Approximate GCDs of polynomials and sparse SOS relaxations

Bin Li [a,*] Jiawang Nie [b] Lihong Zhi [a]

[a] *Key Lab of Mathematics Mechanization, AMSS, Beijing 100190 China*

[b] *Department of Mathematics, UCSD, La Jolla, CA 92093 USA*

Abstract

The problem of computing approximate GCDs of several polynomials with real or complex coefficients can be formulated as computing the minimal perturbation such that the perturbed polynomials have an exact GCD of given degree. We present algorithms based on SOS (Sums Of Squares) relaxations for solving the involved polynomial or rational function optimization problems with or without constraints.

Key words: greatest common divisor, sums of squares, semidefinite programming, global minimization.

1 Introduction

The problem of computing approximate GCDs of several polynomials $f_1, \ldots, f_s \in F[z_1, \ldots, z_t]$, where F is $\mathbb{R}$ or $\mathbb{C}$ can be written as computing the minimal perturbation such that the perturbed polynomials have an exact GCD of total degree $k \geq 1$,

$$r^* := \min_{p, u_1, \ldots, u_s} \|f_1 - p \cdot u_1\|_2^2 + \|f_2 - p \cdot u_2\|_2^2 + \cdots + \|f_s - p \cdot u_s\|_2^2 \qquad (1)$$

where $p, u_1, \ldots, u_s \in F[z_1, \ldots, z_t]$ are polynomials with the total degrees $\mathrm{tdeg}(p) = k$, $\mathrm{tdeg}(p \cdot u_i) \leq d_i = \mathrm{tdeg}(f_i)$ for $1 \leq i \leq s$. The minimization problem has many different formulations, and various numeric optimization techniques have been proposed, see (Chin et al., 1998; Kaltofen et al., 2006a) and references therein. The optimization problem has a global solution under

* Corresponding author.

Email addresses: bli@mmrc.iss.ac.cn (Bin Li), njw@math.ucsd.edu (Jiawang Nie), lzhi@mmrc.iss.ac.cn (Lihong Zhi).

certain conditions given in (Kaltofen et al., 2006b). In particular, an algorithm based on global minimization of rational function was proposed in (Karmarkar and Lakshman Y. N., 1996, 1998) to compute approximate GCDs of univariate polynomials. The most expensive part of their algorithm is to find all the real solutions of two bivariate polynomials of high degrees. It has been shown in (Nie et al., 2008) that SOS (Sums Of Squares) relaxation (Lasserre, 2001; Parrilo, 2000) can be used to find the global minimum of the rational function that arises from the approximate GCD computation. The SOS programs can be solved by reformulating them as semidefinite programs (SDP), which in turn are solved efficiently by using interior point methods (Nesterov and Nemirovskii, 1994; Vandenberghe and Boyd, 1996; Wolkowicz et al., 2000). In the following sections, we apply SOS relaxations to solve different optimization problems formulated in (Chin et al., 1998; Karmarkar and Lakshman Y. N., 1996, 1998; Kaltofen et al., 2006a; Nie et al., 2008). The sparsity of the optimization problem has also been exploited.

2 Minimization problems

In this section, we formulate the approximate GCD problem as polynomial or rational function minimization problem with or without constraints. The SOS relaxations are used to solve these optimization problems. We refer to (Parrilo and Sturmfels, 2003; Parrilo, 2000; Lasserre, 2001; Nie et al., 2008; Jibetean and de Klerk, 2006) for description of SOS relaxations and their dual problem.

2.1 Polynomial minimization problem

The minimization problem (1) is a nonlinear least squares problem. As shown in (Chin et al., 1998), if a good initial guess is taken, then Newton-like optimization method or Levenberg-Marquardt method can converge very fast to the global optimum. However, if we start with poor initial guess, then these methods may converge to local minimum after taking a large number of iterations.

An entirely different approach was introduced by Shor (Shor, 1987; Shor and Stetsyuk, 1997) and further developed by Parrilo (Parrilo, 2000; Parrilo and Sturmfels, 2003) and Lasserre (Lasserre, 2001). The idea is to express the problem (1) as a polynomial minimization problem $r^* = \min_{X \in \mathbb{R}^n} f(X)$ and relax it to the following SOS program:

$$\left.\begin{array}{rl} r^*_{sos} := \sup\limits_{r\in\mathbb{R},W} & r \\ s.t. & f(X) - r = m_d(X)^T W m_d(X), \\ & W \succeq 0, \end{array}\right\} \tag{2}$$

where W is a real symmetric positive semidefinite matrix.

The objective polynomial corresponding to the minimization problem (1) is

$$f(X) = \sum_{i=1}^{s} \|f_i - p \cdot u_i\|_2^2 = \sum_{i=1}^{s} \sum_{|\alpha|\le d_i} |f_{i,\alpha} - \sum_{\beta+\gamma=\alpha} p_\beta u_{i,\gamma}|^2. \tag{3}$$

Denote the numbers of indeterminates in the coefficients of $p, u_1, \ldots, u_s$ by $n(p), n(u_1), \ldots, n(u_s)$ respectively (see Remark 2.1 for details). Then $m_d(X)$ is the column vector of all monomials up to degree $d = \lceil \frac{\mathrm{tdeg}(f)}{2} \rceil = 2$ in the variables

$$X = \{p_1, \ldots, p_{n(p)}\} \cup (\bigcup_{i=1}^{s} \{u_{i,1}, \ldots, u_{i,n(u_i)}\}). \tag{4}$$

The number of variables is $n = n(p) + \sum_{i=1}^{s} n(u_i)$. The length of real symmetric matrix W is $\binom{n+2}{2}$ and there are $\binom{n+4}{4}$ equality constraints in (2).

Remark 2.1 If $F = \mathbb{R}$, the coefficients of p, u_i are real numbers and therefore $n(p) = \binom{t+k}{t}, n(u_i) = \binom{t+d_i-k}{t}$. If $F = \mathbb{C}$, we can assume that at least one coefficient of p is a real number and have $n(p) = 2\binom{t+k}{t} - 1, n(u_i) = 2\binom{t+d_i-k}{t}$ by separating real and imaginary parts of each complex coefficient. In the univariate case ($t = 1$), we can assume that p is monic, then $n(p)$ is k in the real case or $2k$ in the complex case.

Write $f(X) = \sum_\alpha f_\alpha X^\alpha$, then the dual SDP problem of the SOS program (2) can be described as (Lasserre, 2001):

$$\left.\begin{array}{rl} r^*_{mom} := \inf\limits_{y} & \sum\limits_{\alpha} f_\alpha y_\alpha \\ s.t. & y_{0,\ldots,0} = 1, \\ & M_d(y) \succeq 0, \end{array}\right\} \tag{5}$$

where $M_d(y) := (y_{\alpha+\beta})_{0\le|\alpha|,|\beta|\le d}$ is called the d-th moment matrix of the real vector y. The SOS program (2) has a feasible solution with $r = 0$, and $r^* \ge r^*_{sos} = r^*_{mom}$ according to (Lasserre, 2001). When the computed moment matrix $M_d(y^*)$ satisfies some flat extension conditions, the global minimum is achieved and some global minimizers can be extracted numerically by solving an eigenvalue problem (Henrion and Lasserre, 2005).

Let $\mathbf{f_i}, \mathbf{u_i}, \mathbf{p}$ be the coefficient vectors of polynomials f_i, u_i, p respectively, and let $A_i = A_i(\mathbf{p})$ be the convolution matrices such that $A_i\mathbf{u_i}$ produces the coefficient vector of $p \cdot u_i$. Then the straight-forward formulation of the minimization problem (1) can be written as

$$\min_{\mathbf{p},\mathbf{u_1},\ldots,\mathbf{u_s}} \|\mathbf{f_1} - A_1\mathbf{u_1}\|_2^2 + \cdots + \|\mathbf{f_s} - A_s\mathbf{u_s}\|_2^2. \tag{6}$$

If we fix the coefficients of p, the minimum is achieved at

$$\mathbf{u_i} := (A_i^* A_i)^{-1} A_i^* \mathbf{f_i},\ 1 \le i \le s, \tag{7}$$

and the minimization problem becomes

$$\min_{\mathbf{p}} \sum_{i=1}^{s} (\mathbf{f_i}^* \mathbf{f_i} - \mathbf{f_i}^* A_i (A_i^* A_i)^{-1} A_i^* \mathbf{f_i}). \tag{8}$$

Here and hereafter A_i^* and $\mathbf{f_i}^*$ denote the conjugate transpose of A_i and $\mathbf{f_i}$ respectively. This is an unconstrained minimization problem of rational function with the positive denominator

$$\operatorname{lcm}(\det(A_1^* A_1), \ldots, \det(A_s^* A_s)).$$

It generalizes the formulations presented in (Chin et al., 1998; Hitz and Kaltofen, 1998; Karmarkar and Lakshman Y. N., 1996, 1998; Zhi and Wu, 1998; Zhi et al., 2004) for computing approximate GCDs of univariate polynomials and in (Hitz et al., 1999) for computing nearest bivariate polynomials with a linear (or fixed degree) factor.

Express the minimization problem (8) as $\min_{X\in\mathbb{R}^n} \frac{f(X)}{g(X)}$, where $f(X), g(X) \in \mathbb{R}[X_1, \ldots, X_n]$ and $g(X)$ is a real positive definite polynomial. Similar to the polynomial minimization problem, it can be transferred to a constrained SOS program (Nie et al., 2008):

$$\left.\begin{array}{rl} r_{sos}^* := \sup\limits_{r\in\mathbb{R},W} & r \\ s.t. & f(X) - rg(X) = m_d(X)^T W m_d(X), \\ & W \succeq 0. \end{array}\right\} \tag{9}$$

Here $X = \{p_1, \ldots, p_{n(p)}\}$, and $m_d(X)$ is the column vector of all monomials up to degree $d = \lceil \frac{\max(\operatorname{tdeg}(f), \operatorname{tdeg}(g))}{2} \rceil$ where $\operatorname{tdeg}(f) \le \operatorname{tdeg}(g) \le 2\sum_{i=1}^{s} \binom{t+d_i-k}{t}$. The length of the real symmetric matrix W is $\binom{n+d}{n}$ and there are $\binom{n+2d}{n}$ equality constraints in (9) for $n = n(p)$. It can be seen that there is a trade off between choosing the number of variables and the degrees of polynomials.

Example 2.1 Consider two polynomials

$$f_1(z) = z(z+1)^2, \;\; f_2(z) = (z-1)(z+1)^2 + 1/10$$

and $k = 2, F = \mathbb{R}$. Solving the SOS program (2) and its dual problem with

$$\begin{aligned} f(X) = \|f_1 - p\cdot u_1\|_2^2 + \|f_2 - p \cdot u_2\|_2^2 &= p_1^2 u_{1,1}^2 + (1 - p_1 u_{1,2} - p_2 u_{1,1})^2 \\ &+ (2 - p_2 u_{1,2} - u_{1,1})^2 + (1 - u_{1,2})^2 + (-9/10 - p_1 u_{2,1})^2 \\ &+ (-1 - p_1 u_{2,2} - p_2 u_{2,1})^2 + (1 - u_{2,1} - p_2 u_{2,2})^2 + (1 - u_{2,2})^2, \end{aligned}$$

we get the minimal value $r^*_{sos} \approx 9.3876e-4$. The length of the matrix W in the corresponding SDP problem (2) is 28. From the optimal dual solutions, we find that the global minimum is achieved and the minimizer can be extracted:

$$X^* \approx (0.9335, 1.9778, 0.02569, 1.0013, -0.9739, 0.9975).$$

It corresponds to the monic approximate GCD

$$p(z) \approx 0.9335 + 1.9778z + z^2$$

with cofactors $u_1(z) \approx 0.02569 + 1.0013z, u_2(z) \approx -0.9739 + 0.9975z$.

Solving the SOS program (9) and its dual problem with

$$\begin{aligned} f(X) &= 12/5p_2p_1 + 7p_1^4 + 281/100p_2^4 - 281/50p_2^2p_1 + 11/5p_1^2p_2^2 \\ &- 6p_2^3 + 9/5p_1^3 + 981/100p_1^2 + 581/100p_2^2 + 281/100 \\ &- 6p_2p_1^2 - 9/5p_2^3p_1 + 9/5p_1 - 2p_2 - 2p_2p_1^3, \\ g(X) &= p_1^4 + p_1^2p_2^2 + 2p_1^2 + p_2^4 + p_2^2 + 1 - 2p_2^2p_1, \end{aligned}$$

we get the minimal value $r^*_{sos} \approx 9.3876e-4$. The length of the matrix W in the corresponding SDP problem (9) is 6. From the optimal dual solutions, we can extract the minimizer $X^* \approx (0.9335, 1.9778)$. Evaluating the rational function at X^* shows that

$$\frac{f(X^*)}{g(X^*)} \approx 9.3876e-4 \approx r^*_{sos},$$

which implies that X^* is the global minimizer. It corresponds to the same monic approximate GCD $p(z)$.

Example 2.2 Consider two polynomials

$$f_1(z_1, z_2) = z_1^2 + 2z_1z_2 + z_2^2 - 1, \;\; f_2(z_1, z_2) = z_1^2 + z_1z_2 - z_2 - 1.01$$

and $k=1, F=\mathbb{R}$. Solving the SOS program (2) and its dual problem with

$$
\begin{aligned}
f(X) = \|f_1 - p\cdot u_1\|_2^2 + \|f_2 - p\cdot u_2\|_2^2 &= (-1-p_1u_{1,1})^2 + (-p_1u_{1,3}-p_3u_{1,1})^2 \\
&+ (-p_1u_{1,2}-p_2u_{1,1})^2 + (2-p_2u_{1,3}-p_3u_{1,2})^2 + (1-p_3u_{1,3})^2 \\
&+ (1-p_2u_{1,2})^2 + (-1.01-p_1u_{2,1})^2 + (-p_1u_{2,3}-p_3u_{2,1})^2 + p_2^2u_{2,2}^2 \\
&+ (-1-p_1u_{2,2}-p_2u_{2,1})^2 + (1-p_2u_{2,3}-p_3u_{2,2})^2 + (1-p_3u_{2,3})^2,
\end{aligned}
$$

we get the minimal value $r^*_{sos} \approx 3.89306e-5$. The length of the matrix W in the corresponding SDP problem (2) is 55.

Solving the SOS program (9) and its dual problem with

$$
\begin{aligned}
f(X) = {} & -20.02p_2p_3^3p_1^2 + 26.0804p_1^2p_3^2p_2^2 - 22.04p_3p_2p_1^4 - 22.02p_3p_2^3p_1^2 \\
& + 5.98p_1^4p_2^2 + 9p_1^4p_3^2 + 6.0001p_1^2p_2^4 + 10.0201p_1^2p_3^4 + 13.0402p_2^4p_3^2 \\
& + 14.0402p_2^2p_3^4 + 8p_1^6 + 4.0201p_2^6 + 4.0201p_3^6 - 10p_3^3p_2^3 - 6p_2p_3^5 \\
& - 4p_3p_2^5 - 6.04p_3p_1^3p_2^2 - 2p_2p_3^4p_1 - 6.02p_1p_3p_2^4 - 4.02p_1p_3^3p_2^2 \\
& - 2.02p_1^3p_3^3 - 2.02p_1p_3^5 + 2p_2p_1^5 + 2p_2^3p_1^3 + 2p_1p_2^5, \\
g(X) = {} & p_1^6 + 2p_1^4p_2^2 + 2p_1^4p_3^2 + 2p_1^2p_2^4 + 5p_1^2p_3^2p_2^2 + 2p_1^2p_3^4 + p_2^6 + 2p_2^4p_3^2 \\
& + 2p_2^2p_3^4 + p_3^6,
\end{aligned}
$$

we get the minimal value

$$r^*_{sos} \approx 3.89306e-5.$$

Here $f(X), g(X)$ are homogeneous polynomials in the coefficients of p. The length of the matrix W in the corresponding SDP problem (9) is 10. From the optimal dual solutions, we get an approximate GCD

$$p(z) \approx 1.00199 + 0.99937z_2 + z_1.$$

Example 2.3 Consider two polynomials

$$
\begin{aligned}
f_1(z) &= (z-0.3)(z+4.6)(z-1.45)(z+10), \\
f_2(z) &= (z-0.301)(z+4.592)(z-1.458)(z-0.6)(z-15)(z+2)
\end{aligned}
$$

and $k=3, F=\mathbb{R}$. Solving the SOS program (2) and its dual problem we get $r^*_{sos} \approx 0.0156$. The length the of matrix W in the corresponding SDP problem (2) is 55. Solving the SOS program (9) and its dual problem we get the minimal value $r^*_{sos} \approx 0.0156$. The length of the matrix W in the corresponding SDP problem (9) is 84.

Example 2.4 (Kaltofen et al., 2006a) Consider two polynomials

$$f_1(z) = 1000z^{10} + z^3 - 1,\ \ f_2(z) = z^2 - \frac{1}{100}$$

and $k = 1, F = \mathbb{R}$. Solving the SOS program (9) and its dual problem we get $r^*_{sos} \approx 0.042157904$. The length of matrix W in the corresponding SDP problem (9) is 13. It was shown in (Kaltofen et al., 2006a) that after about ten iterations in the average, the STLN algorithm converges to the following local minima:

$$0.0421579, 0.0463113, 0.0474087, 0.0493292, \ldots$$

for different initializations.

Example 2.5 (Kaltofen et al., 2006b) Consider two polynomials

$$f_1(z) = z^2 + 2z + 1,\ f_2(z) = z^2 - 2z + 2$$

and $k = 1, F = \mathbb{R}$. Let $p(z) = p_1 + z, u_1(z) = u_{1,1} + u_{1,2}z, u_2(z) = u_{2,1} + u_{2,2}z$. Solving the SOS program (2) and its dual problem we get $r^*_{sos} \approx 2.000569$ and an approximate GCD

$$p(z) \approx z - 14686.677911.$$

Solving the SOS program (9) and its dual problem with

$$\frac{f(X)}{g(X)} = \frac{(p_1^2 - 2p_1 + 1)^2 + (p_1^2 + 2p_1 + 2)^2}{1 + p_1^2 + p_1^4} = \frac{12p_1^2 + 4p_1 + 3}{1 + p_1^2 + p_1^4} + 2,$$

we get the minimal value $r^*_{sos} \approx 2.000000$, but extract no minimizers. The global minimum $r^* = 2$ is only an infimum, $f(X) - r^* g(X) = 12(p_1 + 1/6)^2 + 8/3$ is an SOS, and there are no global minimizers.

2.3 Minimization problem with constraints

As in (Kaltofen et al., 2006a,b), the problem of computing approximate GCDs of several polynomials can also be formulated as

$$\left.\begin{array}{ll} \min\limits_{\Delta c} & \|\Delta c\|_2^2 \\ s.t. & S_k(c + \Delta c)x = 0,\ \exists\ x \neq 0, \end{array}\right\} \tag{10}$$

where c is the coefficient vector of $f_1, \ldots, f_s$, the perturbations to the polynomials are parameterized via the vector Δc, and $S_k(c + \Delta c)$ is the multi-polynomial generalized Sylvester matrix (Kaltofen et al., 2006a). The minimization problem (10) is a quadratic optimization problem with quadratic constraints.

Similar to the method used in (Kaltofen et al., 2006a,b), we can choose a column of S_k and reformulate the problem as

$$\left.\begin{array}{ll}\min\limits_{\Delta c, x} & \|\Delta c\|_2^2 + \rho\|x\|_2^2 \\ s.t. & A(c+\Delta c)x = b(c+\Delta c).\end{array}\right\} \tag{11}$$

Two alternative formulations are

$$\left.\begin{array}{ll}\min\limits_{\Delta c, x} & \|\Delta c\|_2^2 \\ s.t. & S_k(c+\Delta c)x = 0, \\ & \|x\|_2^2 = 1,\end{array}\right\} \tag{12}$$

and

$$\left.\begin{array}{ll}\min\limits_{\Delta c, x} & \|\Delta c\|_2^2 + \rho\|x\|_2^2 \\ s.t. & S_k(c+\Delta c)x = 0, \\ & v^T x = 1,\end{array}\right\} \tag{13}$$

where ρ is a small positive number and v is a random vector. The dimensions of the vectors $\Delta c, x$ are $\sum_{i=1}^s \binom{t+d_i}{t}$ and $\sum_{i=1}^s \binom{t+d_i-k}{t}$ respectively.

Let us describe the polynomial minimization problem with constraints as:

$$\left.\begin{array}{ll}\min\limits_{X\in\mathbb{R}^n} & \sum\limits_{\alpha} f_\alpha X^\alpha \\ s.t. & h_1(X) \geq 0, \ldots, h_l(X) \geq 0.\end{array}\right\} \tag{14}$$

We can reformulate it as a convex LMI (Linear Matrix Inequality) optimization problem (or semidefinite program):

$$\left.\begin{array}{ll}\inf\limits_{y} & \sum\limits_{\alpha} f_\alpha y_\alpha \\ s.t. & y_{0,\ldots,0} = 1, \\ & M_d(y) \succeq 0, \\ & M_{d-w_i}(h_i y) \succeq 0, \ 1 \leq i \leq l,\end{array}\right\} \tag{15}$$

where $w_i := \lceil \frac{\mathrm{tdeg}(h_i)}{2} \rceil$ for $1 \leq i \leq l$, $d \geq \max(\lceil \frac{\mathrm{tdeg}(f(X))}{2} \rceil, w_1, \ldots, w_l)$, the moment matrix $M_d(y)$ and localizing matrices $M_{d-w_i}(h_i y)$ of real vector y are defined in (Lasserre, 2001).

Example 2.6 Consider two polynomials

$$f_1(z) = z^3 - 1, \ f_2(z) = z^2 - 1.01$$

and $k = 1, F = \mathbb{R}, \rho = 10^{-6}$. We choose the first column of S_1 to be b and the remaining columns to be matrix A. For minimization problem (11), the minimal perturbation computed by the first-order ($d = 1$) semidefinite programs is $9.9673e{-6}$. The length of the matrix involved in the corresponding

SDP is 152. The minimal perturbation computed by the second-order ($d = 2$) semidefinite program is $2.0871e - 5$. The length of the matrix involved in the corresponding SDP is 6476. The minimizer can be extracted by the second-order semidefinite program.

For minimization problem (12), the lower bounds given by the first and second order semidefinite programs are 0 and $2.0852e - 5$ respectively. Here we notice that one feasible solution corresponding to the first-order relaxation in the homogenous model (12) is $\Delta c = 0,\ x = [0, 1]^T$ with objective value zero.

As pointed out by Erich Kaltofen, if we want to compute the lower bound for the minimization problem (10) by solving problem (11), we have to try all the possible selection of b, which is very time consuming. So we suggest the formulation (13). For minimization problem (13), the lower bounds given by the first and second order semidefinite programs depend on the choice of random vector v. The obtained lower bounds are around 10^{-6} and 10^{-5} respectively.

The experiments show that the first-order semidefinite programs give us some useful information on the minimal perturbations. Although we may compute the global minimizer from high-order semidefinite programs, the sizes of the matrices increase quickly.

3 Exploiting sparsity in SOS relaxation

In this section, we investigate how to reduce the size of the SOS program by exploiting the special structures of the minimization problems involved in the approximate GCD computation. Examples 2.1 and 2.2 show that the SOS relaxations are dense for the rational function formulation. So in the following, we only exploit the sparsity in the polynomial formulation SOS program (2). The same technique can be applied to solve Problem (10).

3.1 Exploiting Newton polytope

There are algorithms in (Parrilo, 2000; Kojima et al., 2005; Waki et al., 2006) that remove redundant monomials by exploiting sparsity in the SOS programs. However, it is quite expensive to compute the structured sparsity for problems having large size, whereas the sparsity structure of the approximate GCD problem (1) is obvious and can be analyzed easily.

Given a polynomial $p(x) = \sum_\alpha p_\alpha x^\alpha$, the cage of p, $\mathrm{C}(p)$, is the convex hull of

$\text{supp}(p) = \{\alpha | p_\alpha \neq 0\}$. Denote the convex hull of the degrees α by $\mathrm{H}(\cdot)$.

Theorem 3.1 *(Reznick, 1978) For any polynomial p, $\mathrm{C}(p^2) = 2\mathrm{C}(p)$; for any positive semidefinite (PSD) polynomials f and g, $\mathrm{C}(f) \subseteq \mathrm{C}(f+g)$; if $f = \sum_j g_j^2$ then $\mathrm{C}(g_j) \subseteq \frac{1}{2}\mathrm{C}(f)$.*

Corollary 3.2 *For any PSD polynomials f and g, $\mathrm{C}(f+g) = \mathrm{H}(\text{supp}(f) \cup \text{supp}(g))$; if $f = \sum_j g_j^2$ then $\mathrm{C}(f) = 2\mathrm{H}(\bigcup_j \text{supp}(g_j))$.*

Proof. Since f and g are PSD polynomials, according to Theorem 3.1, we have $\text{supp}(f) \cup \text{supp}(g) \subseteq \mathrm{C}(f) \cup \mathrm{C}(g) \subseteq \mathrm{C}(f+g)$. From

$$\mathrm{C}(f+g) = \mathrm{H}(\text{supp}(f+g)) \subseteq \mathrm{H}(\text{supp}(f) \cup \text{supp}(g)) \subseteq \mathrm{C}(f+g),$$

it is clear that $\mathrm{C}(f+g) = \mathrm{H}(\text{supp}(f) \cup \text{supp}(g))$.

If $f = \sum_j g_j^2$, we have $\mathrm{C}(g_j^2) = 2\mathrm{C}(g_j)$ according to Theorem 3.1. Because $\mathrm{C}(g_j) = \mathrm{H}(\text{supp}(g_j)) \subseteq \mathrm{H}(\bigcup_j \text{supp}(g_j))$, so $\mathrm{H}(\bigcup_j \mathrm{C}(g_j)) = \mathrm{H}(\bigcup_j \text{supp}(g_j))$. Then

$$\mathrm{C}(f) = \mathrm{H}(\bigcup_j \text{supp}(g_j^2)) = \mathrm{H}(\bigcup_j \mathrm{C}(g_j^2)) = 2\mathrm{H}(\bigcup_j \mathrm{C}(g_j)) = 2\mathrm{H}(\bigcup_j \text{supp}(g_j)).$$

□

The SOS program (2) is to compute polynomials $v_j(X)$ such that

$$f(X) - r = m_d(X)^T W m_d(X) = \sum_j v_j(X)^2.$$

Let X^σ be any monomial in v_j. By Theorem 3.1 and Corollary 3.2, we have

$$\sigma \in \frac{1}{2}\mathrm{C}(f(X) - r) = \mathrm{H}(O \cup (\bigcup_{i=1, |\alpha| \leq d_i}^{s} \text{supp}(f_{i,\alpha} - \sum_{\beta+\gamma=\alpha} p_\beta u_{i,\gamma}))),$$

where O is the origin. If there exists nonzero constant term in the coefficients p_β, the monomials existing in $f_{i,\alpha} - \sum_{\beta+\gamma=\alpha} p_\beta u_{i,\gamma}$, $1 \leq i \leq s$, $|\alpha| \leq d_i$ are

$$1, p_1 u_{1,1}, \ldots, p_1 u_{s,n(u_s)}, \ldots, p_{n(p)} u_{1,1}, \ldots, p_{n(p)} u_{s,n(u_s)}, u_{1,1}, \ldots, u_{s,n(u_s)}, \quad (16)$$

let $p_{n(p)+1} = 1$ and $n_1(p) = n(p) + 1$. Otherwise all existing monomials are

$$1, p_1 u_{1,1}, \ldots, p_1 u_{s,n(u_s)}, \ldots, p_{n(p)} u_{1,1}, \ldots, p_{n(p)} u_{s,n(u_s)}, \quad (17)$$

let $n_1(p) = n(p)$. According to the property of convex hull, there exist $\lambda_{j,i,k} \geq 0$ for $1 \leq j \leq n_1(p), 1 \leq i \leq s, 1 \leq k \leq n(u_i)$ such that $\sum_{i,j,k} \lambda_{j,i,k} \leq 1$ and

$$X^\sigma = \prod_{i,j,k} (p_j u_{i,k})^{\lambda_{j,i,k}} = \prod_j p_j^{e_j} \prod_{i,k} u_{i,k}^{e_{i,k}}.$$

Because the exponents $e_j, e_{i,k}$ are nonnegative integers and $\sum_j e_j = \sum_{i,k} e_{i,k} = \sum_{i,j,k} \lambda_{j,i,k} \leq 1$, so the monomial X^σ can only be 1 or $p_j u_{i,k}$ for some j, i, k and only these monomials are needed in the SOS program (2). The sparse SOS program of the approximate GCD problem (1) is:

$$\left.\begin{aligned} r^*_{sos1} := \sup_{r\in\mathbb{R},W} \quad & r \\ s.t. \quad & f(X) - r = m_{\mathcal{G}}(X)^T W m_{\mathcal{G}}(X), \\ & W \succeq 0, \end{aligned}\right\} \tag{18}$$

where $m_{\mathcal{G}}(X) = [1, p_1u_{1,1}, \ldots, p_1u_{s,n(u_s)}, \ldots, p_{n_1(p)}u_{1,1}, \ldots, p_{n_1(p)}u_{s,n(u_s)}]^T$. Let $n(u) = \sum_{i=1}^{s} n(u_i)$, then the length of the real symmetric matrix W is $1 + n_1(p)n(u)$ and there are $1 + n_1(p)n(u) + \binom{n_1(p)+1}{2}\binom{n(u)+1}{2}$ equality constraints.

3.2 Extract solutions in sparse case

The dual SDP problem of the sparse SOS program (18) is:

$$\left.\begin{aligned} \inf_{y} \quad & \sum_{\alpha} f_\alpha y_\alpha \\ s.t. \quad & y_{0,\ldots,0} = 1, \\ & M_{\mathcal{G}}(y) \succeq 0, \end{aligned}\right\} \tag{19}$$

where the moment matrix $M_{\mathcal{G}}(y) := (y_{\alpha+\beta})_{\alpha,\beta\in\mathcal{G}}$ and its rows and columns correspond to the monomial vector $m_{\mathcal{G}}(X)$.

Suppose the moment matrix evaluated at the optimal solution y^* is written as $M_{\mathcal{G}}(y^*) = \begin{bmatrix} 1 & \gamma_1^T \\ \gamma_1 & M_1 \end{bmatrix}$, and we have the Cholesky factorization $M_1 = VV^T$. For any vector γ satisfying $V^T\gamma = 0$, we have $\gamma^T[-\gamma_1, I]M_{\mathcal{G}}(y^*)[-\gamma_1, I]^T\gamma = -(\gamma_1^T\gamma)^2 \geq 0$, hence $\gamma_1^T\gamma = 0$. So there exists a vector γ_2 such that $\gamma_1 = V\gamma_2$, and

$$M_{\mathcal{G}}(y^*) = \begin{bmatrix} \gamma_2^T \\ V \end{bmatrix} \begin{bmatrix} \gamma_2 & V^T \end{bmatrix} + \begin{bmatrix} 1 - \gamma_2^T\gamma_2 & 0 \\ 0 & 0 \end{bmatrix}.$$

It can be seen that $1-\gamma_2^T\gamma_2 = \gamma^T M_{\mathcal{G}}(y^*)\gamma \geq 0$ with $\gamma = [1, -\gamma_2^T(V^TV)^{-1}V^T]^T$. We denote $c = \text{rank}(M_1)$.

If $\text{rank}(M_{\mathcal{G}}(y^*)) > c$, then $1 - \gamma_2^T\gamma_2 > 0$ and $y = [1, 0, \ldots, 0]^T$ is an optimal solution of the sparse SDP problem (19). The global minimum is achieved and $X^* = 0$ is a global minimizer. If $\text{rank}(M_{\mathcal{G}}(y^*)) = c$, then $\gamma_2^T\gamma_2 = 1$. It corresponds to the general case for the approximate GCD problem (1).

If $c = 1$, we write V as vector $[v_{1,1}, \ldots, v_{1,n(u)}, \ldots, v_{n_1(p),1}, \ldots, v_{n_1(p),n(u)}]^T$ and rearrange its elements to define a matrix $B = (v_{i,j})_{n_1(p)\times n(u)}$. From the structure of the moment matrix and $M_1 = VV^T$, we have $v_{i_1,j_1}v_{i_2,j_2} = v_{i_1,j_2}v_{i_2,j_1}$ for any $1 \le i_1, i_2 \le n_1(p), 1 \le j_1, j_2 \le n(u)$. Therefore any two columns of the matrix B are linearly dependent. Hence $\mathrm{rank}(B) = 1$, the global minimum is achieved, and one global minimizer can be extracted by decomposing $B/\gamma_2 = [p_1, \ldots, p_{n_1(p)}]^T[u_{1,1}, \ldots, u_{s,n(u_s)}]$.

If $c > 1$, we do not have sufficient conditions for global optimality like the flat extension conditions in (5) for the sparse SDP problem (19). However, we can assume that the global optimality is achieved (see Remark 4.2) and try to extract some solutions $X^*(j)$ by a method similar to the one in (Henrion and Lasserre, 2005). Therefore we apply Gauss-Newton iterations to refine the solutions (see Remark 4.1). If $f(X^*(j))$ is approximately equal to r^*_{sos1}, we know that the global minimum is achieved approximately since $f(X^*(j)) \ge r^* \ge r^*_{sos1}$.

The extracting method is described below. Suppose

$$[\gamma_2, V^T]^T[\gamma_2, V^T] = V^*(V^*)^T,\ V^* = [\eta_1 m_{\mathcal{G}}(X^*(1)), \ldots, \eta_c m_{\mathcal{G}}(X^*(c))].$$

We write V as $[V_1, \ldots, V_{n_1(p)}]^T$ and $V_1, \ldots, V_{n_1(p)}$ are $c \times n(u)$ matrices. If $n_1(p) = n(p) + 1$, then set $V_0 = V_{n_1(p)}$. Otherwise, we have $n_1(p) = n(p)$, let $V_0 = \sum_{i=1}^{n(p)} \theta_i V_i$ be a random combination and assume $\sum_{i=1}^{n(p)} \theta_i p_i = 1$ for all solutions. Consider the matrix $\widetilde{V} = [V_0, V_1, \ldots, V_{n(p)}, \gamma_2]^T$ and its corresponding monomial vector $m(X) = [u_{1,1}, \ldots, u_{s,n(u_s)}, p_1u_{1,1}, \ldots, p_1u_{s,n(u_s)}, \ldots, 1]^T$.

If $\mathrm{rank}(V_0) = c$, we reduce $\widetilde{V}$ to column echelon form U by Gaussian elimination with column pivoting, and suppose all the pivot elements in U (i.e. the first non-zero elements in each column) correspond to monomial basis $w = [u_{i_1,k_1}, \ldots, u_{i_c,k_c}]^T$. It holds $m(X) = Uw$ for all solutions, so we can extract from U the multiplication matrix N_i and the vector γ_3 such that $p_i w = N_i w$ and $1 = \gamma_3^T w$ for $i = 1, \ldots, n(p)$.

As in (Henrion and Lasserre, 2005), in order to compute common eigenvalues $p_i(j), j = 1, \ldots, c$, we build a random combination $N = \sum_{i=1}^{n(p)} \lambda_i N_i$ and compute the ordered Schur decomposition $N = QTQ^T$ (Corless et al., 1997), where $Q = [q_1, \ldots, q_c]$ is an orthogonal matrix and T is an upper-triangular matrix. For $j = 1, \ldots, c$, the j-th solution is given by $p_i(j) = q_j^T N_i q_j$, and we obtain $u_{1,1}(j), \ldots, u_{s,n(u_s)}(j)$ by solving $Nw = T_{j,j}w, \gamma_3^T w = 1$ and $m(X) = Uw$. It should be noticed that the cofactors can also be computed by (7).

Example 3.1 Consider two polynomials

$$f_1(z) = z^3 - z,\ f_2(z) = 3z^2 - 1$$

and $k = 1, F = \mathbb{R}$. Solving the SOS program (18) and its dual problem (19) with

$$m_{\mathcal{G}}(X) = [1, p_1u_{1,1}, p_1u_{1,2}, p_1u_{1,3}, p_1u_{2,1}, p_1u_{2,2}, u_{1,1}, u_{1,2}, u_{1,3}, u_{2,1}, u_{2,2}]^T,$$

we get the minimal value $r^*_{sos1} \approx 0.0991769059$ and $\text{rank}(M_{\mathcal{G}}(y^*)) = \text{rank}(M_1) = 2$. We compute γ_2 and $V = [V_1, V_2]^T$ via SVD. Since $\text{rank}(V_2) = 2$, we can reduce $\widetilde{V} = [V_2, V_1, \gamma_2]^T$ to U and solve the common eigenvalue problem to get the values of p_1 corresponding to two approximate GCDs $z - 0.5878795$ and $z + 0.5878795$ with cofactors

$$\begin{aligned} u_1 &\approx 1.0518891z^2 + 0.7066490z - 0.4344338,\ u_2 \approx 2.9913164z + 1.7437641,\\ u_1 &\approx 1.0518891z^2 - 0.7066490z - 0.4344338,\ u_2 \approx 2.9913164z - 1.7437641, \end{aligned}$$

respectively. Applying the Gauss-Newton iterations to refine the results, we get

$$f(X^*(1)) \approx 0.0991769059,\ f(X^*(2)) \approx 0.0991769059.$$

The global minimum is achieved since $f(X^*(1)) = f(X^*(2)) \approx r^*_{sos1}$.

3.3 Exploiting correlative sparsity

Since the polynomial $f(X)$ in SOS program (2) is written as

$$f(X) = \sum_{i=1}^{s} \|f_i - p \cdot u_i\|_2^2,$$

we can define the subsets

$$X_{\Delta_i} = \{p_1, \ldots, p_{n(p)}\} \cup \{u_{i,1}, \ldots, u_{i,n(u_i)}\}.$$

The collections of variables $X_{\Delta_1}, \ldots, X_{\Delta_s}$ satisfy the following running intersection property: for every $k = 1, \ldots, s-1$,

$$X_{\Delta_{k+1}} \cap \bigcup_{j=1}^{k} X_{\Delta_j} \subseteq X_{\Delta_i} \quad \text{for some} \quad 1 \le i \le k.$$

According to (Waki et al., 2006; Lasserre, 2006; Nie and Demmel, 2007), we are going to find the maximum r such that

$$\left.\begin{aligned} r^*_{sos2} := \sup_{r\in\mathbb{R}, W_1,\ldots,W_s} \quad & r \\ s.t. \quad & f(X) - r = \sum_{i=1}^{s} m_d(X_{\Delta_i})^T W_i m_d(X_{\Delta_i}), \\ & W_i \succeq 0,\ 1 \le i \le s, \end{aligned}\right\} \tag{20}$$

where $m_d(X_{\Delta_i})$ is the column vector of all monomials up to degree $d = 2$. The length of W_i is $\binom{n(p)+n(u_i)+2}{2}$.

The following sparse SOS program is obtained by considering both the Newton polytope and correlative sparsity:

$$\left.\begin{array}{rll} r^*_{sos3} := & \sup\limits_{r\in\mathbb{R},W_1,\ldots,W_s} & r \\ & s.t. & f(X) - r = \sum\limits_{i=1}^{s} m_{\mathcal{G}_i}(X)^T W_i m_{\mathcal{G}_i}(X), \\ & & W_i \succeq 0,\ 1 \le i \le s, \end{array}\right\} \tag{21}$$

where $m_{\mathcal{G}_i}(X) = [1, p_1u_{i,1}, \ldots, p_1u_{i,n(u_i)}, \ldots, p_{n_1(p)}u_{i,1}, \ldots, p_{n_1(p)}u_{i,n(u_i)}]^T$ and the length of W_i is $1 + n_1(p)n(u_i)$.

3.4 Comparison of sparsity strategies

The relation between the optimums of polynomial minimization problem (1), the SOS program (2) and the three sparse SOS programs (18),(20),(21) is

$$r^* \ge r^*_{sos} = r^*_{sos1} \ge r^*_{sos2} \ge r^*_{sos3}.$$

The sizes of the SDP matrices in the three kinds of sparse SOS programs are:

$$m_1 = (1 + n_1(p)n(u))^2,\ m_2 = \sum_{i=1}^{s} \binom{n(p)+n(u_i)+2}{2}^2,\ m_3 = \sum_{i=1}^{s}(1 + n_1(p)n(u_i))^2.$$

We have that

$$s \cdot m_2 \ge s \cdot m_3 = s\sum_{i=1}^{s}(1 + n_1(p)n(u_i))^2 \ge (s + n_1(p)n(u))^2 \ge m_1 \ge m_3.$$

We show in Table 1 experiments of applying four kinds of SOS relaxations (2),(18),(20),(21) to compute an approximate GCD of three pairs of polynomials f_1 and f_2. We notice that the first and third kinds of sparse SOS programs can reduce the size of the optimization problem remarkably. However, the third kind of sparse SOS program can only give a lower bound in general.

4 Implementation and experiments

The methods described above have been implemented by the first author in Matlab based on algorithms in SOSTOOLS (Prajna et al., 2004), YALMIP

Table 1
Comparison of different sparsity SOS programs.

f_1	$z(z+1)^2$	$(z+1)^2$	$z_1^2+2z_1z_2+z_2^2-1$
f_2	$(z-1)(z+1)^2+0.1$	$z^2+2z+1.01$	$z_1^2+z_1z_2-z_2-1.01$
k	2	1	1
r^*_{sos}	9.3876e-4	3.3167e-5	3.8931e-5
size	785	442	3026
r^*_{sos1}	9.3876e-4	3.3167e-5	3.8931e-5
size	170	82	362
r^*_{sos2}	4.2616e-7	3.3167e-5	3.6525e-6
size	451	201	1569
r^*_{sos3}	1.2123e-10	3.3167e-5	3.6502e-6
size	99	51	201

Table 2
Experimental results of examples in (Chin et al., 1998).

d_i	k	polynomial	rational	poly. sparse	Newton	STLN
5,4	2	1.620473e-8	1.579375e-8	1.560388e-8	1.560294e-8	1.560294e-8
4,6	3	1.561803e-2	1.561770e-2	1.561754e-2	1.561754e-2	1.561754e-2
3,3	2	1.702596e-2	1.702596e-2	1.702596e-2	1.702596e-2	1.702596e-2
5,5	4	7.086761e-5	7.086331e-5	7.086312e-5	7.086311e-5	7.086311e-5
3,2,3	2	1.729192e-5	1.729175e-5	1.729175e-5	1.729175e-5	1.729175e-5

(Löfberg, 2004) and SeDuMi (Sturm, 1999). We apply Gauss-Newton iterations to improve the accuracy of the results computed by SDP solvers.

In Table 2, we compare the minimal residues achieved by different methods, for examples in (Chin et al., 1998). The third through the fifth columns are the minimum residues computed by SOS programs (2),(9),(18) respectively. The sixth column consists of the minimum residues refined by applying Gauss-Newton iteration. The last column consists of the minimal residues computed by STLN method in (Kaltofen et al., 2006a).

Remark 4.1 In our experiments, the fixed precision SDP solvers in Matlab often encounter numerical problems and the accuracy of the computed results is not enough. Sometimes the lower bounds r^*_{sos} are even larger than the local minima computed by STLN method (see the Tables). So we need to apply Gauss-Newton iterations to refine the global minimizer.

Table 3
Experimental results of random examples.

n	d_i	k	rational or poly. sparse	Newton	STLN
1	65,65	1	7.85073312e-6	7.84895293e-6	7.84895293e-6
	20,20	1	8.77479920e-6	8.72559736e-6	8.72559736e-6
	19,19	2	6.90363904e-5	6.88055540e-5	6.88055540e-5
	15,16	3	5.40141468e-5	5.40041186e-5	5.40041186e-5
	15,15	4	2.03530157e-4	2.03408558e-4	2.03408558e-4
	14,14	5	1.61530926e-4	1.61402027e-4	1.61402027e-4
	13,14	6	3.20576318e-4	3.20568077e-4	3.20568077e-4
	13,14	7	1.01396122e-4	1.01208845e-4	1.01208845e-4
	13,13	8	2.71981668e-4	2.71825700e-4	2.71825699e-4
	10,10	5	1.08249380e-4	1.08207747e-4	1.08207747e-4
	6,4	3	1.13839852e-2	1.13839851e-2	1.13839851e-2
	3,3	2	1.32223417e-5	1.32185834e-5	1.32185834e-5
	3,2	2	5.07694667e-3	5.07694668e-3	5.07694668e-3
	2,2	1	1.04706956e-4	1.04706621e-4	1.04706621e-4
2	5,5	1	4.28360491e-4	4.28336831e-4	4.28336831e-4
	4,5	2	7.44701587e-4	7.44633647e-4	7.44633647e-4
	4,5	3	4.94579319e-6	4.94240926e-6	4.94240926e-6
	5,5	4	1.20576057e-5	1.20467080e-5	1.20467080e-5
	3,3	2	4.09496306e-6	4.09320783e-6	4.09320783e-6
	3,2	2	3.51277009e-4	3.51276829e-4	3.51276829e-4

In Table 3, we show the experimental results of random examples generated in the same way described in (Kaltofen et al., 2006a). The first example is solved by the SOS program (9) and the other examples are solved by the SOS program (18). For these random examples, the ranks of all moment matrices are one and the global minimum is achieved.

Remark 4.2 In the last two tables, the minimum residues computed by STLN method (Kaltofen et al., 2006a) are approximately equal to the minima computed by solving SDP and Gauss-Newton iteration. However, as shown in Example 2.4, for different initializations, the STLN method may not converge to global minima, while the results computed by solving SDP are guaranteed to be global minima since the ranks of the moments are all equal to one in these examples.

The results computed by solving SDP are lower bounds. In (Kaltofen et al., 2008), an efficient algorithm is given to certify the exact lower bounds via rationalizing SOS obtained from our SDPs for the approximate GCD problem and other problems.

5 Conclusions

In this paper, we discussed how to solve approximate GCD problem which can be formulated as an unconstrained quartic polynomial optimization problem about the coefficients of factor polynomials. This is a nonconvex nonlinear least squares problem and it is usually very difficult for finding global solutions. This paper proposed various semidefinite relaxation methods for solving this special polynomial optimization. The usual SOS relaxation is often very good for finding global solutions, but it is expensive to solve problems of large sizes. By exploiting the special sparsity structures of the quartic polynomial arising from the GCD approximations, we proposed various sparse SOS relaxations based on different formulations and sparsity techniques. Numerical experiments show the efficiency of these relaxation methods.

There is a trade-off in choosing these various sparse relaxation methods. The sparse SOS relaxation (18) is the best in quality (it has the same quality as the dense SOS relaxations), but it is the most expensive one in these relaxations. The sparse SOS relaxation (21) has the lowest quality, but it is the cheapest one and can solve problems of large sizes. In practice, to solve GCD problems of large sizes, we suggest applying relaxation (21) to find one approximate solution, and then applying local methods like STLN to refine the solution.

The GCD problem can also be equivalently formulated as an unconstrained rational function optimization (9). This formulation is faster than the polynomial SOS program (2) when there are only few variables and the degree of GCD is very small. However, the problem (9) is very difficult to solve when the GCD problem has large size. It is also an interesting work to exploit the special structures of (9) to obtain more efficient methods.

The strength of SOS relaxation methods is that they do not require an initial guess of solutions and can always return a lower bound of the global minimum. When this lower bound is achieved, we immediately know that the global solution is found. Our preliminary experiments show that these SOS relaxation methods work well in solving the GCD problems. They often return global solutions.

Our proposed sparse SOS relaxation methods are based on the nonlinear least squares formulation (1). Since the GCD problem can also be equivalently formulated as (10), it is also possible to exploit special structure of (10). An interesting future work is to get more efficient semidefinite relaxations for (11)-(13) based on their structures.

Acknowledgement: The authors want to thank Erich Kaltofen, Zhengfeng Yang and Kartik Sivaramakrishnan for a number of interesting discussions on SDP for approximate polynomial computation. We are also grateful to the IMA in Minneapolis for hosting us during our collaboration on this project.

References

Chin, P., Corless, R. M., Corliss, G. F., 1998. Optimization strategies for the approximate GCD problem. In: (Gloor, 1998), pp. 228–235.

Corless, R. M., Gianni, P. M., Trager, B. M., 1997. The reordered Schur factorization method for zero-dimensional polynomial systems with multiple roots. In: Küchlin, W. (Ed.), Proc. 1997 Internat. Symp. Symbolic Algebraic Comput. ISSAC'97. ACM Press, New York, N. Y., pp. 133–140.

Gloor, O. (Ed.), 1998. ISSAC 98 Proc. 1998 Internat. Symp. Symbolic Algebraic Comput. ACM Press, New York, N. Y.

Henrion, D., Lasserre, J. B., 2005. Detecting global optimality and extracting solutions in gloptipoly. In: D.Henrion, Garulli, A. (Eds.), Positive Polynomials in Control. Vol. 312 of Lecture Notes in Control and Information Sciences. Springer Verlag, Berlin, pp. 293–310.

Hitz, M. A., Kaltofen, E., 1998. Efficient algorithms for computing the nearest polynomial with constrained roots. In: (Gloor, 1998), pp. 236–243.

Hitz, M. A., Kaltofen, E., Lakshman Y. N., 1999. Efficient algorithms for computing the nearest polynomial with a real root and related problems. In: Dooley, S. (Ed.), Proc. 1999 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'99). ACM Press, New York, N. Y., pp. 205–212.

Jibetean, D., de Klerk, E., 2006. Global optimization of rational functions: a semidefinite programming approach. Mathematical Programming 106 (1), 93–109.

Kaltofen, E., Li, B., Yang, Z., Zhi, L., 2008. Exact certification of global optimality of approximate factorizations via rationalizing sums-of-squares with floating point scalars. In: Jeffrey, D. (Ed.), ISSAC 2008 Proc. 2008 Internat. Symp. Symbolic Algebraic Comput. ACM Press, New York, N. Y.

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

Kaltofen, E., Yang, Z., Zhi, L., Dec. 2006b. Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials. Manuscript, 21 pages.

Karmarkar, N., Lakshman Y. N., 1996. Approximate polynomial greatest common divisors and nearest singular polynomials. In: Lakshman Y. N. (Ed.), ISSAC 96 Proc. 1996 Internat. Symp. Symbolic Algebraic Comput. ACM Press, New York, N. Y., pp. 35–42.

Karmarkar, N. K., Lakshman Y. N., 1998. On approximate GCDs of univariate polynomials. J. Symbolic Comput. 26 (6), 653–666.

Kojima, M., Kim, S., Waki, H., 2005. Sparsity in sums of squares of polynomials. Mathematical Programming 103, 45–62.

Lasserre, J. B., 2001. Global optimization with polynomials and the problem of moments. SIAM Journal on Optimization 11 (3), 796–817.

Lasserre, J. B., 2006. Convergent SDP-relaxations in polynomial optimization with sparsity. SIAM Journal on Optimization 17 (3), 822–843.

Löfberg, J., 2004. YALMIP: A toolbox for modeling and optimization in MATLAB. In: Proceedings of the CACSD Conference. pp. 284–289.

Nesterov, Y., Nemirovskii, A., 1994. Interior-Point polynomial algorithms in convex programming. Vol. 13 of Studies in Applied Mathematics. SIAM, Philidelphia, PA.

Nie, J., Demmel, J., 2007. Sparse SOS relaxations for minimization functions that are summation of small polynomials. URL: `http://www.arxiv.org/abs/math.OC/0606476`.

Nie, J., Demmel, J., Gu, M., 2008. Global minimization of rational functions and the nearest GCDs. Journal of Global Optimization 40 (4), 697–718.

Parrilo, P. A., 2000. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. Ph.D. thesis, California Institute of Technology, Pasadena, CA, URL: `http://www.mit.edu/~parrilo/`.

Parrilo, P. A., Sturmfels, B., 2003. Minimizing polynomial functions. In: Algorithmic and Quantitative Real Algebraic Geometry. Vol. 60 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science. AMS, pp. 83–99.

Prajna, S., Papachristodoulou, A., Seiler, P., Parrilo, P. A., 2004. SOSTOOLS: Sum of squares optimization toolbox for MATLAB. URL: `http://www.mit.edu/~parrilo/sostools`.

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

Shor, N. Z., 1987. Class of global minimum bounds of polynomial functions. Cybernetics and Systems Analysis 23 (6), 731–734.

Shor, N. Z., Stetsyuk, P. I., 1997. The use of a modification of the r-algorithm for finding the global minimum of polynomial functions. Cybernetics and Systems Analysis 33, 482–497.

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

Vandenberghe, L., Boyd, S., 1996. Semidefinite programming. SIAM Review

38 (1), 49–95.
Waki, H., Kim, S., Kojima, M., Muramatsu, M., 2006. Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity. SIAM Journal on Optimization 17 (1), 218–242.
Wolkowicz, H., Saigal, R., Vandenberghe, L. E., 2000. Handbook of Semidefinite Programming: Theory, Algorithms, and Applications. Kluwer Academic, Boston.
Zhi, L., Noda, M.-T., Kai, H., Wu, W., Jun. 2004. Hybrid method for computing the nearest singular polynomials. Japan J. Industrial and Applied Math. 21 (2), 149–162.
Zhi, L., Wu, W., 1998. Nearest singular polynomial. J. Symbolic Comput. 26 (6), 667–675, special issue on Symbolic Numeric Algebra for Polynomials S. M. Watt and H. J. Stetter, editors.