Exact Certification of Global Optimality of Approximate Factorizations Via Rationalizing Sums-Of-Squares with Floating Point Scalars*

Erich Kaltofen
Dept. of Mathematics, NCSU
Raleigh, North Carolina
27695-8205,USA
kaltofen@math.ncsu.edu
http://www.kaltofen.us

Bin Li
Key Laboratory of Mathematics Mechanization
AMSS, Beijing 100190, China
bli@mmrc.iss.ac.cn

Zhengfeng Yang
Dept. of Mathematics, NCSU
Raleigh, North Carolina
27695-8205,USA
zyang4@math.ncsu.edu

Lihong Zhi
Key Laboratory of Mathematics Mechanization
AMSS, Beijing 100190, China
lzhi@mmrc.iss.ac.cn http://www.mmrc.iss.ac.cn/~lzhi

ABSTRACT

We generalize the technique by Peyrl and Parillo [Proc. SNC 2007] to computing lower bound certificates for several well-known factorization problems in hybrid symbolic-numeric computation. The idea is to transform a numerical sum-of-squares (SOS) representation of a positive polynomial into an exact rational identity. Our algorithms successfully certify accurate rational lower bounds near the irrational global optima for benchmark approximate polynomial greatest common divisors and multivariate polynomial irreducibility radii from the literature, and factor coefficient bounds in the setting of a model problem by Rump (up to $n = 14$, factor degree $= 13$).

The numeric SOSes produced by the current fixed precision semi-definite programming (SDP) packages (SeDuMi, SOSTOOLS, YALMIP) are usually too coarse to allow successful projection to exact SOSes via Maple 11's exact linear algebra. Therefore, before projection we refine the SOSes by rank-preserving Newton iteration. For smaller problems the starting SOSes for Newton can be guessed without SDP ("SDP-free SOS"), but for larger inputs we additionally appeal to sparsity techniques in our SDP formulation.

Categories and Subject Descriptors: I.1.2 [Symbolic and Algebraic Manipulation]: Algorithms; G.1.6 [Numerical Analysis]: Global optimization

General Terms: algorithms, experimentation

Keywords: semidefinite programming, sum-of-squares, validated output, approximate factorization, hybrid method

*This material is based on work supported in part by the National Science Foundation under Grants CCF-0514585 and DMS-0532140 (Kaltofen and Yang) and OISE-0456285 (Kaltofen and Zhi).
This research was partially supported by NKBRPC (2004CB318000) and the Chinese National Natural Science Foundation under Grant 10401035 (Li and Zhi).

ISSAC'08, July 20–23, 2008, Hagenberg, Austria.

1. INTRODUCTION

1.1 Motivation

Minimizing the deformations on inexact floating point scalars in the inputs to symbolic computation problems so that the resulting deformed inputs yield non-trivial outputs, such as common roots, factors or sparse interpolants, has resulted in a plethora of numerical optimization algorithms for those hybrid symbolic-numeric tasks (see, e.g., the survey [46]). We have successfully deployed structure preserving total least squares algorithms based on Newton iteration and the method of Lagrangian multipliers for the approximate polynomial greatest common divisor (GCD) problem [16], the approximate multivariate polynomial factorization problem [15], and the sparse multivariate polynomial and rational function interpolation [17] problem. Those optimization techniques efficiently produce local minima, which with randomizing start values and projections experimentally appear to actually be the global optima.

Semidefinite programming (SDP) is a far-reaching generalization to the interior-point methods of linear programming: theoretically, the SDPs can be solved in polynomial-time and can produce approximations to classical combinatorial optimization problems that were unachieved before. In addition and important to our setting, via sum-of-squares (SOS) and truncated moment techniques, SDP is applicable to global polynomial optimization problems (POPs) of the form

$$\left.\begin{array}{ll} \min\limits_{x\in\mathbb{R}^n} & p(x) \\ \text{s. t.} & q_1(x)\geq 0,\ldots,q_l(x)\geq 0 \\ & \text{where } p,q_1,\ldots,q_l\in\mathbb{R}[X_1,\ldots,X_n] \end{array}\right\} \quad (1)$$

(if the q_j are omitted, the problem is unconstrained).

The idea of bringing SDP-based polynomial optimization into hybrid symbolic-numeric computation was discussed

among James Demmel, Erich Kaltofen and Lihong Zhi at the October 2005 BIRS Workshop "Challenges in Linear and Polynomial Algebra in Symbolic Computation Software." We now have a growing body of work [28, 13, 22, 23]. At this moment in time, we can draw the following conclusion: SDP is significantly more complex than our local (upper bound) algorithms, but one also gets more: namely an SOS certificate à la Putinar for the global infimum $r \in \mathbb{R}$ of (1) in the form of polynomials $u_i, v_{j,k} \in \mathbb{R}[X_1, \ldots, X_n]$, $0 \le i \le m$, $1 \le j \le l$, $1 \le k \le \nu_j$ that satisfy the polynomial identity

$$p(X) - r = \sum_{i=1}^{m} u_i(X)^2 + \sum_{j=1}^{l} \left(q_j(X) \cdot \sum_{k=1}^{\nu_j} v_{j,k}(X)^2 \right) \quad (2)$$

Note that the above Positivstellensatz (2) [30] places restrictions on the constraints. Without constraints, not all such polynomials have a polynomial SOS (e.g., Motzkin's polynomial), and either polynomial relaxations or a common square polynomial denominator u_0^2 in the $u_i^2, v_{j,k}^2$ (Emil Artin's theorem) become necessary.

That is what the theory promises. However, an underlying assumption is that the interior-point iteration is performed with sufficiently high bigfloat precision, which current implementations do not support. The SOS certificate is numeric: it is subject to round-off errors of the floating point scalars in the computed equation (the exact solution may have irrational algebraic coefficients) and, more consequentially, to the numerical error from the fixed precision SDP solver itself. In fact, we will demonstrate on Siegfried Rump's model problem how an SDP solver can become quite unstable with scale, see Table 1. In summary, (2) is satisfied only approximately by the polynomials SDP produces.

Our goal is to convert the imprecise and possibly invalid SOS certificate into an exact SOS certificate with exact rational scalars and polynomials, i.e., a proof, by bringing exact symbolic methods into play. Then the floating point precision in the SDP solver can be fixed, and further relaxations such as sparseness and "cheap-and-dirty" heuristics can be introduced without concern for the returned answer. One may even proceed by computing the numeric SOS certificate using Newton iteration directly rather than formulating a corresponding SDP. Since the global optimum can be an algebraic number in an extension of high degree [29], we shall verify a nearby rational lower bound $\tilde{r} \lessapprox r$, $\tilde{r} \in \mathbb{Q}$. We assume that the coefficients of the polynomials p and q_j in (1) are represented exactly, for example as exact rational numbers. If they are floating point numbers, we can take their rational values. If the coefficients are irrational algebraic numbers with an exact representation, our methods also work, see Remark 2. Unlike Rump's and Villard's fully analyzed and validated numerical approach (see [43] and its references), our paradigm allows for unchecked numerical techniques and low precision floating point numbers because our exact rationalization catches all false certificates.

1.2 Used Approach and Results

We can formulate the problem of computing for an absolutely irreducible input polynomial h of total degree t in $\mathbb{R}[Z_1, \ldots, Z_s]$ the nearest polynomial with a (real) factor of given total degree k as an unconstrained polynomial optimization problem [15, 23]:

$$\min_{\substack{h_1, h_2 \in \mathbb{R}[Z_1, \ldots, Z_s], \\ \deg(h_1) = k,\ \deg(h_2) = t-k}} \|h(Z) - h_1(Z)h_2(Z)\|^2 \quad (3)$$

with $n = \binom{s+k}{s} + \binom{s+t-k}{s}$ unknown coefficients. (With $\|\cdot\|$ we always denote the 2-norm of the coefficient vector.) Or we can eliminate the coefficients of h_2 by a symbolic least squares ansatz and obtain a multivariate rational function optimization problem:

$$\min_{x \in \mathbb{R}^n} \frac{f(x)}{g(x)} \quad (\text{where } g(x) > 0 \text{ for all } x \in \mathbb{R}^n) \quad (4)$$

with $n = \binom{s+k}{k}$ [13]. Since any polynomial can always be viewed as a rational function with the denominator as 1, in the following, we focus on the more general problem of minimization of a rational function by SOS:

$$\left.\begin{aligned} r^* := \sup_{r \in \mathbb{R}, W} \ & r \\ \text{s. t.} \ & f(X) - rg(X) = m_d(X)^T \cdot W \cdot m_d(X) \\ & W \succeq 0,\ W^T = W \end{aligned}\right\} \quad (5)$$

where $m_d(X)$ is the column vector of all terms in $X_1, \ldots, X_n$ up to degree d. The dimension of $m_d(X)$ is $\binom{n+d}{d}$. From (5) we know W is a symmetric positive semidefinite real matrix and the program is an SOS. We refer to [33, 31, 19, 28, 12] for description of SOS relaxations and their dual problems.

The problem of finding the rational SOS is equivalent to finding a rational positive semidefinite symmetric matrix W solving SOS problem (5) (see (13) below). Inspired by the method in [34], we start with a numerical solution W computed by semidefinite programming solving problem (5). However, we refine the matrix W further by using Newton iteration. In the next step, we lower r^* to a rational bound $\tilde{r}$, convert W to a rational matrix and project the matrix onto the affine linear hyperplane (cf. (5))

$$\mathcal{X} = \{A \mid A^T = A, f(X) - \tilde{r}g(X) = m_d(X)^T \cdot A \cdot m_d(X)\} \quad (6)$$

denoting the nearest matrix in $\mathcal{X}$ by $\widetilde{W}$. The projection is done by solving a rational least squares problem (11) exactly. The hope is that $\widetilde{W}$ is positive semidefinite, yielding a SOS proof (2) for the lower bound $\tilde{r}$. If $\widetilde{W}$ is not positive semidefinite, then we may increase the precision or reduce $\tilde{r}$ further and try again. It is always possible to compute a valid rational solution using sufficiently many digits, provided the optimal W remains positive semidefinite in a neighborhood [34, Proposition 3.1].

We demonstrate our exact certification strategy on three problems, the approximate greatest common divisor problem, the model problem by Rump, and the approximate factorization problem. Thus we are able to certify for several non-trivial problems from the literature minimum required distances to the nearest solvable problem that are fairly near their upper bounds. In fact, within minutes of computing we can prove a lower bound for our approximate GCD problem [16, Example 4.2] that is accurate to 5 decimal mantissa digits (6 decimal places). The lower bound establishes that the corresponding ill-conditioned Sylvester matrix is structurally well-conditioned. Rump's [38] problem minimizes

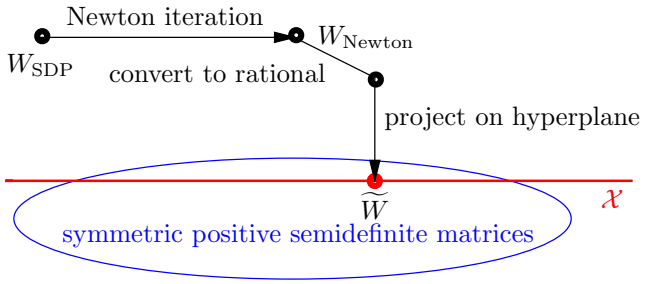

Figure 1: Rationalization of SOS

the factor coefficient bound

$$\mu_n = \min_{P,Q} \frac{1}{B_{n-1}} \quad (\frac{1}{\mu_n} = \max_{P,Q} B_{n-1})$$
$$\text{s. t. } \|P(Z)\|^2 \cdot \|Q(Z)\|^2 = B_{n-1}\|P(Z)\cdot Q(Z)\|^2$$
$$P, Q \in \mathbb{R}[Z] \setminus \{0\}, \deg(P) \le n-1, \deg(Q) \le n-1$$

Mignotte's [25] bound is $\frac{1}{\mu_n} \le \binom{2n-2}{n-1}^2$. We can prove accurate lower bounds for μ_n as high as $n = 14$ and upper bounds at least as large as $n = 63$. Our upper bounds are obtained by Newton-Lagrange optimization with an incremental number of decimal mantissa digits (see also [1]). For $n = 14$, we have $1.62 \cdot 10^{11} < \frac{1}{\mu_{14}} < 3.12 \cdot 10^{11}$ with Digits := 16, while $\binom{26}{13}^2 \approx 1.08 \cdot 10^{14}$. The SOS-based lower bound for μ_{14} is at the limits of our fixed precision (15 decimal mantissa digits) SDP solvers. Finally, we have certified the irreducibility radius of [14, Example 3]. Our certified lower bound computed in under one minute is accurate to 5 decimal mantissa digits (8 decimal places).

Our method has also produced the following exact SOS representation of the non-negative polynomial $\mathcal{D}(\lambda)$ in [5] (the benchmark "Vor1" in Safey El Din's paper in these Proceedings [40]): $16(au+au^2)^2+(ay+a\beta+2auy+4a\beta u-a^2x-a^2\alpha+4a\beta u^2-2a^2\alpha u)^2+(y+\beta+2\beta u-ax-a\alpha-2aux-4a\alpha u-4a\alpha u^2)^2$. The YALMIP numeric sparse SOS + Gauss-Newton refinement + rational projection takes no more than 2 seconds in total.

In section 2 we illustrate how to extend the method in [34] to find the rational sums of squares of polynomials. The initial symmetric positive semidefinite matrix W can be obtained by solving the SDP problem (5). We also explore the possibility of starting at a random symmetric positive semidefinite matrix with given rank as W. We have thus successfully certified, without using SDP, the lower bounds for Rump's problem up to $n = 10$. The size of $m_d(X)$ increases very fast with d and the number of variables. So the sparsity of the SDP problem (5) has to be exploited [36, 32, 44, 20, 27, 23]. See section 3.2.2 for a detailed discussion.

2. RATIONALIZING AN SOS DERIVED FROM SDP AND NEWTON ITERATION

The SOS program (5) can be solved efficiently by algorithms in SOSTOOLS [35], YALMIP [24] and SeDuMi [41]. However, since we are running fixed precision SDP solvers in Matlab, we can only obtain a numerical positive semidefinite matrix W and floating point number r^* which satisfy approximately

$$f(X) - r^* g(X) \approx m_d(X)^T \cdot W \cdot m_d(X), \ W \gtrapprox 0. \qquad (7)$$

So r^* is a lower bound of $f(x)/g(x)$, $x \in \mathbb{R}^n$, approximately! For some applications, such as Rump's model problem (16), due to the numerical error, the computed lower bounds can even be much bigger than upper bounds, see Table 1. This motivates us to consider how to use exact linear algebra tools to certify the lower bounds computed by SDP.

The lower bound $\tilde{r}$ is certified if $\tilde{r}$ and $\widetilde{W}$ hold the following conditions exactly:

$$f(X) - \tilde{r}g(X) = m_d(X)^T \cdot \widetilde{W} \cdot m_d(X), \ \widetilde{W} \succeq 0. \qquad (8)$$

In the following subsections, we start with using Gauss-Newton iterations to refine r^* and W which satisfy (7) approximately, then compute the rational number $\tilde{r}$ and rational positive semidefinite symmetric matrix $\widetilde{W}$ which satisfy (8) exactly. The projection steps are shown in Figure 1.

2.1 Newton Iteration

Let θ denote the backward error:

$$\theta = \|f(X) - rg(X) - m_d(X)^T \cdot W \cdot m_d(X)\|.$$

The floating point number r can be computed by solving the SOS program (5) or by other local optimization methods [6, 15]. In order to derive a satisfactory certified lower bounds, it is important to start with an accurate lower bound, denoted by r^*.

For the refined r^*, the positive semidefinite symmetric matrix W can be obtained by solving the following SOS program:

$$\left.\begin{array}{ll} \inf\limits_{W} & \text{Trace}(W) \\ \text{s. t.} & f(X) - r^* g(X) = m_d(X)^T \cdot W \cdot m_d(X) \\ & W \succeq 0, W^T = W \end{array}\right\} \qquad (9)$$

(here Trace(W) acts as a dummy objective function that is commonly used in SDP for optimization problem without an objective function.)

We expand the quadratic form obtained from the SOS decomposition:

$$f(X) - r^* g(X) \approx \sum_{i=1}^{k} (\sum_{\alpha} c_{i,\alpha} X^{\alpha})^2 \in \mathbb{R}[X]. \qquad (10)$$

Here k is the rank of the matrix W. The rank deficiency of W corresponds to the number of global optima. For some applications, if the number of global optima is known, then k is known too. For Rump's model problem (18), the rank deficiency of W is 1 if n is even; and 2 if n is odd. For the given tolerance, we compute the rank k by using the singular value decomposition of W.

We apply Gauss-Newton iteration to compute $\Delta c_{i,\alpha} X^\alpha$ such that

$$f(X) - r^* g(X) = \sum_{i=1}^{k} (\sum_\alpha c_{i,\alpha} X^\alpha + \Delta c_{i,\alpha} X^\alpha)^2 + O(\sum_{i=1}^{k} (\sum_\alpha \Delta c_{i,\alpha} X^\alpha)^2).$$

We update the matrix W accordingly to $W + \Delta W$ and the iteration is stopped when θ is less than the given tolerance τ. If θ remains greater than the given tolerance τ after several Gauss-Newton iterations, we may increase the precision of the SDP and Gauss-Newton iteration computations or use a smaller r^* and try the computations again.

The total number of X^α in $m_d(X)$ is $\binom{n+d}{d}$. So the computation of Gauss-Newton iteration is very heavy. It is necessary to exploit the sparsity of the SOS program (9). Fortunately, for many optimization problems arising from approximate polynomial computation, the sparsity can be discovered by analyzing the Newton polytope. We show in section 3.2.2 how to explore the sparsity for the problem (18).

Remark 1 It is possible to construct W as a random symmetric positive semidefinite matrix satisfying some given conditions such as rank, sparsity. However, if the backward error θ is large, we may need a big number of Gauss-Newton iterations to reduce θ below τ.

2.2 Rationalizing an SOS

In [34], a Macaulay 2 package is presented to compute an exact SOS decomposition from a numerical solution for non-negative polynomials with rational coefficients. We extend their technique to construct an exact rational SOS decomposition for the polynomial $f(X) - \tilde{r} g(X)$. Then $\tilde{r}$ is certified to be the lower bound of the minimization problem (4).

Suppose W has been refined by Gauss-Newton iterations such that $\|f(X) - r^* g(X) - m_d(X)^T \cdot W \cdot m_d(X)\| < \tau$. We approximate r^* by a nearby rational number $\tilde{r} \lessapprox r^*$ and convert W to a rational matrix. Then we orthogonally project the refined matrix W to the rational matrix $\widetilde{W}$ on the hyperplane $\mathcal{X}$ in (6). The projection is achieved by solving exactly the following least squares problems:

$$\left.\begin{array}{l} \min\limits_{\widetilde{W}} \|W - \widetilde{W}\|_F^2 \\ \text{s. t. } f(X) - \tilde{r} g(X) = m_d(X)^T \cdot \widetilde{W} \cdot m_d(X) \end{array}\right\} \quad (11)$$

It is equivalent to solve a set of smaller least squares problems:

$$\left.\begin{array}{l} \min\limits_{\widetilde{W}} \sum\limits_\alpha \sum\limits_{\beta+\gamma=\alpha} (W_{\beta,\gamma} - \widetilde{W}_{\beta,\gamma})^2 \\ \text{s. t. } f_\alpha - \tilde{r} g_\alpha = \sum\limits_{\beta+\gamma=\alpha} \widetilde{W}_{\beta,\gamma} \end{array}\right\} \quad (12)$$

By solving the least squares problem (12) for each α, we get the minimal rational solution, denoted by $\widetilde{W}$. Then we compute the exact $\mathrm{L^TDL}$-decomposition [7] to check whether $\widetilde{W}$ is a symmetric positive semidefinite matrix. The optimum $\tilde{r}$ is verified as the lower bound if

$$\left.\begin{array}{rl} f(X) - \tilde{r} g(X) = & m_d(X)^T \cdot \widetilde{W} \cdot m_d(X) \\ = & m_d(X)^T \cdot L^T \cdot D \cdot L \cdot m_d(X), \\ & \text{such that } \forall i\colon D_{i,i} \geq 0, \end{array}\right\} \quad (13)$$

where $D_{i,i}$ is the i-th diagonal entry of the diagnal matrix D. If $\widetilde{W}$ is not positive semidefinite, i.e., one entry, usually the last, in the diagonal matrix D is negative, we can decrease $\tilde{r}$ or increase the precision of Newton iterations to repeat the above computation. In our experiments, we usually choose $\rho \in [0.1 \cdot \theta, 0.5 \cdot \theta]$, and lower r^* to $\tilde{r} = r^* - \rho$. In our experience, the matrix $\widetilde{W}$ becomes a positive semidefinite matrix.

Remark 2 The crucial property for verification is that the $\mathrm{L^TDL}$-factorization of $\widetilde{W}$ can be computed by an algorithm using rational arithmetic. The corresponding SOS is exact, but not rational:

$$f(X) - \tilde{r} g(X) = \sum_i (\sqrt{D_{i,i}} \cdot L_i \cdot m_d(X))^2, \quad (14)$$

where $D_{i,i} > 0$ is rational and L_i is the i-th row of L. We do not need this in our certificate, but if all coefficients are rational, a rational SOS exists [10]. If the coefficients of f and g are in an exact (not necessarily totally) real algebraic extension $K = \mathbb{Q}[\eta]/(\varphi(\eta))$, where $\varphi(\eta) \in \mathbb{Q}[\eta]$ and the symbol η represents a designated real root of φ, the certificate is verified exactly in the same manner by arithmetic in K. Note that $\tilde{r}$ is still chosen in $\mathbb{Q}$.

Algorithm *Lower Bound Verification*

Input:
- $f(X_1, \ldots, X_n), g(X_1, \ldots, X_n) \in \mathbb{Q}[X_1, \ldots, X_n]$: the numerator and denominator of a multivariate rational function.
- r^*(optional): the approximate optimum of the minimization problem.
- $\tau \in \mathbb{R}_{>0}$: the given tolerance.

Output:
- $\tilde{r}$: the verified lower bound.

1. Gauss-Newton refinement
   (a) Get an approximate SOS decomposition.
   Case SDP is used to compute W:
   A. If r^* is given, set up SDP to compute W such that W satisfy (7).
   Otherwise, set up SDP to compute r^* and W such that they satisfy (7).
   B. Compute the numerical rank k of W and exploit the sparsity structure of polynomials in the SOS.
   Case SDP is not used to compute W:
   Construct the symmetric positive semidefinite matrix W randomly that satisfies the rank condition and given structures.
   (b) Apply Gauss-Newton method to refine (10) and compute θ.
   (c) If $\theta < \tau$, then get the refined matrix W.
   Otherwise, decrease r^* and go back to step 1(a)A.
2. Compute the exact SOS
   (a) Lower r^* to a rational number $\tilde{r}$ and convert W as a rational matrix.
   (b) Compute the rational matrix $\widetilde{W}$ by solving (11).
   (c) Check whether $\widetilde{W}$ is positive semidefinite. If so, return $\tilde{r}$. Otherwise, choose $\rho \in [0.1 \cdot \theta, 0.5 \cdot \theta]$, let $\tilde{r} = r^* - \rho$ and go back to step 2b.

Remark 3 Our projection method tries to achieve positive semidefiniteness for a rational $\tilde{r}$ and $\widetilde{W}$ such that $\tilde{r}$ is as close as possible to r^*. We apply Gauss-Newton refinement to W for r^* (or a lowered r^*) and project using the even smaller $\tilde{r}$. Refinement with the actual target $\tilde{r}$ seems to bring W too close to the boundary of the cone of positive semidefinite matrices, and orthogonal projection fails to preserve that property.

3. THREE APPLICATIONS

3.1 Approximate Greatest Common Divisors

Approximate Greatest Common Divisors (GCDs) of univariate and multivariate polynomials can be transformed as global minimization of rational function (see [23] and the reference there). An approximate global optimum r^* can be attained by structured total least norm method [16] or SOS relaxations [23, 27]. By Algorithm Lower Bound Verification, we can find the rational number $\tilde{r}$ which is verified as the lower bound.

Example 1 (Example 4.2 in [16]) Consider the polynomials

$$1000Z_1^{10} + Z_1^3 - 1 \text{ and } Z_1^2 - \frac{1}{100}.$$

The local minimum computed by our STLN method is

$$r^* = 0.042157\mathit{9164} \tag{15}$$

which can also be attained by SOS relaxations [23]. We decrease r^* by 4×10^{-8} to $r^* = 0.042157\mathit{8636}$ and then verify this lower bound. We reformulate the approximate GCD problem as rational function minimization problem. The numerator and denominator $f(X), g(X) \in \mathbb{Q}[X]$ can be obtained from the formulation in [18, 11, 23].

The matrix W is computed by solving the SOS program (5) with $\dim(m_d(X)) = 13$. We check that

$$\begin{aligned}\theta &= \|f(X) - r^* g(X) - m_d(X)^T \cdot W \cdot m_d(X)\| \\ &= 1.9430\text{e-}7.\end{aligned}$$

Case 1: Without applying Gauss-Newton iteration, we lower r^* to the rational lower bound

$$\tilde{r}_1 = 1414583 \cdot 2^{-27} \approx 0.042157\mathit{8586}.$$

Our SDP-SOS took 0.02 seconds* to obtain the rational matrix $\widetilde{W}$ by solving (11) with the exact computation in Maple 11 with Digits:=20. By $\mathrm{L^TDL}$-decomposition we find that $\widetilde{W}$ is a positive semidefinite matrix:

$$f(X) - \tilde{r}_1 g(X) = m_d(X)^T \cdot \widetilde{W} \cdot m_d(X) \geq 0.$$

Hence, $\tilde{r}_1$ is verified as the lower bound.

Case 2: Applying the Gauss-Newton iterations, the error is reduced to

$$\begin{aligned}\theta &= \|f(X) - r^* g(X) - m_d(X)^T \cdot W \cdot m_d(X)\| \\ &= 1.8140\text{e-}13.\end{aligned}$$

We lower r^* to the rational lower bound

$$\tilde{r}_2 = 45266661 \cdot 2^{-30} \approx 0.042157\mathit{8633}.$$

Our SDP-SOS took 47.6 seconds to obtain the rational positive semidefinite matrix $\widetilde{W}$; $\tilde{r}_2$ is also verified as the lower bound.

Comparing the two cases, $\tilde{r}_2$ is a bit better than $\tilde{r}_1$. So we let $\tilde{r} = \tilde{r}_2$ as the certified lower bound for this GCD problem. And the local minimum computed by STLN method in [16] is the approximate global optimum.

Note that in [16] we had the globality of (15) verified by minimization with interval arithmetic [45]. The verification of the minimum was important because the Sylvester matrix of (1) is highly ill-conditioned, while the structured distance to singularity is large, a phenomenon that has been studied more since our discovery [2, 21].

3.2 Siegfried Rump's Model Problem

In this section, we solve the Rump's model problem by using sparse semidefinite programming (sparse SDP). The Newton-Lagrange method is applied to refine the global optima computed by SDP. Furthermore, we certify exact lower bounds by rationalizing sum-of-squares (SOS) decompositions.

3.2.1 Problem and Mathematical Background

Rump's model problem [39, 38] is that of computing the global minimum μ_n:

$$\left.\begin{aligned}\mu_n = \min\{\|PQ\|^2 \mid P, Q \in \mathbb{R}[Z], \|P\| = \|Q\| = 1 \\ \text{and } \deg(P) = \deg(Q) = n-1\}.\end{aligned}\right\} \tag{16}$$

For a non-singular system of linear equations $Ax = b$, we denote the Toeplitz condition number by $\kappa^{\mathrm{Toep}}(A, x)$. It characterizes the sensitivity of the solution x with respect to infinitely small Toeplitz structured perturbations of the matrix A and perturbations of b. It has been proved in [37] that the ratio between the Toeplitz and the unstructured condition number satisfies

$$\frac{\kappa^{\mathrm{Toep}}(A,x)}{\kappa(A,x)} = \alpha \frac{\|A^{-1}J\Psi_x\|}{\|A^{-1}\|\|x\|} \geq \frac{1}{\sqrt{n}} \frac{\sigma_{min}(\Psi_x)}{\|x\|},$$

where the matrix Ψ_x is defined by

$$\Psi_x := \begin{bmatrix} x_1 & \dots & x_n & & \\ & \dots & & & \\ & & x_1 & \dots & x_n \end{bmatrix} \in \mathbb{R}^{n \times (2n-1)},$$

$J \in \mathbb{R}^{n \times n}$ is the permutation matrix mapping $(1, \dots, n)^T$ into $(n, \dots, 1)^T$ and $\frac{1}{\sqrt{n}} \leq \alpha \leq \sqrt{2}$. It was shown in [39] that

$$\begin{aligned}\sqrt{\mu_n} = \min_{\|x\|=1} \sigma_{min}(\Psi_x) &= \min_{\|x\|=\|y\|=1} \|\Psi_x y\| \\ &= \min_{\|x\|=\|y\|=1} \|\Psi_y x\|\end{aligned}$$

for all n. The upper bound of μ_n can be computed very efficiently. The challenge is to compute rigorous lower bounds for μ_n. In [38], verified lower bounds are given for n up to 8. In the following, we show that by rationalizing SOS computed from sparse SOS, we can efficiently compute the certified lower bounds for n up to 14.

3.2.2 Rational Function Minimization by Sparse SOS

The problem (16) is equivalent to the rational function

*All reported timings are on a 4 CPU (3GHz Xeon) MacPro with 9GB of memory under Linux 2.6.22-14 (Ubuntu).

n	μ_n^* from fixed prec. SDP	Newton-Lagrange	Rump's upper bound
3	0.111111111111132	0.1111111111111112	0.1111111111111113
4	0.017429173321435 2	0.017429173321432 6	0.017429173321432 9
5	0.002339595548 19155	0.0023359595548155594	0.0023359595548155599
6	0.000289731875 28375	0.0002897318752796807	0.0002897318752796843
7	0.000034185 0701964797	0.0000341850698000 8289	0.0000341850698000 8323
8	0.000003905435 64465773	0.0000039054356497557 21	0.0000039054356497557 845
9	4.3600 4072290608e-007	4.3600165391 81021e-007	4.3600165391 81362e-007
10	4.7839 5278113997e-008	4.7839395687 71179e-008	4.7839395687 72086e-008
11	5.1 8272812166654e-009	5.1787490974 46552e-009	5.1787490974 51150e-009
12	5.54 188889223539e-010	5.5458818311 62859e-010	5.5458818311 73105e-010
13	4.0629943853787 2e-011	5.8866880811 95787e-011	5.8866880812 16679e-011
14	2.2641068186946 0e-010	6.2024449920 01861e-012	6.2024449921 72272e-012

Table 1: Rump's upper bounds and the optima computed by SDP and Newton-Lagrange

minimization:

$$\mu_n = \min\{\frac{\|PQ\|^2}{\|P\|^2\|Q\|^2} \mid P(Z) = \sum_{i=1}^{n} p_i Z^{i-1} \in \mathbb{R}[Z], \\ Q(Z) = \sum_{i=1}^{n} q_i Z^{i-1} \in \mathbb{R}[Z]\}. \quad (17)$$

The numbers of unknowns in coefficients of P, Q are $n(P) = n(Q) = n$. It has been shown in [39] that polynomials P, Q realizing the polynomials achieving μ_n must be symmetric (self-reciprocal) or skew-symmetric. Thus the problem can be rewritten into three optimization problems with three different constraints

$$\begin{array}{llll} p_{n+1-i} = p_i, & q_{n+1-i} = q_i, & 1 \le i \le n, \\ p_{n+1-i} = p_i, & q_{n+1-i} = -q_i, & 1 \le i \le n, \\ p_{n+1-i} = -p_i, & q_{n+1-i} = -q_i, & 1 \le i \le n, \end{array}$$

and the smallest of three minima is equal to μ_n. We are going to minimize the rational function $f(X)/g(X)$ with

$$f(X) = \|PQ\|^2 = \sum_{k=2}^{2n}(\sum_{i+j=k} p_i q_j)^2,$$
$$g(X) = \|P\|^2\|Q\|^2 = (\sum_{i=1}^{n} p_i^2)(\sum_{j=1}^{n} q_j^2)$$

and the variables

$$X = \{p_1, \ldots, p_{n(P)}\} \cup \{q_1, \ldots, q_{n(Q)}\},$$

where $n(P) = n(Q) = \lceil n/2 \rceil$. An SOS-relaxed lower bound of μ_n can be obtained by solving the following SOS program:

$$\begin{array}{ll} \mu_n^* := \sup\limits_{r\in\mathbb{R}, W} & r \\ \text{s. t.} & f(X) - rg(X) = m_d(X)^T \cdot W \cdot m_d(X) \\ & W \succeq 0,\ W^T = W \end{array} \quad (18)$$

here $\deg(f) = \deg(g) = 4$ and $m_d(X)$ is the column vector of all terms up to degree $d = 2$. The dimension of real symmetric matrix W is $\binom{n(P)+n(Q)+2}{2}$ and there are $\binom{n(P)+n(Q)+4}{4}$ equality constraints.

Similar to [23], we exploit the sparsity in the SOS program (18). Let $\mathrm{C}(p)$ denote the convex hull of $\mathrm{supp}(p) = \{\alpha \mid p_\alpha \neq 0\}$ for polynomial $p(X) = \sum_\alpha p_\alpha X^\alpha$. Suppose

$$f(X) - rg(X) = m_d(X)^T \cdot W \cdot m_d(X) = \sum_k u_k(X)^2.$$

Let the polynomial $h(X) = \sum_{i=1}^{n(P)} \sum_{j=1}^{n(Q)} p_i q_j$, and X^α be any term in the SOS, then

$$\alpha \in \frac{1}{2}\mathrm{C}(f(X) - rg(X)) \subseteq \frac{1}{2}\mathrm{C}(h(X)^2) = \mathrm{C}(h(X)).$$

According to the property of convex hulls, there exist non-negative numbers $\lambda_{i,j}$ for $1 \le i \le n(P), 1 \le j \le n(Q)$ such that

$$X^\alpha = \prod_{i,j}(p_i q_j)^{\lambda_{i,j}} = \prod_i p_i^{\sum_j \lambda_{i,j}} \prod_j q_j^{\sum_i \lambda_{i,j}}, \ \sum_{i,j} \lambda_{i,j} = 1.$$

So X^α must be $p_i q_j$ for some i, j. The sparse SOS program of the rational function minimization problem (17) is:

$$\begin{array}{ll} \mu_n^* := \sup\limits_{r\in\mathbb{R}, W} & r \\ \text{s. t.} & f(X) - rg(X) = m_{\mathcal{G}}(X)^T \cdot W \cdot m_{\mathcal{G}}(X) \\ & W \succeq 0,\ W^T = W \end{array} \quad (19)$$

where $m_{\mathcal{G}}(X)$ is the column vector containing all terms $p_i q_j$ for $1 \le i \le n(P), 1 \le j \le n(Q)$. The dimension of the real symmetric matrix W is $n(P)n(Q)$ and the number of equality constraints is $\binom{n(P)+1}{2}\binom{n(Q)+1}{2}$. For $n = 14$, the sparse SOS program (19) has 784 equality constraints and $\dim(W) = 49$ while the SOS program (18) has 3060 equality constraints and $\dim(W) = 120$.

The SOS program (19) can be solved efficiently by algorithms in SOSTOOLS [35], YALMIP [24] and SeDuMi [41]. However, we can see from the Table 1, the lower bounds computed by the fixed precision SDP solver can be even larger than the upper bounds given in [38]. There are different techniques to improve the output of SDP solver, for example, Newton iteration.

Again, let $f(X) = \|PQ\|^2$, $c_1(X) = \|P\|^2 - 1$ and $c_2(X) = \|Q\|^2 - 1$, we apply Newton-Lagrange method to solve the equality-constrained problem

$$\begin{array}{ll} \min\limits_x & f(x) \\ \text{s. t.} & c_1(x) = c_2(x) = 0. \end{array}$$

The initial values of X are the coefficients of the polynomials P, Q computed by solving the sparse SOS program (19).

From the Table 1, we can see that the global optima computed by solving the sparse SOS program (19) in Matlab and refined by Newton-Lagrange method in Maple 11 with Digits := 16 are strictly smaller than Rump's upper bounds [38]. Actually, we can apply the Newton-Lagrange method to any initial guess for the polynomials P and Q, it is not necessary to start with solving the SOS program (19).

n	time(s)	Digits	certified lower bound	Rump's lower bound
3	0.028	20	0.1111111111111*1111*	0.111111111111*083*
4	0.368	20	0.017429173321432*651*	0.017429173321431*74*
5	8.128	20	0.00233959554815*590*	0.00233959554815*278*
6	182.8	20	0.0002897318752796*799*	0.0002897318752795*867*
7	837.4	20	0.000034185069800*8202*	0.000034185069800*4407*
8	2.112	15	0.00000390543564*455700*	0.00000390543564*743504*
9	7.008	15	4.36001*623918100*e-007	?
10	34.14	15	4.78393*556877000*e-008	?
11	27.93	15	5.17*774909740000*e-009	?
12	174.1	15	5.5*1588183110000*e-010	?
13	181.2	15	5.*78668808100000*e-011	?
14	1556	15	*3.20244499200000*e-012	?

Table 2: The certified lower bounds and Rump's lower bounds

However, the output of (19) provides us an initial positive semidefinite matrix W which can be used in the following lower bound certification.

Remark 4 Boyd and Collins have investigated the maximization of the single factor coefficient bound for integer polynomials with respect to infinity norm [3, and its references]. Collins establishes that the ratio of (the minimal height among irreducible factors)/(height of their product) can be ≥ 2.2005. In [1] we compute from the Rump polynomials (see Table 1) with $n = 67$ an integer polynomial of degree 132 with the record-breaking ratio ≥ 3.30846991.

3.2.3 Lower Bound Certification

Given the rational number $r_n = \mu_n^* - \rho_n$ for the small positive number ρ_n. We use the matrix W computed by solving the sparse SOS program (19) as the initial approximate positive semidefinite matrix. Applying the Newton iteration method to the system of equations obtained from the sparse SOS decomposition

$$\begin{aligned} f(X) - r_n g(X) &= m_{\mathcal{G}}(X)^T \cdot W \cdot m_{\mathcal{G}}(X) \\ &= \textstyle\sum_k (\sum_\alpha c_{k,\alpha} X^\alpha)^2 \in \mathbb{R}[X], \end{aligned}$$

we get the refined positive semidefinite matrix W. The numerical matrix W is converted to the rational matrix W. By solving the least squares problem

$$\begin{aligned} &\min_{\widetilde{W}} \sum_{i,j} (w_{i,j} - \tilde{w}_{i,j})^2 \\ &\text{s. t. } f(X) - r_n g(X) = m_{\mathcal{G}}(X)^T \cdot \widetilde{W} \cdot m_{\mathcal{G}}(X) \end{aligned}$$

we project W to the hyperplane defined by the SOS decomposition. If the computed projected rational matrix $\widetilde{W}$ is positive semidefinite, then we get the rational SOS decomposition and r_n is the certified lower bound for μ_n. If it's not, then we can increase the precision for solving SDP and Newton iteration or try smaller r_n.

In the Table 2, we compare the certified lower bounds obtained by our algorithm applying Gauss-Newton iterations with the lower bounds given in [38]. For $n \leq 7$ we used Maple 11's Digits:=20, while for $n \geq 8$ we used machine precision. The case of $n = 14$ is our most difficult certificate (see also Remark 3). The numerators and denominators of its rational numbers have up to 945 decimal digits: http://www4.ncsu.edu/~kaltofen/software/certif/model14_SOS.txt.

Remark 5 We have also tried to avoid SDP altogether and to start with a random positive semidefinite matrix with rank deficiency 1 for n being even and 2 for n being odd. For n up to 10, we can certify the same lower bounds listed in Table 2.

3.3 Approximate Polynomial Factorizations

We can formulate the problem of computing for an absolutely irreducible input polynomial of total degree t in $\mathbb{R}[Z_1, \ldots, Z_s]$ the nearest polynomial with a factor P of given total degree k and cofactor Q of given total degree $t - k$ as a polynomial optimization problem $\min_{X \in \mathbb{R}^n} f(X)$ with $n = \binom{s+k}{s} + \binom{s+t-k}{s}$. It has been pointed out in [13], without exploring the sparsity of the SOS problem (5), it seems difficult to certify the lower bound even for Example 3 in [14], which originally was posed in [26, Example 1]. But the sparsity of the SOS problem for the approximate factorization can been exploited (see [23]). We have the following sparse SOS program:

$$\left.\begin{aligned} r^* := \sup_{r \in \mathbb{R}, W} \; & r \\ \text{s. t.} \quad & f(X) - r = m_{\mathcal{G}}(X)^T \cdot W \cdot m_{\mathcal{G}}(X) \\ & W \succeq 0,\ W^T = W \end{aligned}\right\} \quad (20)$$

where $m_{\mathcal{G}}(X)$ is the column vector containing 1 and all terms $p_i q_j$ for $1 \leq i \leq \binom{s+k}{s}$ and $1 \leq j \leq \binom{s+t-k}{s}$, where p_i and q_j are unknown coefficients of polynomials P and Q respectively.

Example 2 (Example 3 in [14]) Consider the polynomial

$$(Z_1^2 + Z_2 Z_1 + 2Z_2 - 1)(Z_1^3 + Z_2^2 Z_1 - Z_2 + 7) + \frac{1}{5} Z_1$$

we are going to certify the lower bounds for polynomial having a degree one factor or a degree two factor. The global minima r^* computed by solving the sparse SOS program (20) is 0.2947781736518 for $k = 1$, and 0.0004137*02070* for $k = 2$. The corresponding upper bound in [15, Table 1, Example 1] is 0.0004137*0181014226*, which is lower than the numeric SOS bound. The objective function $f(X) \in \mathbb{Q}[X]$ can be obtained from the formulation (3) (see also in [23]).

For $k = 1$, the matrix W is computed by solving the sparse SOS program (20) with $\dim(m_{\mathcal{G}}(X)) = 46$. We check that

$$\theta = \|f(X) - r^* - m_{\mathcal{G}}(X)^T \cdot W \cdot m_{\mathcal{G}}(X)\| = 1.06591917\text{e-}11.$$

Without applying the Gauss-Newton iterations, we lower r^* to a rational number

$$\tilde{r} = 633031307 \cdot 2^{-31} \approx 0.2947781733237$$

and obtain the rational matrix $\widetilde{W}$ by solving (11) with the exact computation in Maple 11. By L^TDL-decomposition we find that $\widetilde{W}$ is a positive semidefinite matrix:

$$f(X) - \tilde{r}g(X) = m_{\mathcal{G}}(X)^T \cdot \widetilde{W} \cdot m_{\mathcal{G}}(X) \geq 0.$$

Hence, $\tilde{r}$ is verified as the lower bound. And the local minimum computed by solving the sparse SOS program (20) is the approximate global optimum. The algorithm takes 0.56 second to certify the lower bound.

For $k = 2$, the matrix W is computed by solving the sparse SOS program (20) with $\dim(m_{\mathcal{G}}(X)) = 61$. We check that

$$\theta = \|f(X) - r^* - m_{\mathcal{G}}(X)^T \cdot W \cdot m_{\mathcal{G}}(X)\| = 1.04543742\text{e-}11.$$

Without applying the Gauss-Newton iterations, we lower r^* to a rational number

$$\tilde{r} = 111052 \cdot 2^{-28} \approx 0.00041370\textit{0938}$$

and obtain the rational matrix $\widetilde{W}$ by solving (11) with the exact computation in Maple 11. By L^TDL-decomposition we find that $\widetilde{W}$ is a positive semidefinite matrix:

$$f(X) - \tilde{r}g(X) = m_{\mathcal{G}}(X)^T \cdot \widetilde{W} \cdot m_{\mathcal{G}}(X) \geq 0.$$

Hence, $\tilde{r}$ is verified as the lower bound. And the local minimum computed by the approximate factorizer is the approximate global optimum. The algorithm takes 2.56 seconds to certify the lower bound. This is, due to 61 squares, our largest certificate. The numerators and denominators of its rational numbers have up to 1132 decimal digits: http://www4.ncsu.edu/~kaltofen/software/certif/factor_SOS.txt.

Remark 6 Even though our certificates seem to contain rational numbers with many digits in their numerators and denominators (their logarithmic height), the exact linear algebra (Steps 2b and 2c) does not contribute significantly to the running time. In fact, once a floating point matrix W that projects to a positive semidefinite rationalization $\widetilde{W}$ is found (see Remark 3), our algorithm completes quickly, in part because very fast exact linear algebra methods are available (see, e.g., [8, 47]). In fact, assuming that the floating point numbers in the numeric SOS are reasonably bounded in magnitude, the number of digits in the rational scalars of the certificate grow no more than quadratically (least squares + L^TDL factorization) in the dimension of the W matrix (see (14)). Note that if the rank of W, i.e., the number of squares in the SOS, is much smaller than the dimension (see Step 1a), the number of digits is also proportionally less. The best worst case logarithmic height bounds for exact symbolic methods appear to grow exponentially in the number of variables in the arising polynomial systems [4]. The logarithmic height of our certificates is reduced because we certify a nearby rational lower bound of small height and we project an SOS of fixed floating point precision. The latter places a form of well-conditionedness restriction on the optimization problems that we currently can certify.

4. CONCLUSION

By rationalizing a numeric sum-of-squares representation to an exact certificate we can eliminate numerical inaccuracies from our stated lower bounds. We have certified lower bounds for several problems in hybrid symbolic-numeric computation that previously remained unproven. The limiting requirement of our approach is that the rationalized moment matrix remains positive semidefinite, so that an exact sum-of-squares representation can be obtained. We achieve the former by refining SDP-based numeric SOSes by rank and structure-preserving Gauss-Newton iteration, and the latter by relaxing the lower bound. It is not clear that one needs SDP to seed the Gauss-Newton iteration, or that orthogonal projection is the best for preserving positive semidefiniteness (see Remark 3). Finally, the SOSes can be tightened to Artin-style rational function sums-of-squares, or so-called "big-ball" constraints can be added (we know non-negative polynomials f for which $f + \delta$ are not SOSes for all real $\delta \geq 0$). We plan to investigate those directions further, and we can freely deploy wild and unanalyzed numerical optimization heuristics, because our truly hybrid paradigm yields exact symbolic certificates that leave no doubt.

Acknowledgments: We thank Kartik Sivaramakrishnan for suggesting the exploitation of sparsity in our SDPs.

Lihong Zhi is also grateful to the IMA in Minneapolis for hosting her during the winter quarter of the IMA thematic year on applications of algebraic geometry.

Note added Sep. 8, 2009: The trailing digits of the floating point entries for our rational lower bounds in Table 2, Column 4, $n = 4$ to 7, have been changed so that all digits in all values constitute proven lower bounds.

5. REFERENCES

[1] Boyd, D., Kaltofen, E., and Zhi, L. Integer polynomials with large single factor coefficient bounds. In preparation, 2008.

[2] Chèze, G., and Yakoubsohn, J.-C. Distance computation to the resultant variety, Nov. 2007. Talk at the Journées 2007 da l'ANR GECKO, URL: http://www-sop.inria.fr/galaad/conf/07gecko/Yakoubshon.J.C.pdf.

[3] Collins, G. E. Single-factor coefficient bounds. *J. Symbolic Comput. 38*, 6 (2004), 1507–1521.

[4] Dahan, X., and Schost, É. Sharp estimates for triangular sets. In Gutierrez [9], pp. 103–110.

[5] Everett, H., Lazard, D., Lazard, S., and Safey El Din, M. The Voronoi diagram of three lines in r^3. In *SoCG '07: Proceedings of the 23-rd Annual Symposium on Computational Geometry* (2007), ACM, New York, USA, pp. 255–264.

[6] Gao, S., Kaltofen, E., May, J. P., Yang, Z., and Zhi, L. Approximate factorization of multivariate polynomials via differential equations. In Gutierrez [9], pp. 167–174.

[7] Golub, G. H., and Van Loan, C. F. *Matrix Computations*, third ed. Johns Hopkins University Press, Baltimore, Maryland, 1996.

[8] Gregory, B., and Kaltofen, E. Analysis of the binary complexity of asymptotically fast algorithms for linear system solving. *SIGSAM Bulletin 22*, 2 (Apr. 1988), 41–49.

[9] Gutierrez, J., Ed. *ISSAC 2004 Proc. 2004 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2004), ACM Press.

[10] Hillar, C. Sums of polynomial squares over totally real fields are rational sums of squares. *Proc. American Math. Society* (2008). To appear. URL: http://www.math.tamu.edu/~chillar/files/totallyrealsos.pdf.

[11] Hitz, M. A., and Kaltofen, E. Efficient algorithms for computing the nearest polynomial with constrained roots. In *Proc. 1998 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'98)* (New York, N. Y., 1998), O. Gloor, Ed., ACM Press, pp. 236–243.

[12] Jibetean, D., and de Klerk, E. Global optimization of rational functions: a semidefinite programming approach. *Math. Program. 106*, 1 (2006), 93–109.

[13] Kaltofen, E., Li, B., Sivaramakrishnan, K., Yang, Z., and Zhi, L. Lower bounds for approximate factorizations via semidefinite programming (extended abstract). In Verschelde and Watt [42], pp. 203–204.

[14] Kaltofen, E., and May, J. On approximate irreducibility of polynomials in several variables. In *ISSAC 2003 Proc. 2003 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2003), J. R. Sendra, Ed., ACM Press, pp. 161–168.

[15] Kaltofen, E., May, J., Yang, Z., and Zhi, L. Approximate factorization of multivariate polynomials using singular value decomposition. *J. Symbolic Comput. 43*, 5 (2008), 359–376.

[16] Kaltofen, E., Yang, Z., and Zhi, L. Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials. In *ISSAC MMVI Proc. 2006 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2006), J.-G. Dumas, Ed., ACM Press, pp. 169–176.

[17] Kaltofen, E., Yang, Z., and Zhi, L. On probabilistic analysis of randomization in hybrid symbolic-numeric algorithms. In Verschelde and Watt [42], pp. 11–17.

[18] Karmarkar, N. K., and Lakshman Y. N. On approximate GCDs of univariate polynomials. *J. Symbolic Comput. 26*, 6 (1998), 653–666. Special issue on Symbolic Numeric Algebra for Polynomials S. M. Watt and H. J. Stetter, editors.

[19] Lasserre, J. B. Global optimization with polynomials and the problem of moments. *SIAM J. on Optimization 11*, 3 (2000), 796–817.

[20] Lasserre, J. B. Global SDP-relaxations in polynomial optimization with sparsity. *SIAM J. on Optimization 17*, 3 (2006), 822–843.

[21] Li, B., Liu, Z., and Zhi, L. Structured condition numbers of Sylvester matrices (extended abstract), Dec. 2007. Presented at MACIS 2007, URL: http://www-spiral.lip6.fr/MACIS2007/Papers/submission_17.pdf.

[22] Li, B., Nie, J., and Zhi, L. Approximate GCDs of polynomials and SOS relaxation. In Verschelde and Watt [42], pp. 205–206.

[23] Li, B., Nie, J., and Zhi, L. Approximate GCDs of polynomials and sparse SOS relaxations. Manuscript, 16 pages. Submitted, 2007.

[24] Löfberg, J. YALMIP : A toolbox for modeling and optimization in MATLAB. In *Proc. IEEE CCA/ISIC/CACSD Conf.* (Taipei, Taiwan, 2004). URL: http://control.ee.ethz.ch/~joloef/yalmip.php.

[25] Mignotte, M. Some useful bounds. In *Computer Algebra*, B. Buchberger, G. Collins, and R. Loos, Eds., 2 ed. Springer Verlag, Heidelberg, Germany, 1982, pp. 259–263.

[26] Nagasaka, K. Towards certified irreducibility testing of bivariate approximate polynomials. In *Proc. 2002 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'02)* (New York, N. Y., 2002), T. Mora, Ed., ACM Press, pp. 192–199.

[27] Nie, J., and Demmel, J. Sparse SOS relaxations for minimization functions that are summation of small polynomials, 2007.

[28] Nie, J., Demmel, J., and Gu, M. Global minimization of rational functions and the nearest GCDs. *ArXiv Mathematics e-prints* (Jan. 2006). URL: http://arxiv.org/pdf/math/0601110.

[29] Nie, J., Ranestad, K., and Sturmfels, B. The algebraic degree of semidefinite programming, 2006. URL: http://www.citebase.org/abstract?id=oai:arXiv.org:math/0611562.

[30] Nie, J., and Schweighofer, M. On the complexity of Putinar's Positivstellensatz. *J. Complexity 23* (2007), 135–70.

[31] Parrilo, P. A. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. PhD thesis, California Institute of Technology, May 2000. URL: http://www.cds.caltech.edu/~pablo/.

[32] Parrilo, P. A. Exploiting algebraic structure in sum of squares programs. *Lecture Notes in Control and Information Sciences 312* (2005), 181–194.

[33] Parrilo, P. A., and Sturmfels, B. Minimizing polynomial functions. *DIMACS series in discrete mathematics and theoretical computer 60* (2003), 83. URL: http://www.citebase.org/abstract?id=oai:arXiv.org:math/0103170.

[34] Peyrl, H., and Parrilo, P. A. A Macaulay 2 package for computing sum of squares decompositions of polynomials with rational coefficients. In Verschelde and Watt [42], pp. 207–208.

[35] Prajna, S., Papachristodoulou, A., and Parrilo, P. A. SOSTOOLS: Sum of squares optimization toolbox for MATLAB. URL: http://www.cds.caltech.edu/sostools.

[36] Reznick, B. Extremal PSD forms with few terms. *Duke Mathematical Journal 45*, 2 (1978), 363–374.

[37] Rump, S. M. Structured perturbations part I: Normwise distances. *SIAM J. Matrix Anal. Applic. 25*, 1 (2003), 1–30.

[38] Rump, S. M. Global optimization: a model problem, 2006. URL: http://www.ti3.tu-harburg.de/rump/Research/ModelProblem.pdf.

[39] Rump, S. M., and Sekigawa, H. The ratio between the Toeplitz and the unstructured condition number, 2006. To appear. URL: http://www.ti3.tu-harburg.de/paper/rump/RuSe06.pdf.

[40] Safey El Din, M. Computing the global optimum of a multivariate polynomial over the reals. In *ISSAC 2008 Proc. 2008 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2008), D. Jeffrey, Ed., ACM Press. These Proceedings.

[41] Sturm, J. F. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software 11/12* (1999), 625–653.

[42] Verschelde, J., and Watt, S. M., Eds. *SNC'07 Proc. 2007 Internat. Workshop on Symbolic-Numeric Comput.* (New York, N. Y., 2007), ACM Press.

[43] Villard, G. Certification of the QR factor R and of lattice basis reducedness. In *ISSAC 2007 Proc. 2007 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2007), C. W. Brown, Ed., ACM Press, pp. 361–368.

[44] Waki, H., Kim, S., Kojima, M., and Muramatsu, M. Sums of squares and semidefinite programming relaxations for polynomial optimization problems with structured sparsity. *SIAM Journal on Optimization 17* (2006), 218–242.

[45] Zhang, T., Xiao, R., and Xia, B. Real soultion isolation based on interval Krawczyk operator. In *Proc. of the Seventh Asian Symposium on Computer Mathematics* (Seoul, South Korea, 2005), S. Pae and H. Park, Eds., Korea Institute for Advanced Study, pp. 235–237. Extended abstract.

[46] Zhi, L. Numerical optimization in hybrid symbolic-numeric computation. In Verschelde and Watt [42], pp. 33–35.

[47] Zhou, W., and Jeffrey, D. J. Fraction-free matrix factors: new forms for LU and QR factors. In *Frontiers of Computer Science in China* (2008). To appear. URL http://www.apmaths.uwo.ca/~djeffrey/Offprints/FFLUQR.pdf.