Advances in Linear Algebra & Matrix Theory, **2013, 3, 39-46**
Published Online December 2013 (http://www.scirp.org/journal/alamt)
http://dx.doi.org/10.4236/alamt.2013.34008

Computing Approximation GCD of Several Polynomials by Structured Total Least Norm*

Xuefeng Duan[1], Xinjun Zhang[1], Qingwen Wang[2]
[1]College of Mathematics and Computational Science, Guilin University of Electronic Technology, Guilin, China
[2]Department of Mathematics, Shanghai University, Shanghai, China
Email: zhangxinjunguilin@163.com

Received September 28, 2013; revised October 30, 2013; accepted November 8, 2013

ABSTRACT

The task of determining the greatest common divisors (GCD) for several polynomials which arises in image compression, computer algebra and speech encoding can be formulated as a low rank approximation problem with Sylvester matrix. This paper demonstrates a method based on structured total least norm (STLN) algorithm for matrices with Sylvester structure. We demonstrate the algorithm to compute an approximate GCD. Both the theoretical analysis and the computational results show that the method is feasible.

Keywords: Sylvester Matrix; Approximate Greatest Common Divisor; Low Rank Approximation; Structured Total Least Norm; Numerical Method

1. Introduction

Let $\deg\left(f\left(x\right)\right)$ be the degree of $f\left(x\right)$ and $C\left[x\right]$ be the set of univariate polynomials. $\|A\|_2$ stands for the spectral norm of the matrix A. C^n and $C^{m\times n}$ are the vector spaces of complex n vectors and $m\times n$ matrices, respectively. Transpose matrices and vectors are denoted by A^T and u^T. $GCD\left(f,g\right)$ denotes the greatest common divisor for the polynomials f and g. We use $rank\left(A\right)$ to stand for the rank of matrix A.

In this paper, we consider the following problem. Let $f_1\left(x\right),\ f_2\left(x\right),\ \cdots,\ f_t\left(x\right)\in C\left[x\right]\setminus\{0\}$, namely

$$f_1\left(x\right)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0,$$

$$f_i\left(x\right)=b_{ip}x^p+b_{i(p-1)}x^{p-1}+\cdots+b_{i1}x+b_{i0},i=2,\cdots,t.$$

Problem 1.1. Set k be a positive integer with $k\le\min\left(n,p\right)$. We wish to compute $\Delta f_1\left(x\right)$, $\Delta f_2\left(x\right)$, $\cdots$, $\Delta f_t\left(x\right)\in C\left[x\right]\setminus\{0\}$, such that

$$\deg\left(\Delta f_1\left(x\right)\right)\le n,\ \deg\left(\Delta f_i\left(x\right)\right)\le p,\ 2\le i\le t,$$

$$\deg\left(GCD\left(f_1\left(x\right)+\Delta f_1\left(x\right),f_2\left(x\right)+\Delta f_2\left(x\right),\cdots,f_t\left(x\right)+\Delta f_t\left(x\right)\right)\right)\ge k,$$

and

$$\left\|\Delta f_1\left(x\right)\right\|_2^2+\left\|\Delta f_2\left(x\right)\right\|_2^2+\cdots+\left\|\Delta f_t\left(x\right)\right\|_2^2$$

is minimized.

The problem of computing approximate GCD of several polynomials is widely applied in speech encoding and filter design [1], computer algebra [2] and signal processing [3] and has been studied in [4-7] in recent years. Several methods to the problem have been presented. The generally-used computational method is based on the truncated singular decomposition(TSVD) [8] which may not be appropriate when a matrix has a special structure since they do not preserve the special structure (for example, Sylvester matrix). Another common method based on QR decomposition [9,10] may suffer from loss of accuracy when it is applied to ill-conditioned problems and the algorithm derived in [11] can produce a more accurate result for ill-conditioned problems. Cadzow algorithm [12] is also a popular method to solve this problem which has been rediscovered in the literature [13].

*The author is supported by a grant from National Natural Science Foundation of China (11101100; 11391240185; 11261014; 11226323) and Natural Science Foundation of Guangxi Province (No.2012GXN SFBA053006; 2013GX NSFBA019009; 2011GXNSFA018138).

Somehow it only finds a structured low rank matrix that is nearby a given target matrix but certainly is not the closet even in the local sense. Another method is based on alternating projection algorithm [14]. Although the algorithm can be applied to any low rank and any linear structure, the speed may be very slow. Some other methods have been proposed such as the ERES method [15], STLS method [16] and the matrix pencil method [17]. An approach to be described is called Structured Total Least Norm (STLN) which has been described for Hankel structure low rank approximation [18,19] and Sylvester structure low rank approximation with two polynomials [20]. STLN is a problem formulation for obtaining an approximate solution $(A+E)X = B+H$ to an overdetermined linear system $AX = B$ preserving the given structure in A or $[A\ B]$.

In this paper, we apply the algorithm to compute the structured preserving rank reduction of Sylvester matrix. We introduce some notations and discuss the relationship between the GCD problems and low rank approximation of Sylvester matrices in Section 2. Based on STLN method, we describe the algorithm to solve Problem 1.1 in Section 3. In Section 4, we use some examples to illustrate the method is feasible.

2. Main Results

First of all, we shall prove that Problem 1.1 always has a solution.

Theorem 2.1. Suppose that $f_1,\ f_2,\ \cdots,\ f_t$, $\deg(f_1),\ \deg(f_2),\ \cdots,\ \deg(f_t)$ and k are defined as those in Problem 1.1. There exist $\hat{f}_1$, $\hat{f}_i \in C[x]$ with $\deg(\hat{f}_1) \le n$, $\deg(\hat{f}_i) \le p$ and $\deg(GCD(\hat{f}_1, \hat{f}_2, \cdots, \hat{f}_t)) \ge k$ such that for all $\bar{f}_1$, $\bar{f}_i \in C[x]$ with $\deg(\bar{f}_1) \le n$, $\deg(\bar{f}_i) \le p$ and $\deg(GCD(\bar{f}_1, \bar{f}_2, \cdots, \bar{f}_t)) \ge k,\ \ 2 \le i \le t$.

We have

$$\left\|\hat{f}_1 - f_1\right\|_2^2 + \left\|\hat{f}_2 - f_2\right\|_2^2 + \cdots + \left\|\hat{f}_t - f_t\right\|_2^2 \le \left\|\bar{f}_1 - f_1\right\|_2^2 + \left\|\bar{f}_2 - f_2\right\|_2^2 + \cdots + \left\|\bar{f}_t - f_t\right\|_2^2.$$

Proof. Let $h \in C[x]$ be monic with $\deg(h) = k$ and set $u_i \in C[x]$ with $\deg(u_i) \le \deg(f_i) - k$. For the real and imaginary parts of the coefficients of h and of u_i, $(1 \le i \le t)$. We are considered with the continuous objective function

$$F(h, u_1, u_2, \cdots, u_t) = \|u_1 h - f_1\|_2^2 + \|u_2 h - f_2\|_2^2 + \cdots + \|u_t h - f_t\|_2^2.$$

We will prove that the function has a value on a closed and bounded set of its real argument vector which is smaller than elsewhere. Consider $\bar{f}_1 = a_n x^n$ and $\bar{f}_i = b_{ip} x^p$ with a GCD of degree $\ge k$ for $2 \le i \le t$. Clearly, any h and u_i with

$$F(h, u_1, u_2, \cdots, u_t) > \left\|\bar{f}_1 - f_1\right\|_2^2 + \left\|\bar{f}_2 - f_2\right\|_2^2 + \cdots + \left\|\bar{f}_t - f_t\right\|_2^2$$

can be discarded. So from above,we know that the coefficients of $u_1 h,\ u_2 h,\ \cdots,\ u_t h$ can be bounded and so can the coefficients of $h,\ u_1,\ u_2,\ \cdots,\ u_t$ by any polynomials factor coefficient bound. Thus the function's domain $F(h, u_1, u_2, \cdots, u_t)$ is restricted to a sufficiently large ball. It remains to exclude $u_1 = u_2 = \cdots = u_t = 0$ as the minimal solution. We have

$$F(h, 0, 0, \cdots, 0) = \|f_1\|_2^2 + \|f_2\|_2^2 + \cdots + \|f_t\|_2^2 > \left\|\bar{f}_1 - f_1\right\|_2^2 + \left\|\bar{f}_2 - f_2\right\|_2^2 + \cdots + \left\|\bar{f}_t - f_t\right\|_2^2.$$

In conclusion, the theorem is true.

Now we begin to solve Problem 1.1, we first define a $p \times (n+p)$ matrix associated with $f_1(x)$ as follows

$$S_1 = \begin{bmatrix} a_n & a_{n-1} & a_{n-2} & \cdots & a_1 & a_0 & 0 & \cdots & 0 \\ 0 & a_n & a_{n-1} & \cdots & \cdots & a_1 & a_0 & \ddots & \\ \vdots & & \ddots & \ddots & & & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & a_n & a_{n-1} & \cdots & \cdots & a_1 & a_0 \end{bmatrix},$$

and an $n \times (n+p)$ matrix associated with $f_i(x)$, $i = 2, 3, \cdots, t$ as

$$S_i = \begin{bmatrix} b_{ip} & b_i(p-1) & b_i(p-2) & \cdots & b_{i1} & b_{i0} & 0 & \cdots & \cdots & 0 \\ 0 & b_{ip} & b_i(p-1) & \cdots & \cdots & b_{i1} & b_{i0} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & & & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & & & & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & b_{ip} & b_i(p-1) & \cdots & \cdots & \cdots & b_{i1} & b_{i0} \end{bmatrix},$$

An extended Sylvester matrix or a generalized resultant is then defined by

$$S=\begin{bmatrix}S_1\\S_2\\\vdots\\S_t\end{bmatrix}^{\mathrm{T}}\in C^{(p+n)\times((t-1)n+p)}.$$

Deleting the last $k-1$ rows of S and the last $k-1$ columns of coefficients of $f_1, f_2, \cdots, f_t$ separately is S, We get the k-th Sylvester matrix $S_k \in C^{(n+p-k+1)\times((t-1)n+p-tk+t)}$

$$S_k=\begin{bmatrix}a_n & 0 & & b_{2p} & & & \cdots & b_{tp} & & \\ a_{n-1} & \ddots & & b_{2(p-1)} & \ddots & & \cdots & b_{t(p-1)} & \ddots & \\ \vdots & \ddots & a_n & \vdots & \ddots & b_{2p} & \vdots & \vdots & \ddots & b_{tp}\\ a_0 & & a_{n-1} & b_{10} & & b_{2(p-1)} & \cdots & b_{t0} & & b_{t(p-1)}\\ & \ddots & \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots\\ & & a_0 & & & b_{20} & \cdots & & & b_{t0}\end{bmatrix}.$$

It is well-known that $\deg(GCD(f_1\cdots f_t))\ge k$ if and only if $S_k(f_1\cdots f_t)$ has rank deficiency at least 1. We have

$$\min_{\deg(GCD(\bar f_1\cdots\bar f_t))\ge k}\|\bar f_1-f_1\|_2^2+\|\bar f_2-f_2\|_2^2+\cdots+\|\bar f_t-f_t\|_2^2$$
$$\Leftrightarrow \min_{\mathrm{dimNullspace}(\bar S_k)\ge 1}\|\bar f_1-f_1\|_2^2+\|\bar f_2-f_2\|_2^2+\cdots+\|\bar f_t-f_t\|_2^2. \tag{2.1}$$

where $\bar S_k$ is the k-th Sylvester matrix generated by $\bar f_1, \bar f_2, \cdots, \bar f_t$.

From above, we know that (2.1) can be transformed to the low rank approximation of a Sylvester matrix.

If we use STLN [16] to solve the following overdetermined system

$$A_kX=b_k,$$

for $S_k=[b_k \ \ A_k]$, where b_k is the first column of S_k and A_k are the remainder columns of S_k, then we get a minimal perturbation $[h_k \ \ E_k]$ of Sylvester structure satisfying

$$b_k+h_k\in A_k+E_k.$$

So the solution with Sylvester structure is $\bar S_k=[h_k+b_k \ \ E_k+A_k]$ and $\mathrm{dimNullspace}(\bar S_k)>1$.

We will give the following example and theorem to explain why we choose the first column to form the overdetermined system.

Example 2.1. Suppose three polynomials are given

$$f_1(x)=x^2-1,$$
$$f_2(x)=x^2+x-2,$$
$$f_3(x)=x^2+2x-3.$$

The matrix S is the Sylvester matrix generated by $f_1(x)$, $f_2(x)$ and $f_3(x)$

$$S=\begin{bmatrix}1&0&1&0&1&0\\0&1&1&1&2&1\\-1&0&-2&1&-3&2\\0&-1&0&-2&0&-3\end{bmatrix}$$

The matrix S is partitioned as $S=[\hat b_1 \ \ \hat A_1]$ or $S=[\bar A_1 \ \ \bar b_1]$, where $\hat b_1$ is the first column of S, whereas $\bar b_1$ is the last column of S.

The overdetermined system

$$\hat A_1X=\hat b_1$$

has a solution $X=[1 \ \ 1 \ \ 1 \ \ -1 \ \ -1]^{\mathrm{T}}$, while the system

$$\bar A_1X=\bar b_1$$

has no solution.

Theorem 2.2. Suppose that $f_1, f_2, \cdots, f_t$, $\deg(f_1), \deg(f_2), \cdots, \deg(f_t)$ and k are defined as those in Problem 1.1 and S_k is the k-th Sylvester matrix of $f_1, f_2, \cdots, f_t$. Then the following statements are equivalent.

1) $\mathrm{dimNullspace}(S_k)\ge 1$;

2) $A_kX=b_k$ has a solution, where b_k is the first column of S_k and A_k are the remainder columns of S_k.

Proof. $\Leftarrow$ Suppose $A_kX=b_k$ has a nonzero solution, then $b_k\in\mathrm{Range}(A_k)$. Since b_k is the first column of S_k, obviously, the dimension of the rank deficiency of $S_k=[b_k \ \ A_k]$ is at least 1.

$\Rightarrow$ Suppose the rank deficiency of $S_k=[b_k \ \ A_k]$ is at least 1 and $D(x)=GCD(f_1, f_2, \cdots, f_t)$, $f_1^*=f_1/D(x)$, $f_2^*=f_2/D(x)$, $\cdots$, $f_t^*=f_t/D(x)$. Multiplying the vector $V=[x^{n+p-k}, x^{n+p-k-1}, \cdots x, 1]$ to the matrix S_k, then we obtain

$$VS_k=[x^{p-k}f_1, x^{p-k-1}f_1, \cdots, f_1, x^{n-k}f_2, \cdots, f_2, \cdots, x^{n-k}f_t, \cdots, f_t]. \tag{2.2}$$

Next we will prove that $A_k X = b_k$ has a solution. If we multiply the vector V to two sides of the equation $A_k X = b_k$, it turns out to be

$$\begin{aligned}&\left[x^{p-k-1} f_1, \cdots, f_1, x^{n-k} f_2, \cdots, f_2, \cdots, x^{n-k} f_t, \cdots, f_t\right] X \\ &= x^{p-k} f_1.\end{aligned} \tag{2.3}$$

The solution X of equation (2.3) is equal to the coefficients of polynomials u_1, u_2, $\cdots$, u_t such that

$$x^{p-k} f_1 = u_1 f_1 + u_2 f_2 + \cdots + u_t f_t.$$

We can get $\deg\left(D(x)\right) \geq k$ and $\deg\left(f_1^*\right) \leq n-k,$ $\deg\left(f_i^*\right) \leq p-k,$ $(2 \leq i \leq t)$ from $\dim\mathrm{Nullspace}\left(S_k\right) \geq 1$. Dividing x^{p-k} by $f_2^* + \cdots + f_t^*$, we obtain a quotient q and a remainder m satisfy

$$x^{p-k} = q\left(f_2^* + \cdots + f_t^*\right) + m.$$

where $\deg(q) \leq \deg\left(D(x)\right) - k,$ $\deg(m) \leq p-k-1$. Now we can get that

$$u_1 = p, u_2 = mf_1^*, \cdots, u_t = mf_1^*.$$

are solutions of Equation (2.3), since

$$\deg\left(u_1\right) \leq \deg\left(D(x)\right) - k,$$

$$\begin{aligned}\deg\left(u_i\right) &= \deg(q) + \deg\left(f_1^*\right) \\ &\leq \deg\left(D(x)\right) - k + \deg\left(f_1^*\right) \leq n-k,\end{aligned}$$

and

$$u_1 f_1 + u_2 f_2 + \cdots + u_t f_t = fq\left(f_2^* + \cdots + f_t^*\right) + fp = fx^{p-k}.$$

Next, we will illustrate for any given Sylvester matrix, as long as all the elements are allowed to be perturbed, we can always find k -Sylvester structure matrices $\left[h_k \quad E_k\right]$ satisfy $b_k + h_k \in \mathrm{Range}\left(A_k + E_k\right)$, where b_k is the first column of S_k and A_k are the remainder column of S_k.

Theorem 2.3. Given the positive integer n, p, t, there exists a Sylvester matrix $S \in C^{(n+p)\times(p+(t-1)n)}$ with rank deficiency k.

Proof. We can always find polynomials $f_1, f_2, \cdots, f_t \in C[x]$ with $\deg\left(f_1\right) = n$, $\deg\left(f_i\right) = p, 2 \leq i \leq t$ and $\deg\left(GCD\left(f_1, f_2, \cdots, f_t\right)\right) = k$. Hence S is the Sylvester matrix of $f_1, f_2, \cdots, f_t$ and its rank deficiency is k.

Corollary 2.1. Given the positive integer n, p, t, and k -th Sylvester matrix $S_k = \left[A_k \quad b_k\right]$, where $A_k \in C^{(n+p-k+1)\times(p+(t-1)n-tk+t)}$, $b_k \in C^{(n+p-k+1)\times 1}$, it is always possible to find a k -th Sylvester structure perturbation $\left[h_k \quad E_k\right]$ such that $b_k + h_k \in \mathrm{Range}\left(A_k + E_k\right)$.

3. STLN for Overdetermined System with Sylvester Structure

In this section, we will use STLN method to solve the overdetermined system

$$A_k X = b_k,$$

According to theorem 2.3 and corollary 2.1, we can always find Sylvester structure $\left[h_k \quad E_k\right]$ with $h_k + b_k \in \mathrm{Range}\left(A_k + E_k\right)$. Next we will use STLN method to find the minimum solution.

First, we define the Sylvester structure preserving perturbation $[h_k \quad E_k]$ of S_k

$$\left[h_k \quad E_k\right] = \begin{bmatrix} z_1 & & & z_{n+2} & & & \cdots & z_{n+(t-2)p+t} & & \\ z_2 & \ddots & & z_{n+3} & \ddots & & \cdots & z_{n+(t-2)p+t+1} & \ddots & \\ \vdots & \ddots & z_1 & \vdots & \ddots & z_{n+2} & \vdots & \vdots & \ddots & z_{n+(t-2)p+t} \\ z_{n+1} & & z_2 & z_{n+p+2} & & z_{n+3} & \cdots & z_{n+(t-1)p+t} & & z_{n+(t-2)p+t+1} \\ & \ddots & \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots \\ & & z_{n+1} & & & z_{n+p+2} & \cdots & & & z_{n+(t-1)p+t} \end{bmatrix}$$

can be represented by a vector

$$Z = \left[z_1, z_2, \cdots, z_{(n+(t-1)p+t)}\right]^{\mathrm{T}}.$$

We can define a matrix P_k such that $h_k = P_k Z$.

$$P_k = \begin{bmatrix} I_{n+1} & 0 \\ 0 & 0 \end{bmatrix} \in C^{(n+p-k+1)\times(n+(t-1)p+t)},$$

where I_{n+1} is a $(n+1)\times(n+1)$ identity matrix.

We will solve the equality-constrained least squares problem

$$\min_{Z,X} \|Z\|_2^2, \text{subject to } r = 0. \tag{3.1}$$

where the structured residual r is

$$r = r(Z, X) = b_k + h_k - \left(A_k + E_k\right) X.$$

By using the penalty method, the formulation (3.1) can be transformed into

$$\min_{Z,X} \left\| \begin{matrix} wr(Z, X) \\ Z \end{matrix} \right\|_2^2, w \gg 1, \tag{3.2}$$

where w is a large penalty value.

Let ΔZ and ΔX stand for a small change in Z and X, respectively and ΔE_k be the corresponding change in E_k. Then the first order approximate to $r\left(Z+\Delta Z, X+\Delta X\right)$ is

$$\begin{aligned} r\left(Z+\Delta Z, X+\Delta X\right) &= b_k + P_k\left(Z+\Delta Z\right) \\ &\quad -\left(A_k+E_k+\Delta E_k\right)\left(X+\Delta X\right) \\ &\approx b_k + P_k Z - \left(A_k+E_k\right)X + P_k\Delta Z \\ &\quad -\left(A_k+E_k\right)\Delta X - \Delta E_k X. \end{aligned}$$

Introducing a matrix of Sylvester structure Y_k and

$$X = \left[x_1, x_2, \cdots, x_{p+(t-1)n-tk+t-1}\right]^{\mathrm{T}}$$

(3.2) can be approximated by

$$\min_{\Delta X,\Delta Z}\left\|\begin{bmatrix} w\left(Y_k-P_k\right) & w\left(A_k+E_k\right) \\ I_{n+(t-1)p+t} & 0 \end{bmatrix}\begin{bmatrix}\Delta Z \\ \Delta X\end{bmatrix}+\begin{bmatrix}-wr \\ Z\end{bmatrix}\right\|_2 \quad (3.3)$$

where $Y_k \in C^{(n+p-k+1)\times(n+(t-1)p+t)}$ satisfies that

$$Y_k Z = E_k X. \quad (3.4)$$

In the following, we present a method to obtain the matrix Y_k. Suppose f_1, f_2, $\cdots$, f_t, E, Z and X are defined as above. Multiplying the vector

$$V = \left[x^{n+p-k}, x^{n+p-k-1}, \cdots, x, 1\right],$$

to the two sides of equation (3.4), it becomes

$$VY_k Z = VE_k X.$$

Let $\hat{X} = \begin{bmatrix} 0 \\ X \end{bmatrix}$, we obtain

$$VE_k X = V\left[h_k \quad E_k\right]\hat{X} = \hat{g}_1\hat{u}_1 + \hat{g}_2\hat{u}_2 + \cdots + \hat{g}_t\hat{u}_t, \quad (3.5)$$

where $\hat{g}_1$ is the polynomial with degree n which is generated by the subvector of Z:

$$\left[z_1 \quad z_2 \quad \cdots \quad z_{n+1}\right],$$

$\hat{g}_2$ is the polynomial with degree p which is generated by the subvector of Z:

$$\left[z_{n+2} \quad z_{n+3} \quad \cdots \quad z_{n+p+2}\right],$$

$\vdots$

$\hat{g}_t$ is the polynomial with degree p which is generated by the subvector of Z:

$$\left[z_{n+(t-2)p+t} \quad z_{n+(t-2)p+t+1} \quad \cdots \quad z_{n+(t-1)p+t}\right],$$

$\hat{u}_1$ is the polynomial with degree $p-k-1$ which is generated by the subvector of X:

$$\left[0 \quad x_1 \quad \cdots \quad x_{p-k}\right],$$

$\hat{u}_2$ is the polynomial with degree $n-k$ which is generated by the subvector of X:

$$\left[x_{p-k+1} \quad x_{p-k+2} \quad \cdots \quad x_{n+p-2k+1}\right],$$

$\vdots$

$\hat{u}_t$ is the polynomial with degree $n-k$ which is generated by the subvector of X:

$$\left[x_{p+(t-2)n-(t-1)k+t-1} \quad x_{p+(t-2)n-(t-1)k+t} \quad \cdots \quad x_{p+(t-1)n-tk+t-1}\right],$$

Here we will present a simple example to illustrate how to find Y_k.

Example 3.1. Suppose $k=1$, $X=\left[x_1, x_2, \cdots, x_7\right]$ and $f_1\left(x\right) = a_3x^3 + a_2x^2 + a_1x + a_0$,

$$f_2\left(x\right) = b_{22}x^2 + b_{21}x + b_{20},$$

$$f_3\left(x\right) = b_{32}x^2 + b_{31}x + b_{30}.$$

$$A = \begin{bmatrix} 0 & b_{22} & 0 & 0 & b_{32} & 0 & 0 \\ a_3 & b_{21} & b_{22} & 0 & b_{31} & b_{32} & 0 \\ a_2 & b_{20} & b_{21} & b_{22} & b_{30} & b_{31} & b_{32} \\ a_1 & 0 & b_{20} & b_{21} & 0 & b_{30} & b_{31} \\ a_0 & 0 & 0 & b_{20} & 0 & 0 & b_{30} \end{bmatrix}, b_1 = \begin{bmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \\ 0 \end{bmatrix},$$

then

$$Y_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & x_2 & 0 & 0 & x_5 & 0 & 0 \\ x_1 & 0 & 0 & 0 & x_3 & x_2 & 0 & x_6 & x_5 & 0 \\ 0 & x_1 & 0 & 0 & x_4 & x_3 & x_2 & x_7 & x_6 & x_5 \\ 0 & 0 & x_1 & 0 & 0 & x_4 & x_3 & 0 & x_7 & x_6 \\ 0 & 0 & 0 & x_1 & 0 & 0 & x_4 & 0 & 0 & x_7 \end{bmatrix}.$$

4. Approximate GCD Algorithm and Experiments

The following algorithm is designed to solve Problem 1.1.

Algorithm 4.1.

Input-A Sylvester matrix S generated by f_1, f_2, $\cdots$, f_t, respectively, an integer k and a tolerance tol.

Output-Polynomials $\overline{f}_1$, $\overline{f}_2$, $\cdots$, $\overline{f}_t$ and the Euclidean distance $\left\|\overline{f}_1 - f_1\right\|_2^2 + \left\|\overline{f}_2 - f_2\right\|_2^2 + \cdots + \left\|\overline{f}_t - f_t\right\|_2^2$ is to a minimum.

1) Form the k-th Sylvester matrix S_k as the above section, set the first column of S_k as b_k and the remainder columns of S_k as A_k. Let $E_k = 0$, $h_k = 0$.

2) Calculate X from $\min\left\|A_k X - b_k\right\|_2$ and $r = b_k - A_k X$. Compute P_k and Y_k as the above section.

3) Repeat

(1) $$\min_{\Delta X,\Delta Z}\left\|\begin{bmatrix} w\left(Y_k-P_k\right) & w\left(A_k+E_k\right) \\ I_{n+(t-1)p+t} & 0 \end{bmatrix}\begin{bmatrix}\Delta Z \\ \Delta X\end{bmatrix}+\begin{bmatrix}-wr \\ Z\end{bmatrix}\right\|_2.$$

(2) Let $X = X + \Delta X$, $Z = Z + \Delta Z$,

(3) Form the matrix E_k and h_k from Z, and Y_k from X. Let $A_k = A_k + E_k$,

$b_k = b_k + h_k$, $r = b_k - A_k X$ until $\|\Delta X\|_2 \le tol$ and $\|\Delta Z\|_2 \le tol$

4) Output the polynomials $\bar{f}_1, \bar{f}_2, \cdots, \bar{f}_t$ constructed from $[b_k \ A_k]$.

Given a tolerance ε, we can use the Algorithm 4.1 to compute an approximate GCD of $f_1, f_2, \cdots, f_t$. The method begin with $k \le \min\{n, p\}$ using Algorithm 4.1 to compute the minimum perturbation $N = \sum_i \|\bar{f}_i - f_i\|_2^2$ with $\text{dimNullspace}(S) \ge 1$. If $N < \varepsilon$, then we can compute the approximate GCD form matrix S_k. Otherwise, we reduce k by one and repeat the Algorithm 4.1.

Example 4.1. We wish to find the minimal polynomial perturbations Δf and Δg of

$$f = x^2 - 6x + 5,$$

$$g = x^2 - 6.3x + 5.72,$$

satisfy that the polynomials $\Delta f + f$ and $\Delta g + g$ have a common root. We take two cases into account.

Case 1: The leading coefficients can be perturbed. Let $k = 1$ and $tol = 10^{-3}$, after 3 iterations, we get the polynomials $\bar{f}$ and $\bar{g}$

$$\bar{f} = 0.9850x^2 - 6.0029x + 4.9994,$$

$$\bar{g} = 1.0150x^2 - 6.2971x + 5.7206,$$

with a minimum distance

$$N = \|\bar{f} - f\|_2^2 + \|\bar{g} - g\|_2^2 = 0.0004663.$$

Case 2: The leading coefficients can be perturbed. Let $k = 1$ and $tol = 10^{-3}$, after 3 iterations, we have the polynomials $\bar{f}$ and $\bar{g}$:

$$\bar{f} = x^2 - 6.0750x + 4.9853,$$

$$\bar{g} = x^2 - 6.222x + 5.7353,$$

with a minimum distance

$$N = \|\bar{f} - f\|_2^2 + \|\bar{g} - g\|_2^2 = 0.01213604583.$$

Example 4.2. Let $k = 1$, $tol = 10^{-3}$ and

$$f_1 = 1.98x^3 + 5x^2 + 5x + 2.96,$$

$$f_2 = 1.99x^2 - 1.01 + 3.01,$$

$$f_3 = 2x^2 + 4.99x + 2.99,$$

after 8 iterations, we have the polynomials

$$\bar{f}_1 = 1.9800x^3 + 5.0000x^2 + 5.0000x + 2.9600,$$

$$\bar{f}_2 = 1.9963x^2 + 1.0100x - 3.0099,$$

$$\bar{f}_3 = 2.0071x^2 + 4.9900x + 2.9902,$$

with a minimum distance

$$N = \|\bar{f}_1 - f_1\|_2^2 + \|\bar{f}_2 - f_2\|_2^2 + \|\bar{f}_3 - f_3\|_2^2 = 9.0763 \times 10^{-5},$$

and the CPU time

$$t = 0.974920(s)$$

Example 4.3. Let $k = 1.$, $tol = 10^{-3}$ and

$$f_1 = 1.85x^4 - 2x^3 - 2.69x^2 - 1.42x,$$

$$f_2 = 1.47x^3 - 2.94x^2 + 1.18x - 2.36,$$

$$f_3 = 0.52x^3 - 4.01x^2 + 5.94x,$$

$$f_4 = 0.52x^3 - 0.13x^2 - 1.05x - 2.58.$$

after 11 iterations, we have the polynomials

$$\bar{f}_1 = 1.85x^4 - 2x^3 - 2.69x^2 - 1.42x,$$

$$\bar{f}_2 = 1.47x^3 - 2.94x^2 + 1.18x - 2.36,$$

$$\bar{f}_3 = 0.5242x^3 - 4.01x^2 + 5.9405x,$$

$$\bar{f}_4 = 0.52x^3 - 0.134x^2 + -1.05x - 2.58.$$

with a minimum distance

$$N = \|\bar{f}_1 - f_1\|_2^2 + \|\bar{f}_2 - f_2\|_2^2 + \|\bar{f}_3 - f_3\|_2^2 + \|\bar{f}_4 - f_4\|_2^2$$
$$= 3.4387 \times 10^{-5},$$

and the CPU time

$$t = 0.642582(s).$$

Example 4.4. Let $k = 2$, $tol = 10^{-3}$ and

$$f_1 = 0.144x^4 - 0.761x^3 + 1.316x^2 - 0.74x,$$

$$f_2 = 0.393x^3 - 2.212x^2 + 4.132x - 2.56$$

$$f_3 = 0.182x^3 - 0.358x^2 - 0.752x + 1.48,$$

$$f_4 = 0.173x^3 - 0.544x^2 + 0.01x + 0.592.$$

after 1 iteration, we have the polynomials

$$f_1 = 0.144x^4 - 0.761x^3 + 1.316x^2 - 0.74x,$$

$$f_2 = 0.393x^3 - 2.212x^2 + 4.132x - 2.56,$$

$$f_3 = 0.182x^3 - 0.3548x^2 - 0.752x + 1.48,$$

$$f_4 = 0.173x^3 - 0.4701x^2 + 0.01x + 0.6257.$$

with a minimum distance

$$N = \|\bar{f}_1 - f_1\|_2^2 + \|\bar{f}_2 - f_2\|_2^2 + \|\bar{f}_3 - f_3\|_2^2 + \|\bar{f}_4 - f_4\|_2^2$$
$$= 0.0066,$$

and the CPU time

$$t = 0.001335(s).$$

Example 4.5. Let $k = 2$, $tol = 10^{-3}$ and

$$f_1 = 0.64x^6 - 2.56x^5 + 2.56x^4 + 0.62x^3 - 1.03x^2 - 3.32x + 5.8,$$

$$f_2 = 1.85x^5 - 5.7x^4 + 1.31x^3 + 3.96x^2 + 2.84,$$

$$f_3 = 0.44x^5 - 3.23x^4 + 7.64x^3 - 4.7x^2 - 4.72x + 4.72,$$

$$f_4 = 1.23x^5 - 4.92x^4 + 2.65x^3 + 9.645x^2 - 11.12x + 2.04,$$

$$f_5 = 1.31x^5 - 5.76x^4 + 8.49x^3 - 5.47x^2 - 0.48x + 5.16.$$

after 1 iteration, we have the polynomials

$$f_1 = 0.6401x^6 - 2.56x^5 + 2.5603x^4 + 0.62x^3 - 1.03x^2 - 3.32x + 5.8,$$

$$f_2 = 1.85099x^5 - 5.7x^4 + 1.3102x^3 + 3.96x^2 + 2.84,$$

$$f_3 = 0.4389x^5 - 3.23x^4 + 7.64x^3 - 4.7003x^2 - 4.72x + 4.72,$$

$$f_4 = 1.23x^5 - 4.92x^4 + 2.65x^3 + 9.645x^2 - 11.12x + 2.04,$$

$$f_5 = 1.307x^5 - 5.76x^4 + 8.4902x^3 - 5.47x^2 - 0.48x + 5.16.$$

with a minimum distance

$$N = \left\|\bar{f}_1 - f_1\right\|_2^2 + \left\|\bar{f}_2 - f_2\right\|_2^2 + \left\|\bar{f}_3 - f_3\right\|_2^2 + \left\|\bar{f}_4 - f_4\right\|_2^2 + \left\|\bar{f}_5 - f_5\right\|_2^2 = 1.1460 \times 10^{-5},$$

and the CPU time

$$t = 0.0923583(s).$$

Examples 4.1, 4.2, 4.3, 4.4 and 4.5 show that Algorithm 4.1 is feasible to solve Problem 1.1.

In **Table 1**, we present the performance of Algorithm 4.1 and compare the accuracy of the new fast algorithm with the algorithms in [9,21]. Denote n be the total degree of polynomials f_1 and p be the total degree of polynomials f_i, $2 \le i \le t$. It (Chu) stands for the number of iterations by the method in [14] whereas it (STLN) denotes the number of iterations by Algorithm 4.1. Denoted by error(Zeng) and error (STLN) are the perturbations $\sum_i \left\|\bar{f}_i - f_i\right\|_2^2$ computed by the method in [21] and Algorithm 4.1, respectively. The last two columns denote the CPU time in seconds costed by AFMP algorithm and our algorithm, respectively.

As shown in the above table, we show that our method based on STLN algorithm converges quickly to the minimal approximate solutions, needing no more than 2 iterations whereas the method in [14] requires more iteration steps. We also note that our algorithm still converges very quickly when the degrees of polynomials become large while the algorithm in [14] needs more iteration steps. Besides, our algorithm needs less CPU time than the AFMP algorithm. So the convergence speed of our method is faster. From the errors, we demonstrate that our method has smaller magnitudes compared with the method in [21]. So our algorithm can generate much more accurate solutions.

5. Conclusion

In this paper, we present that approximation GCD of several polynomials can be solved by a practical and reliable way based on STLN method and transformed to the approximation of Sylvester structure problem. For the matrices related to the minimization problems are all structured matrix with low displacement rank, applying the algorithm to solve these minimization problems would be possible. The complexity of the algorithm is reduced with respect to the degrees of the given polynomials. Although the problem of structured low rank ap-

Table 1. Algorithm performance on benchmarks.

Ex	n, p	k	it (Chu)	it (STLN)	error (Zeng)	error (STLN)	time (s) (AFMP)	time (s) (STLN)
1	2, 2	1	5	2	1.89e-4	2.87e-5	7.76	2.5
2	3, 3	2	8	2	1.36e-3	1.05e-4	19.51	7.81
3	5, 4	3	11	2	1e-3	1.56e-6	6.81	2.44
4	6, 6	4	23	2	1.46e-3	1.96e-10	31.829	12.08
5	8, 7	5	33	2	6.53e-4	1.98e-16	50.222	16.95
6	10, 10	6	43	2	1.61e-3	1.51e-12	157.09	61.40
7	14, 13	7	58	2	1.23e-3	2.61e-4	273.7	122.31
8	28, 28	10	634	2	2.6e-3	3.54e-4	559.3	210.65

proximation has been studied in many literatures and obtained many accomplishments, there is still much work to be done, for example, low rank approximation of finite dimensional matrix has not been fully resolved.

REFERENCES

[1] B. DeMoor, "Total Least Squares for Affinely Structured Matrices and the Noisy Realization Problem," *IEEE Transactions on Signal Process*, Vol. 42, No. 11, 1994, pp. 3104-3113. http://dx.doi.org/10.1109/78.330370

[2] R. M. Corless, P. M. Gianni, B. M. Tragerm and S. M. Watt, "The Singular Value Decomposition for Polynomial System," *Proceedings of International Symposium on Symbolic and Algebraic Computation*, Montreal, 1995, pp. 195-207

[3] S. R. Khare, H. K. Pillai and M. N. Belur, "Numerical Algorithm for Structured Low Rank Approximation Problem," *Proceeding of the* 19*th International Symposium on Mathematical Theory of Networks and Systems*, Budapest, Hungary, 2010.

[4] E. Kaltofen, Z. F. Yang and L. H. Zhi, "Approximate Greatest Common Divisors of Several Polynomials with Linearly Constrained Coecients and Simgular Polynomials," Proceedings of International Symposium on Symbolic and Algebraic Computations, Genova, 2006.

[5] N. Karkanias, S. Fatouros, M. Mitrouli and G. H. Halikias, "Approximate Greatest Common Divisor of Many Polynomials, Generalised Resultants, and Strength of Approximation," *Computers & Mathematics with Applications*, Vol. 51, No. 12, 2006, pp. 1817-1830. http://dx.doi.org/10.1016/j.camwa.2006.01.010

[6] I. Markovsky, "Structured Low-Rank Approximation and Its Applications," *Automatica*, Vol. 44, No. 4, 2007, pp. 891-909. http://dx.doi.org/10.1016/j.automatica.2007.09.011

[7] D. Rupprecht, "An Algorithm for Computing Certied Approximate GCD of Univariate Polynomials," *Journal of Pure and Applied Algebra*, Vol. 139, No. 1-3, 1999, pp. 255-284. http://dx.doi.org/10.1016/S0022-4049(99)00014-6

[8] J. A. Cadzow, "Signal Enhancement: A Composite Property Mapping Algorithm," *IEEE Transactions on Acoustic Speech Signal Process*, Vol. 36, No. 1, 1988, pp. 49-62. http://dx.doi.org/10.1109/29.1488

[9] G. Cybenko, "A General Orthogonalization Technique with Applications to Time Series Analysis and Signal Processing," *Mathematics of Computation*, Vol. 40, 1983, pp. 323-336. http://dx.doi.org/10.1090/S0025-5718-1983-0679449-6

[10] J. R. Winkler and J. D. Allan, "Structured Total Least Norm and Approximate GCDs of Inexact Polynomials," *Journal of Computational and Applied Mathematics*, Vol. 215, No. 1, 2008, pp. 1-13. http://dx.doi.org/10.1016/j.cam.2007.03.018

[11] A. Frieze, R. Kannaa and S. Vempala, "Fast Monte-Carlo Algorithm for Finding Low Rank Approximations," *Journal of ACM*, Vol. 51, No. 6, 2004, pp. 1025-1041. http://dx.doi.org/10.1145/1039488.1039494

[12] R. Beer, "Quantitative *in Vivo* NMR (Nuclear Magnetic Resonance on Living Objects)," University of Technology Delft, 1995.

[13] B. Paola, "Structured Matrix-Based Methods for Approximate Polynomial GCD," Edizioni della Normale, 2011.

[14] M. T. Chu, R. E. Funderlic and R. J. Plemmons, "Structured Low Rank Approximation," *Linear Algebra Applications*, Vol. 366, No. 1, 2003, pp. 157-172. http://dx.doi.org/10.1016/S0024-3795(02)00505-0

[15] B. Beckermann and G. Labahn, "A Fast and Numerically Stable Euclidean-Like Algorithm for Detecting Relative Prime Numerical Polynomials," *IEEE Journal of Symbolic Computation*, Vol. 26, No. 6, 1998, pp. 691-714. http://dx.doi.org/10.1006/jsco.1998.0235

[16] B. Y. Li, Z. F. Yang and L. H. Zhi, "Fast Low Rank Approximation of a Sylvester Matrix by Structured Total Least Norm," *Journal of JSSAC* (*Japan Society for Symbolic and Algebraic Computation*), Vol. 11, No. 34, 2005, pp. 165-174.

[17] B. Botting, M. Giesbrecht and J. May, "Using Riemannian SVD for Problems in Approximate Algebra," *Proceedings of the* 2005 *International Workshop on Symbolic-Numeric*, 2005, Xi'an.

[18] E. Kaltofen, Z. F. Yang and L. H. Zhi, "Structured Low Rank Approximation of a Sylvester Matrix," *International Workshop on Symbolic-Numeric Computation*, Xi'an, 2005, pp. 19-21.

[19] H. Park, L. Zhang and J. B. Rosen, "Low Rank Approximation of a Hankel Matrix by Structured Total Least Norm," *BIT Numerical Mathematics*, Vol. 39, No. 4, 1999, pp. 757-779. http://dx.doi.org/10.1023/A:1022347425533

[20] L. H. Zhi and Z. F. Yang, "Computing Approximate GCD of Univariate Polynomials by Structure Total Least Norm," *Mathematics-Mechanization Research Preprints*, No. 24, 2004, pp. 375-387.

[21] Z. Zeng and B. H. Dayton, "The Approximate GCD of Inexact Polynomials Part 2: A Multivariate Algorithm," *Proceedings of the* 2004 *International Symposium on Symbolic and Algebraic Computation*, Santander, 2004.