{k-r_j}(g_j \mathbf{y}_i) \succeq 0, & i = 1, \dots, N, j = 1, \dots, m \\ & L_{\mathbf{y}_1}(q_1) = 1 \\ & L_{\mathbf{y}_i}(\mathbf{x}^{\alpha} q_i) = L_{\mathbf{y}_1}(\mathbf{x}^{\alpha} q_1), & \forall \alpha \in \mathbb{N}_{2(k-u_i)}^n, i = 2, \dots, N. \end{aligned} \tag{5}$$

Theorem 2.2 *Let Assumption 2.1 hold and consider the hierarchy of SDP relaxations (5). Then it follows that*

(a) $f_k^* \uparrow f^*$ as $k \rightarrow \infty$.

(b) Moreover, if (\mathbf{y}_i^k) is an optimal solution of (5), and if

$$\text{rank } M_k(\mathbf{y}_i^k) = \text{rank } M_{k-u_i}(\mathbf{y}_i^k) =: R, \quad i = 1, \dots, N$$

then $f_k^* = f^*$ and one may extract R global minimizers.

Proof: The proof of (a) is classical. One first prove that if (\mathbf{y}_i^k) is a nearly optimal solution of (5), i.e.

$$f_k^* \leq \sum_{i=1}^N L_{\mathbf{y}_i^k}(p_i) \leq f_k^* + \frac{1}{k},$$

then there exists a subsequence (k_ℓ) and a sequence \mathbf{y}_i , $i = 1, \dots, N$, such that

$$\lim_{\ell \rightarrow \infty} y_{i\alpha}^{k_\ell} = y_{i\alpha}, \quad \forall \alpha \in \mathbb{N}^n, i = 1, \dots, N.$$

From this pointwise convergence it easily follows that for every $i = 1, \dots, N$ and $j = 1, \dots, m$,

$$M_k(\mathbf{y}_i) \succeq 0, \quad M_k(g_j \mathbf{y}_i) \succeq 0, \quad k = 0, 1, \dots$$

By Putinar's theorem [9, Theorem 2.14] this implies that the sequence \mathbf{y}_i has a representing measure supported on \mathbf{K} , i.e., there exists a finite Borel measure μ_i on \mathbf{K} such that

$$L_{\mathbf{y}_i}(f) = \int_{\mathbf{K}} f d\mu_i, \quad \forall f \in \mathbb{R}[\mathbf{x}].$$

Moreover, still by pointwise convergence,

$$L_{\mathbf{y}_i}(q_i \mathbf{x}^\alpha) = \int_{\mathbf{K}} \mathbf{x}^\alpha q_i(\mathbf{x}) d\mu_i = L_{\mathbf{y}_1}(q_1 \mathbf{x}^\alpha) = \int_{\mathbf{K}} \mathbf{x}^\alpha q_1(\mathbf{x}) d\mu_1, \quad \forall \alpha \in \mathbb{N}^n. \quad (6)$$

Therefore, let $d\nu_i := q_i(\mathbf{x}) d\mu_i$ which is a probability measure supported on \mathbf{K} . As \mathbf{K} is compact, by (6), $\nu_i = \nu_1$ for every $i = 1, \dots, N$. Finally, again by pointwise convergence:

$$\begin{aligned} f^* &\geq \lim_{\ell \rightarrow \infty} f_{k_\ell}^* = \lim_{\ell \rightarrow \infty} \sum_{i=1}^N L_{\mathbf{y}_i^{k_\ell}}(p_i) = \sum_{i=1}^N \int_{\mathbf{K}} p_i d\mu_i \\ &= \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} q_i d\mu_i = \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} d\nu_1 \\ &= \int_{\mathbf{K}} \left(\sum_{i=1}^N \frac{p_i}{q_i} \right) d\nu_1 \geq f^* \end{aligned}$$

which proves (a) because f_k^* is monotone non-decreasing. In addition, ν_1 is an optimal solution of (4) with optimal value $f^* = \hat{f}$.

Statement (b) follows from the flat extension theorem of Curto and Fialkow [9, Theorem 3.7] and each \mathbf{y}_i has an atomic representing measure supported on R points of \mathbf{K} . \square

3 Sparse SDP relaxations

In this section we assume that n , the number of variables in problem (1), is large, say from 10 to 100, and moreover that some sparsity pattern is present in the polynomial data.

3.1 GMP formulation

Let $I_0 := \{1, \dots, n\} = \cup_{i=1}^N I_i$ with possible overlaps, and let $\mathbb{R}[x_k : k \in I_i]$ denote the ring of polynomials in the variables x_k , $k \in I_i$. Denote by n_i the cardinality of I_i .

One will assume that $\mathbf{K} \subset \mathbb{R}^n$ in (2) is compact, and one knows some $M > 0$ such that $\mathbf{x} \in \mathbf{K} \Rightarrow M - \|\mathbf{x}\|^2 \geq 0$. For every $i \leq N$, introduce the quadratic polynomial $\mathbf{x} \mapsto g_{m+i}(\mathbf{x}) = M - \sum_{k \in I_i} x_k^2$. The index set $\{1, \dots, m + N\}$ has a partition $\cup_{i=1}^N J_i$ with $J_i \neq \emptyset$ for every $i = 1, \dots, N$. In the sequel we assume that for every $i = 1, \dots, N$, $p_i, q_i \in \mathbb{R}[x_k : k \in I_i]$ and for every $j \in J_i$, $g_j \in \mathbb{R}[x_k : k \in I_i]$. Next, for every $i = 1, \dots, N$, let

$$\mathbf{K}_i := \{\mathbf{z} \in \mathbb{R}^{n_i} : g_k(\mathbf{z}) \geq 0, k \in J_i\}$$

so that \mathbf{K} in (2) has the equivalent characterization

$$\mathbf{K} = \{\mathbf{x} \in \mathbb{R}^n : (x_k, k \in I_i) \in \mathbf{K}_i, i = 1, \dots, N\}.$$

Similarly, for every $i, j \in \{1, \dots, N\}$ such that $i \neq j$ and $I_i \cap I_j \neq \emptyset$,

$$\mathbf{K}_{ij} = \mathbf{K}_{ji} := \{(x_k, k \in I_i \cap I_j) : (x_k, k \in I_i) \in \mathbf{K}_i; (x_k, k \in I_j) \in \mathbf{K}_j\}.$$

Let $\mathcal{M}(\mathbf{K})$ be the space of finite Borel measures on \mathbf{K} , and for every $i = 1, \dots, N$, let $\pi_i : \mathcal{M}(\mathbf{K}) \rightarrow \mathcal{M}(\mathbf{K}_i)$ denote the projection on \mathbf{K}_i , that is, for every $\mu \in \mathcal{M}(\mathbf{K})$:

$$\pi_i \mu(B) := \mu(\{\mathbf{x} : \mathbf{x} \in \mathbf{K}; (x_k, k \in I_i) \in B\}), \quad \forall B \in \mathcal{B}(\mathbf{K}_i)$$

where $\mathcal{B}(\mathbf{K}_i)$ is the usual Borel σ -algebra associated with \mathbf{K}_i .

For every $i, j \in \{1, \dots, N\}$ such that $i \neq j$ and $I_i \cap I_j \neq \emptyset$, the projection $\pi_{ij} : \mathcal{M}(\mathbf{K}_i) \rightarrow \mathcal{M}(\mathbf{K}_{ij})$ is also defined in an obvious similar manner. For every $i = 1, \dots, N - 1$ define the set:

$$U_i := \{j \in \{i + 1, \dots, N\} : I_i \cap I_j \neq \emptyset\},$$

and consider the infinite dimensional problem

$$\begin{aligned} \hat{f} &:= \inf_{\mu_i \in \mathcal{M}(\mathbf{K}_i)} \sum_{i=1}^N \int_{\mathbf{K}_i} p_i d\mu_i \\ \text{s.t.} & \int_{\mathbf{K}_i} q_i d\mu_i = 1, \quad i = 1, \dots, N \\ & \pi_{ij}(q_i d\mu_i) = \pi_{ji}(q_j d\mu_j), \quad \forall j \in U_i, i = 1, \dots, N - 1. \end{aligned} \tag{7}$$

Definition 3.1 *Sparsity pattern $(I_i)_{i=1}^N$ satisfies the running intersection property if for every $i = 2, \dots, N$:*

$$I_i \cap \left(\bigcup_{k=1}^{i-1} I_k \right) \subseteq I_j, \quad \text{for some } j \leq i - 1.$$

Theorem 3.1 *Let $\mathbf{K} \subset \mathbb{R}^n$ in (2) be compact. If the sparsity pattern $(I_i)_{i=1}^N$ satisfies the running intersection property then $\hat{f} = f^*$.*

Proof: That $\hat{f} \leq f^*$ is straightforward. As \mathbf{K} is compact and $q_i > 0$ on \mathbf{K} for every $i = 1, \dots, N$, $f^* = \sum_{i=1}^N f_i(\mathbf{x}^*)$ for some $\mathbf{x} \in \mathbf{K}$. So let μ be the Dirac measure $\delta_{\mathbf{x}^*}$ at \mathbf{x}^* and let ν_i be the projection $\pi_i \mu$ of μ on \mathbf{K}_i . That is $\nu_i = \delta_{(x_k^*, k \in I_i)}$, the Dirac measure at the point $(x_k^*, k \in I_i)$ of \mathbf{K}_i . Next, for every $i = 1, \dots, N$, define the measure $d\mu_i := q_i(\mathbf{x}^*)^{-1} d\nu_i$. Obviously, (μ_i) is a feasible solution of (7) because $\mu_i \in \mathcal{M}(\mathbf{K}_i)$ and $\int q_i d\mu_i = 1$, for every $i = 1, \dots, N$, and one also has:

$$(x_k^*, k \in I_i \cap I_j) = \pi_{ij} \mu_i = \pi_{ji} \mu_j, \quad \forall j \neq i \text{ such that } I_j \cap I_i \neq \emptyset.$$

Finally, its value satisfies

$$\sum_{i=1}^N \int_{\mathbf{K}_i} p_i d\mu_i = \sum_{i=1}^N p_i(\mathbf{x}^*)/q_i(\mathbf{x}^*) = f^*,$$

and so $\hat{f} \leq f^*$.

We next prove the converse inequality $\hat{f} \geq f^*$. Let (μ_i) be an arbitrary feasible solution of (7), and for every $i = 1, \dots, N$, denote by ν_i the probability measure on \mathbf{K}_i with density q_i with respect to μ_i , that is,

$$\nu_i(B) := \int_{\mathbf{K}_i \cap B} q_i(\mathbf{x}) d\mu_i(\mathbf{x}), \quad \forall B \in \mathcal{B}(\mathbf{K}_i).$$

By definition of the linear program (7), $\pi_{ij} \nu_i = \pi_{ji} \nu_j$ for every couple $j \neq i$ such that $I_j \cap I_i \neq \emptyset$. Therefore, by [9, Lemma B.13] there exists a probability measure ν on \mathbf{K} such that $\pi_i \nu = \nu_i$ for every $i = 1, \dots, N$. But then

$$\begin{aligned} \sum_{i=1}^N \int_{\mathbf{K}_i} p_i d\mu_i &= \sum_{i=1}^N \int_{\mathbf{K}_i} \frac{p_i}{q_i} d\nu_i = \sum_{i=1}^N \int_{\mathbf{K}_i} \frac{p_i}{q_i} d\nu \\ &= \int_{\mathbf{K}} \left(\sum_{i=1}^N \frac{p_i}{q_i} \right) d\nu \geq f^* \end{aligned}$$

and so $\hat{f} \geq f^*$. □

3.2 A hierarchy of sparse SDP relaxations

Let $\mathbf{y} = (y_\alpha)$ be a real sequence indexed in the canonical basis (\mathbf{x}^α) of $\mathbb{R}[\mathbf{x}]$. Define the linear functional $L_{\mathbf{y}} : \mathbb{R}[\mathbf{x}] \rightarrow \mathbb{R}$, by:

$$f \left(= \sum_{\alpha \in \mathbb{N}^n} f_\alpha \mathbf{x}^\alpha \right) \mapsto \sum_{\alpha \in \mathbb{N}^n} f_\alpha y_\alpha, \quad \forall f \in \mathbb{R}[\mathbf{x}].$$

For every $i = 1, \dots, N$, let

$$\mathbb{N}^{(i)} := \{ \alpha \in \mathbb{N}^n : \alpha_k = 0 \text{ if } k \notin I_i \}; \quad \mathbb{N}_k^{(i)} := \{ \alpha \in \mathbb{N}^{(i)} : \sum_i \alpha_i \leq k \}.$$

An obvious similar definition of $\mathbb{N}^{(ij)}$ ($= \mathbb{N}^{(ji)}$) and $\mathbb{N}_k^{(ij)}$ ($= \mathbb{N}_k^{(ji)}$) applies when considering $I_j \cap I_i \neq \emptyset$.

Let $\mathbf{y} = (y_\alpha)$ be a given sequence indexed in the canonical basis of $\mathbb{R}[\mathbf{x}]$. For every $i = 1, \dots, N$, the sparse moment matrix $M_k(\mathbf{y}, I_i)$ associated with y , has its rows and columns indexed in the canonical basis (\mathbf{x}^α) of $\mathbb{R}[x_k : k \in I_i]$, and with entries:

$$M_k(\mathbf{y}, I_i)_{\alpha, \beta} = L_{\mathbf{y}}(\mathbf{x}^{\alpha+\beta}) = y_{\alpha+\beta}, \quad \forall \alpha, \beta \in \mathbb{N}_k^{(i)}.$$

Similarly, for a given polynomial $h \in \mathbb{R}[x_k : k \in I_i]$, the sparse localizing matrix $M_k(h \mathbf{y}, I_i)$ associated with \mathbf{y} and h , has its rows and columns indexed in the canonical basis (\mathbf{x}^α) of $\mathbb{R}[x_k : k \in I_i]$, and with entries:

$$M_k(h \mathbf{y}, I_i)_{\alpha, \beta} = L_{\mathbf{y}}(h \mathbf{x}^{\alpha+\beta}) = \sum_{\gamma \in \mathbb{N}^{(i)}} h_\gamma y_{\alpha+\beta+\gamma}, \quad \forall \alpha, \beta \in \mathbb{N}_k^{(i)}.$$

With $\mathbf{K} \subset \mathbb{R}^n$ defined in (2), let $r_j := \lceil (\deg g_j)/2 \rceil$, for every $j = 1, \dots, m + N$. Consider the hierarchy of semidefinite relaxations:

$$\begin{aligned} f_k^* &= \inf_{\mathbf{y}} \sum_{i=1}^N L_{\mathbf{y}}(p_i) \\ \text{s.t. } & M_k(\mathbf{y}, I_i) \succeq 0, & i = 1, \dots, N \\ & M_{k-r_j}(g_j \mathbf{y}, I_i) \succeq 0, & \forall j \in J_i, i = 1, \dots, N \\ & L_{\mathbf{y}}(q_i) = 1, & i = 1, \dots, N \\ & L_{\mathbf{y}}(\mathbf{x}^\alpha q_i) = L_{\mathbf{y}}(\mathbf{x}^\alpha q_j) = 0, & \forall \alpha \in \mathbb{N}^{(ij)}, \forall j \in U_i, i = 1, \dots, N-1 \\ & & \text{with } |\alpha| + \max[\deg q_i, \deg q_j] \leq 2k. \end{aligned} \quad (8)$$

Theorem 3.2 *Let $\mathbf{K} \subset \mathbb{R}^n$ in (2) be compact. Let the sparsity pattern $(I_i)_{i=1}^N$ satisfy the running intersection property, and consider the hierarchy of semidefinite relaxations defined in (8). Then:*

- (a) $f_k^* \uparrow f^*$ as $k \rightarrow \infty$.
- (b) *If an optimal solution \mathbf{y}^* of (8) satisfies*

$$\text{rank } \mathbf{M}_k(\mathbf{y}^*, I_i) = \text{rank } \mathbf{M}_{k-v_i}(\mathbf{y}^*, I_i) =: R_i, \quad \forall i = 1, \dots, N,$$

(where $v_i = \max_{j \in J_i} [r_j]$), and

$$\text{rank } \mathbf{M}_k(\mathbf{y}^*, I_i \cap I_j) = 1, \quad \forall j \in U_i, i = 1, \dots, N-1,$$

then $f_k^* = f^*$ and one may extract finitely many global minimizers.

Proof: The proof is similar to that of Theorem 2.2 and also to that of [9, Theorem 4.7]. One first prove that if (\mathbf{y}^k) is a nearly optimal solution of (5), i.e.

$$f_k^* \leq \sum_{i=1}^N L_{\mathbf{y}^k}(p_i) \leq f_k^* + \frac{1}{k},$$

then there exists a subsequence (k_ℓ) and a sequence \mathbf{y} , such that

$$\lim_{\ell \rightarrow \infty} y_\alpha^{k_\ell} = y_\alpha, \quad \forall \alpha \in \mathbb{N}^{(i)}, i = 1, \dots, N.$$

From this pointwise convergence it easily follows that for every $i = 1, \dots, N$ and $j \in J_i$,

$$M_k(\mathbf{y}, I_i) \succeq 0, M_k(g_j \mathbf{y}, I_i) \succeq 0, \quad j \in J_i; i = 1, \dots, N.$$

Now observe that each set $\mathbf{K}_i \subset \mathbb{R}^{n_i}$ satisfies Assumption 2.1. Therefore, by Putinar's theorem [9, Theorem 2.14] the sequence $\mathbf{y}^i = (y_\alpha)$, $\alpha \in \mathbb{N}^{(i)}$ (a subsequence of \mathbf{y}), has a representing measure μ_i supported on \mathbf{K}_i . For every (i, j) with $j \in U_i$, denote by \mathbf{y}^{ij} the sequence (y_α) , $\alpha \in \mathbb{N}^{(ij)}$. Again, by pointwise convergence, $L_{\mathbf{y}}(q_i) = 1$, $i = 1, \dots, N$, and

$$L_{\mathbf{y}}(q_i \mathbf{x}^\alpha) = \int_{\mathbf{K}_i} \mathbf{x}^\alpha q_j(\mathbf{x}) d\mu_j = \int_{\mathbf{K}_j} \mathbf{x}^\alpha q_j(\mathbf{x}) d\mu_j, \quad \forall \alpha \in \mathbb{N}^{(ij)}, \forall j \in U_i. \quad (9)$$

Therefore, for every $i = 1, \dots, N$, $d\nu_i := q_i(\mathbf{x})d\mu_i$ is a finite Borel probability measure supported on \mathbf{K}_i . As measures on compact sets are moment determinate, (9) yields:

$$\pi_{ij}\nu_i = \pi_{ji}\nu_j, \quad \forall (i, j), j \in U_i.$$

Therefore, by [9, Lemma B.13] there exists a probability measure ν on \mathbf{K} such that $\pi_i\nu = \nu_i$ for every $i = 1, \dots, N$. But then

$$\begin{aligned} f^* &\geq \lim_{\ell \rightarrow \infty} f_{k_\ell}^* = \lim_{\ell \rightarrow \infty} \sum_{i=1}^N L_{\mathbf{y}^{k_\ell}}(p_i) = \sum_{i=1}^N \int_{\mathbf{K}} p_i d\mu_i \\ &= \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} q_i d\mu_i = \sum_{i=1}^N \int_{\mathbf{K}} \frac{p_i}{q_i} d\nu_i \\ &= \int_{\mathbf{K}} \left(\sum_{i=1}^N \frac{p_i}{q_i} \right) d\nu \geq f^*. \end{aligned}$$

As the converging subsequence was arbitrary, and (f_k^*) is monotone non decreasing, we finally get $f_k^* \uparrow f^*$. In addition, ν is an optimal solution of (7) with optimal value $f^* = \hat{f}$.

The proof of (b) is as in [7] and uses the flat extension theorem of Curto and Fialkow [9, Theorem 3.7] from which, the sequence $\mathbf{y}_i^* = (y_\alpha^*)$, $\alpha \in \mathbb{N}^{(i)}$, has an atomic representing measure supported on R_i points of \mathbf{K}_i , for every $i = 1, \dots, N$. \square

4 GloptiPoly and examples

In this section we show that the generalized moment problem (GMP) formulation of rational optimization problem (1) has a straightforward Matlab implementation when using our software GloptiPoly 3 [5]. Rather than explaining the approach in full generality with awkward notations, we describe three simple examples.

4.1 Wilkinson-like rational function

Consider the elementary univariate rational optimization problem

$$f^* = \sup_{x \in \mathbb{R}} f(x), \quad f(x) = \sum_{i=1}^N \frac{p_i(x)}{q_i(x)} = \sum_{i=1}^N \frac{1}{x^2 + i}$$

with N an integer. The only real critical point is $x = 0$, at which the objective function takes its maximum

$$f^* = f(0) = \sum_{i=1}^N \frac{1}{i}.$$

Reducing to the same denominator

$$f(x) = \frac{\sum_i \prod_{j \neq i} (x^2 + j)}{\prod_i (x^2 + i)} = \frac{p(x^2)}{q(x^2)}$$

yields the well-known Wilkinson polynomial q whose squared root moduli are the integers from 1 to N . This polynomial was described in the mid 1960s by James H. Wilkinson to illustrate the difficulty of finding numerically the roots of polynomials. If we choose e.g. $N = 20$, reduction to the same denominator is hopeless since the constant coefficient in monic polynomial q is $20! = 2432902008176640000$. The GMP formulation (4) of this problem reads (up to replacing inf with sup in the objective function):

$$\begin{aligned} & \sup_{\mu_i \in \mathcal{M}(\mathbb{R})} \sum_{i=1}^N \int_{\mathbb{R}} p_i d\mu_i \\ & \text{s.t.} \quad \int_{\mathbb{R}} q_1 d\mu_1 = 1 \\ & \quad \int_{\mathbb{R}} x^\alpha q_i d\mu_i = \int_{\mathbb{R}} x^\alpha q_1 d\mu_1, \quad \forall \alpha \in \mathbb{N}^n, i = 1, \dots, N. \end{aligned}$$

Our Matlab script to model and solve this problem is as follows:

```
N = 20; mpol('x',N); % create variables
q = cell(N,1); % problem data
mu = cell(N,1); % measures
for i = 1:N, q{i} = i+x(i)^2; mu{i} = meas(x(i)); end
% model GMP
k = 0; % relaxation order
f = mass(mu{1}); % objective function
e = [mom(q{1}) == 1]; % moment constraints
for i = 2:N
    f = f + mass(mu{i});
    e = [e; mom(mmon(x(1),k)*q{1}) == mom(mmon(x(i),k)*q{i})];
end
% model SDP relaxation of GMP
P = msdp(max(f),e);
% solve SDP relaxation
[stat,obj] = msol(P)
```

Instructions `mpol`, `meas`, `mass`, `mom`, `mmon`, `msdp`, `max` and `msol` are GloptiPoly 3 commands, see the user's guide [5] for more information. For readers who are not familiar with this package, variable `f` is the objective function to be maximized. Since $p_i = 1$ for all $i = 1, \dots, N$, it is the sum of masses of measures μ_i . Vector `e` stores the linear moment constraints and the instruction `mmon(x,k)` generates all monomials of variable `x` up to degree `k`. Finally, instruction `msdp` generates the SDP relaxation of the GMP, and `msol` solves the SDP problem with the default conic solver (SeDuMi 1.3 in our case).

At the first SDP relaxation (i.e. for `k=0`) we obtain a rank-one moment matrix corresponding to a Dirac at $x^* = 0$:

```
>> [stat,obj] = msol(P)
```

```

Global optimality certified numerically
stat =
    1
obj =
    3.5977

```

which is consistent with Maple's

```

> f := sum(1/(x^2+i), i=1..20);
> evalf(subs(x = 0, f));
    3.5977

```

Note that for this example Assumption 2.1 is violated, since we optimize over the non-bounded set $\mathbf{K} = \mathbb{R}$. In spite of this, we could solve the problem globally.

4.2 Relevance of the compactness assumption

With this elementary example we would like to emphasize the practical relevance of Assumption 2.1 on the existence of an algebraic certificate of compactness of set \mathbf{K} . Consider the univariate problem

$$f^* = \inf_{x \in \mathbf{K}} f(x), \quad f(x) = \frac{1+x+x^2}{1+x^2} + \frac{1+x^2}{1+2x^2}. \quad (10)$$

First let $\mathbf{K} = \mathbb{R}$. The numerator of the gradient of $f(x)$ has two real roots, one of which being the global minimum located at $x^* = -1.4215$ for which $f^* = 1.1286$. The following GloptiPoly script models and solves the SDP relaxations of orders $k = 0, \dots, 9$ of the GMP formulation of this problem:

```

mpol x1 x2
f1 = 1+x1+x1^2; g1 = 1+x1^2; f2 = 1+x2^2; g2 = 1+2*x2^2;
mu1 = meas(x1); mu2 = meas(x2);
bounds = [];
for k = 0:9
    P = msdp(min(mom(f1)+mom(f2)), ...
        mom(mmon(x1,k)*g1) == mom(mmon(x2,k)*g2), mom(g1) == 1);
    [stat, obj] = msol(P);
    bounds = [bounds; obj];
end
bounds

```

In vector `bounds` we retrieve the following monotonically increasing sequence of lower bounds f_k^* (up to 5 digits) obtained by solving the SDP relaxations (5): At SDP relaxation $k = 9$, GloptiPoly certifies global optimality and extracts the global minimizer. Table 1 shows that the convergence of the hierarchy of SDP relaxations is rather slow for this very simple example. This is due to the fact that Assumption 2.1 is violated, since we optimize over the non-bounded set $\mathbf{K} = \mathbb{R}$.

On Figure 1 we report the sequences of lower bounds obtained by solving the SDP relaxations of problem (10) on compact sets $\mathbf{K} = [-R, R]$ for $R = 2, 3, \dots, 9$.

order k	bound f_k^*	order k	bound f_k^*
0	1.0000	5	1.0793
1	1.0000	6	1.1264
2	1.0170	7	1.1283
3	1.0220	8	1.1286
4	1.0633	9	1.1286

Table 1: Lower bounds for SDP relaxations of problem (10).

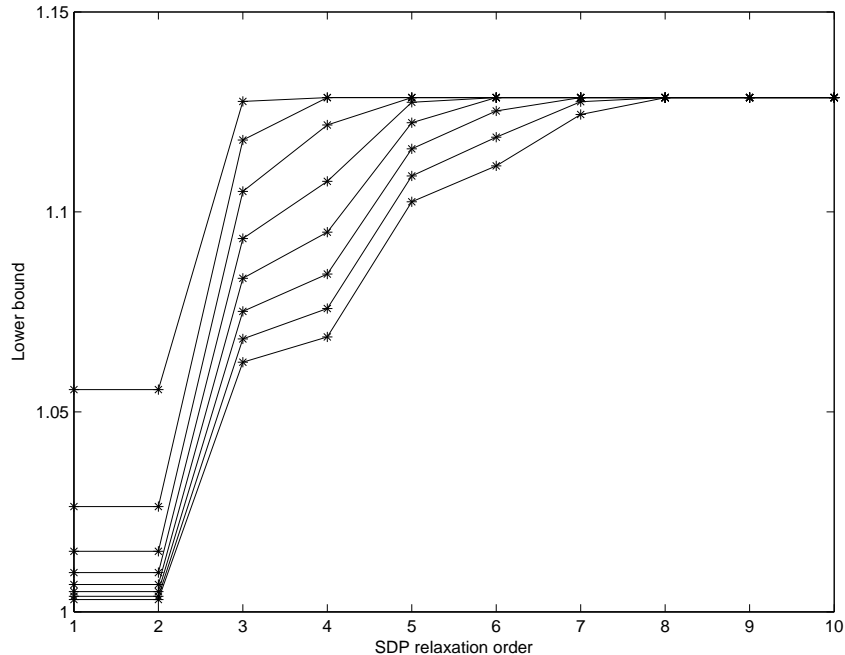


Figure 1: Lower bounds for SDP relaxations of problem (10) on bounded sets $K = [-R, R]$ for $R = 2$ (top curve) to $R = 9$ (bottom curve).

4.3 Exploiting sparsity with GloptiPoly

Even though version 3 of GloptiPoly is designed to exploit problem sparsity, there is no illustration of this feature in the software user's guide [5]. In this section we provide such a simple example. Note also that GloptiPoly is not able to detect sparsity in a given problem, contrary to SparsePOP which uses a heuristic to find chordal extensions of graphs [12]. However, SparsePOP is not designed to handle directly rational optimization problems.

Consider the elementary example of [7, Section 3.2]:

$$\begin{aligned}
& \inf_{\mathbf{x} \in \mathbb{R}^4} && x_1x_2 + x_1x_3 + x_1x_4 \\
& \text{s.t.} && x_1^2 + x_2^2 \leq 1 \\
& && x_1^2 + x_3^2 \leq 2 \\
& && x_1^2 + x_4^2 \leq 3
\end{aligned}$$

for which the variable index subsets $I_1 = \{1, 2\}$, $I_2 = \{1, 3\}$, $I_3 = \{1, 4\}$ satisfy the running intersection property of Definition 3.1. Note that this problem is a particular case of (1) with a polynomial objective function.

Without exploiting sparsity, the GloptiPoly script to solve this problem is as follows:

```

mpol x1 x2 x3 x4
Pdense = msdp(min(x1*x2+x1*x3+x1*x4), ...
              x1^2+x2^2<=1,x1^2+x3^2<=2,x1^2+x4^2<=3,2);
[stat,obj] = msol(Pdense);

```

GloptiPoly certifies global optimality with a moment matrix of size 15, and 3 localizing matrices of size 5. And here is the script exploiting sparsity, splitting the variables into several measures μ_i consistently with subsets I_i :

```

mpol x1 3
mpol x2 x3 x4
mu(1) = meas([x1(1) x2]); % first measure on x1 and x2
mu(2) = meas([x1(2) x3]); % second measure on x1 and x3
mu(3) = meas([x1(3) x4]); % third measure on x1 and x4
f = mom(x1(1)*x2)+mom(x1(2)*x3)+mom(x1(3)*x4); % objective function
k = 3; % SDP relaxation order
m1 = mom(mmon(x1(1),k)); % moments of first measure
m2 = mom(mmon(x1(2),k)); % moments of second measure
m3 = mom(mmon(x1(3),k)); % moments of third measure
K = [x1(1)^2+x2^2<=1, x1(2)^2+x3^2<=2, x1(3)^2+x4^2<=3]; % supports
Psparse = msdp(min(f),m1==m2,m3==m2,K,mass(mu)==1);
[stat,obj] = msol(Psparse);

```

GloptiPoly certifies global optimality with 3 moment matrices of size 6, and 3 localizing matrices of size 3.

4.4 Comparison with the epigraph approach

In most of the examples we have processed, the epigraph approach described in the Introduction (consisting of introducing one lifting variable for each rational term in the objective function) was less efficient than the GMP approach. Typically, the order of the SDP relaxation (and hence its size) required to certify global optimality is typically larger with the epigraph approach.

When evaluating the epigraph approach, we also observed that it is numerically preferable to replace the inequality constraints $r_i q_i(\mathbf{x}) - p_i(\mathbf{x}) \geq 0$ with equality constraints $r_i q_i(\mathbf{x}) - p_i(\mathbf{x}) = 0$ in the definition of semi-algebraic set $\tilde{\mathbf{K}}$ in (3). For the example of Section 4.1 the epigraph approach with inequalities certifies global optimality at order $k = 5$, whereas the epigraph approach with equalities requires $k = 1$.

As a typical illustration of the issues faced with the epigraph approach consider the example with eighth-degree terms

$$\inf_{\mathbf{x} \in \mathbb{R}^2} f(x) = \sum_{i=1}^{10} \frac{(x_1 + x_2)(x_1^2 + x_1^2 x_2^2 + x_2^4 + i^2) - (ix_2^2 + 1)(x_1^4 + x_2^2 + 2i)}{(x_1^4 + x_2^2 + 2i)(x_1^2 + x_1^2 x_2^2 + x_2^4 + i^2)} \quad (11)$$

which is cooked up to have several local optima and sufficiently high degree to prevent reduction to the same denominator. After a suitable scaling to make critical points fit within the box $[-1, 1]^2$, as required by the moment SDP relaxations formulated in the power basis [4, Section 6.5], the GMP approach yields a certificate of global optimality with $x_1^* = -0.60450$, $x_2^* = -2.2045$, $f^* = -6.2844$ at order $k = 6$ in a few seconds on

a standard PC. In contrast, the epigraph approach does not provide a certificate for an order as high as $k = 10$, requiring more than one minute of CPU time.

4.5 Shekel's foxholes

Consider the modified Shekel foxholes rational function minimization problem [2]

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^N \frac{1}{\sum_{j=1}^n (x_j - a_{ij})^2 + c_i} \quad (12)$$

whose data a_{ij} , c_i , $i = 1, \dots, N$, $j = 1, \dots, n$ can be found in [1, Table 16]. This function is designed to have many local minima, and we know the global minimum in the case $n = 5$, see [1, Table 17]. After a suitable scaling to make critical points fit within the box $[0, 1]^5$, and after addition of a Euclidean ball constraint centered in the box, the GMP approach yields a certificate of global optimality at order $k = 3$ in less than one minute on a standard PC. The extracted minimizer is $x_1^* = 8.0254$, $x_2^* = 9.1483$, $x_3^* = 5.1138$, $x_4^* = 7.6213$, $x_5^* = 4.5638$, which matches with the known global minimizer to four significant digits. This point can be refined if given an initial guess for a local optimization method. If we use a standard quasi-Newton BFGS algorithm, we obtain after a few iterations a point matching the known global minimizer to eight significant digits.

In the case $n = 10$, for which the global minimum is given in [1, Table 17], the GMP approach yields a certificate of global optimality at order $k = 2$ in about 750 seconds of CPU time. Here too, we observe that the extracted minimizer $x_1^* = 8.0249$, $x_2^* = 9.1518$, $x_3^* = 5.1140$, $x_4^* = 7.6209$, $x_5^* = 4.5640$, $x_6^* = 4.7110$, $x_7^* = 2.996$, $x_8^* = 6.1259$, $x_9^* = 0.73424$, $x_{10}^* = 4.9820$ is a good approximation to the minimizer, with four correct significant digits. If necessary, this point can be used as an initial guess for refining with a local solver.

Note that it is not possible to exploit problem sparsity in this case, since all the variables appear in each term in sum (12).

4.6 Rosenbrock's function

Consider the rational optimization problem

$$f^* = \max_{x \in \mathbb{R}^n} \sum_{i=1}^{n-1} \frac{1}{100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 + 1} \quad (13)$$

which has the same critical points as the well-known Rosenbrock problem

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$$

whose geometry is troublesome for local optimization solvers. It can be easily shown that the global maximum $f^* = 1$ of problem (13) is achieved at $x_i^* = 1$, $i = 1, \dots, n$. Our experiments with local optimization algorithms reveal that standard quasi-Newton solvers or functions of the Optimization toolbox for Matlab, called repeatedly with random initial guesses, typically yield local maxima quite far from the global maximum.

With our GMP approach, after exploiting sparsity and adding bound constraints $x_i^2 \leq 16$, $i = 1, \dots, n$, we could solve problem (13) with a certificate of global optimality for n up to 1000. Typical CPU times range from 10 seconds for $n = 100$ to 500 seconds for $n = 1000$.

5 Conclusion

The problem of minimizing the sum of *many* low-degree (typically non-convex) rational fractions on a (typically non-convex) semi-algebraic set arises in several important applications, and notably in computer vision (triangulation, estimation of the fundamental matrix in epipolar geometry) and in systems control (H_2 optimal control with a fixed-order controller of a linear system subject to parametric uncertainty). These engineering problems motivated our work, but the application of our techniques to computer vision and systems control will be described elsewhere. These fractional programming problems being non convex, local optimization approaches yield only upper bounds on the optimum.

In this paper we were interested in computing the global minimum (and possibly global minimizers) or at least, computing valid lower bounds on the global minimum, for fractional programs involving a sum with many terms. We have used a semidefinite programming (SDP) relaxation approach by formulating the rational optimization problem as an instance of the generalized moment problem (GMP). In addition, problem structure can be sometimes exploited in the case where the number of variables is large but sparsity is present. Numerical experiments with our public-domain software GloptiPoly interfaced with off-the-shelf semidefinite programming solvers indicate that the approach can solve problems that can be challenging for state-of-the-art global optimization algorithms. This is consistent with the experiments made in [3] where the (dense) SDP relaxation approach was first applied to (polynomial) optimization problems of computer vision.

For larger and/or ill-conditioned problems, it can happen that GloptiPoly extracts from the moment matrix a minimizer which is not very accurate. It can also happen that GloptiPoly is not able to extract a minimizer, in which case first-order moments approximate the minimizer (provided it is unique, which is generically true for rational optimization). The approximate minimizer can be then input to any local optimization algorithm as an initial guess.

A comparison of our approach with other techniques of global optimization (reported e.g. on Hans Mittelmann's or Arnold Neumaier's webpages) is out of the scope of this paper. We believe however that such a comparison would be fair only if no expert tuning is required for alternative algorithms. Indeed, when using GloptiPoly the only assumption we make is that we know a ball containing the global optimizer. Besides this, our results are fully reproducible (Matlab files reproducing our examples are available upon request) and our SDP relaxations are solved with general-purpose semidefinite programming solvers.

Acknowledgments

We are grateful to Michel Devy, Jean-José Orteu, Tomáš Pajdla, Thierry Sentenac and Rekha Thomas for insightful discussions on applications of real algebraic geometry and SDP in computer vision, and to Josh Taylor for his feedback on the example of section 4.3.

References

- [1] M. M. Ali, C. Khompatraporn, Z. B. Zabinsky. A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems. *J. Global Optim.*, 31(4):635-672, 2005.
- [2] H. Bersini, M. Dorigo, S. Langerman, G. Seront, L. Gambardella. Results of the first international contest on evolutionary optimisation. *IEEE Intl. Conf. Evolutionary Computation*, Nagoya, Japan, 1996.
- [3] F. Kahl, D. Henrion. Globally optimal estimates for geometric reconstruction problems. *IEEE Intl. Conf. Computer Vision*, Beijing, China, 2005.
- [4] D. Henrion, J. B. Lasserre. GloptiPoly: global optimization over polynomials with Matlab and SeDuMi. *ACM Trans. Math. Software*, 29:165–194, 2003.
- [5] D. Henrion, J. B. Lasserre, J. Löfberg. GloptiPoly 3: moments, optimization and semidefinite programming. *Optim. Methods and Software*, 24:761–779, 2009.
- [6] D. Jibetean, E. de Klerk. Global optimization of rational functions: a semidefinite programming approach. *Math. Programming*, 106:93-109, 2006.
- [7] J. B. Lasserre. Convergent SDP relaxations in polynomial optimization with sparsity. *SIAM J. Optim.*, 17:822-843, 2006.
- [8] J. B. Lasserre. A semidefinite programming approach to the generalized problem of moments. *Math. Programming*, 112:65-92, 2008.
- [9] J. B. Lasserre. *Moments, positive polynomials and their applications*, Imperial College Press, London, 2009.
- [10] S. Schaible, J. Shi. Fractional programming: the sum-of-ratios case. *Optim. Methods Software*, 18(2):219–229, 2003.
- [11] S. Waki, S. Kim, M. Kojima, M. Muramatsu. Sums of squares and semidefinite programming relaxations for polynomial optimization problems with structured sparsity. *SIAM J. Optim.*, 17:218-242, 2006.
- [12] H. Waki, S. Kim, M. Kojima, M. Muramatsu, H. Sugimoto. SparsePOP: a sparse semidefinite programming relaxation of polynomial optimization problems. *ACM Trans. Math. Software*, 35:1-13, 2008.