

# A fast algorithm for solving the Sylvester structured total least squares problem<sup>☆</sup>

Bingyu Li<sup>a,\*</sup>, Zhuojun Liu<sup>b</sup>, Lihong Zhi<sup>b</sup>

<sup>a</sup>*School of Mathematics and Statistics, Northeast Normal University, Changchun 130024, China*

<sup>b</sup>*Key Laboratory of Mathematics Mechanization, Academy of Mathematics and Systems Science, Beijing 100080, China*

Received 1 October 2006; received in revised form 25 January 2007; accepted 1 March 2007

Available online 12 March 2007

## Abstract

In this paper, we develop a fast structured total least squares (STLS) algorithm for computing an approximate greatest common divisor (GCD) of two univariate polynomials. By exploiting the displacement structure of the Sylvester matrix and applying the generalized Schur algorithm, each single iteration of the proposed algorithm has quadratic computational complexity in the degrees of the given polynomials.

© 2007 Elsevier B.V. All rights reserved.

*Keywords:* Approximate greatest common divisor; Sylvester matrix; Structured total least squares; Displacement structure

## 1. Introduction

The problem of finding a greatest common divisor (GCD) of univariate polynomials with inexact coefficients is of great practical importance, for example in signal pole estimation and system identification [1–3]. Several different algorithms have been proposed in [4–20]. An approximate GCD problem can be formulated as an optimization problem:

**Problem 1.1.** *Given two univariate polynomials  $f, g \in \mathbb{R}[x]$  with  $\deg(f) = m$  and  $\deg(g) = n$ . For a positive*

*integer  $k$  with  $k \leq \min(m, n)$ , we wish to find perturbations  $\Delta f, \Delta g \in \mathbb{R}[x]$  with  $\deg(\Delta f) \leq m$  and  $\deg(\Delta g) \leq n$ , such that  $\deg(\text{GCD}(f + \Delta f, g + \Delta g)) \geq k$  and  $\|\Delta f\|_2^2 + \|\Delta g\|_2^2$  is minimized.*

In [19], the authors transformed the above problem 1.1 into the following one. Given a Sylvester matrix, compute a new Sylvester matrix of lower rank whose entries are near the corresponding entries of that input matrix. They presented an iterative algorithm based on the structured total least norm (STLN) algorithm in [21,22] to solve the approximate GCD problem. Each single iteration of their algorithm is carried out with a number of floating point operations that is of cubic order in the degrees of input polynomials. By exploiting the displacement structure of the Sylvester matrix, a fast version of the STLN algorithm was proposed in [23,24], in which each iteration has the quadratic order of complexity.

<sup>☆</sup>This research was partially supported by NKBRPC (2004CB318000) and the Chinese National Natural Science Foundation under Grant 10371127 (Liu) and 10401035 (Li and Zhi).

\*Corresponding author.

E-mail addresses: [liby@mmrc.iss.ac.cn](mailto:liby@mmrc.iss.ac.cn) (B. Li),  
[zliu@mmrc.iss.ac.cn](mailto:zliu@mmrc.iss.ac.cn) (Z. Liu), [lzhi@mmrc.iss.ac.cn](mailto:lzhi@mmrc.iss.ac.cn) (L. Zhi).

However, the fast algorithm was applied to solve the ill-conditioned least squares problems formed by using large penalty values. Hence, it may need much more iterations to achieve a minimum in some cases. In this paper, we derive a new fast algorithm which generalizes the STLS method developed in [25] for computing the approximate GCDs.

The paper is organized as follows. In Section 2, we introduce some notations on displacement structured matrices and discuss the equivalence between the approximate GCD problem and the problem of structured low rank approximation of a Sylvester matrix. In Section 3, we describe a new fast STLS algorithm to compute the structured low rank approximation of a Sylvester matrix. In Section 4, we derive a fast implementation for approximate GCD computations, and demonstrate the practical performance of the algorithm on a diverse set of pairs of univariate polynomials.

**2. Preliminary**

Let  $i$  be a positive integer,  $Z_i$  be an  $i \times i$  lower shift matrix and  $I_i$  be an  $i \times i$  identity matrix. Assume  $f$  is an univariate polynomial, then  $\|f\|_2$  denotes the Euclidean 2-norm of its coefficient vector.

The displacement of an  $n \times n$  Hermitian matrix  $T$  is defined as

$$\nabla T = T - ZTZ^T, \tag{1}$$

where  $Z$  is an  $n \times n$  lower-triangular matrix. The choice of  $Z$  depends on the matrix  $T$ , e.g., if  $T$  is a Toeplitz matrix, then  $Z$  is chosen equal to a lower shift matrix  $Z_n$ . If  $\nabla T$  has a lower rank  $\alpha$  ( $\ll n$ ) independent of  $n$ , the size of  $T$ , then  $T$  is said to be structured with respect to the displacement defined by (1) and  $\alpha$  is referred to as the displacement rank of  $T$ . It follows that  $\nabla T$  can be factored as

$$\nabla T = GJG^T,$$

where  $G$  is an  $n \times \alpha$  matrix and  $J$  is a signature matrix of the form

$$J = \begin{bmatrix} I_{p'} & \mathbf{0} \\ \mathbf{0} & -I_{q'} \end{bmatrix}, \quad p' + q' = \alpha.$$

The integers  $p', q'$  denote the numbers of positive eigenvalues and negative eigenvalues of  $\nabla T$ , respectively. The pair  $(G, J)$  is said to be a generator pair for  $T$ .

When  $T$  is strongly regular (i.e. all its leading principal submatrices are of full rank), the  $LDL^T$  decomposition of  $T$  can be efficiently carried out by the generalized Schur algorithm that operates on the generator pair  $(G, J)$  directly and costs  $O(\alpha n^2)$  flops. The notion of displacement structure can be extended to non-Hermitian matrices. A more extensive description about the generalized Schur algorithm can be found in [26].

Given two univariate polynomials  $f, g \in \mathbb{R}[x]$  with  $\deg(f) = m, \deg(g) = n$ ,

$$f = f_m x^m + \dots + f_1 x + f_0, \\ g = g_n x^n + \dots + g_1 x + g_0.$$

The Sylvester matrix generated by  $f$  and  $g$  is denoted by  $S$ . For a given positive integer  $k$  with  $k \leq \min(m, n)$ ,  $S_k \in \mathbb{R}^{(m+n-k+1) \times (m+n-2k+2)}$  is a submatrix of  $S$  obtained by deleting the last  $k - 1$  rows of  $S$  and the last  $k - 1$  columns of coefficients of  $f$  and  $g$  separately in  $S$ :

$$S_k = \begin{bmatrix} f_m & & & & g_n & & & & \\ & \ddots & & & & \ddots & & & \\ f_{m-1} & & & & g_{n-1} & & & & \\ & \ddots & & & & \ddots & & & \\ \vdots & & \ddots & f_m & \vdots & & \ddots & g_n & \\ f_0 & & & f_{m-1} & g_0 & & & g_{n-1} & \\ & & \ddots & \vdots & & & \ddots & \vdots & \\ & & & f_0 & & & & g_0 & \end{bmatrix}. \tag{2}$$

$\underbrace{\hspace{10em}}_{n-k+1} \qquad \underbrace{\hspace{10em}}_{m-k+1}$

For  $k = 1$ ,  $S_1 = S$  is the Sylvester matrix. There is a well-known strong relationship between the degree of GCD and the rank deficiency of Sylvester matrix and its  $k$ th submatrix.

**Theorem 2.1** (Emiris et al. [7], Gao et al. [27]). *Given univariate polynomials  $f, g \in \mathbb{R}[x]$ ,  $\deg(f) = m$  and  $\deg(g) = n$ , and a positive integer  $k$  with  $k \leq \min(m, n)$ .  $S(f, g)$  is the Sylvester matrix of  $f$  and  $g$ ,  $S_k$  is the  $k$ th submatrix of  $S(f, g)$ . Then the following statements are equivalent:*

- (1)  $\deg(\gcd(f, g)) \geq k$ ;
- (2)  $\text{rank}(S) \leq m + n - k$ ;
- (3) Rank deficiency of  $S_k$  is greater than or equal to one.

**Theorem 2.2** (Kaltofen et al. [19]). *Based on the same assumptions as in Theorem 2.1, let*

$S_k = [\mathbf{b}_k, A_k]$ , where  $\mathbf{b}_k$  is the first column of  $S_k$  and  $A_k$  consists of the last  $n + m - 2k + 1$  columns of  $S_k$ , then we have

$$\dim \text{Nullspace}(S_k) \geq 1 \iff A_k \mathbf{x} = \mathbf{b}_k \quad \text{has a solution.} \quad (3)$$

**Theorem 2.3** (Kaltofen et al. [19]). Given the integers  $m, n$  and  $k, k \leq \min(m, n)$ , then there exists a Sylvester matrix  $S \in \mathbb{R}^{(m+n) \times (m+n)}$  with rank  $m + n - k$ .

Based on the above theorems, given two univariate polynomials  $f$  and  $g$  and a positive integer  $k \leq \min(m, n)$ , suppose  $S_k = [\mathbf{b}_k, A_k]$ , where  $\mathbf{b}_k \in \mathbb{R}^{(m+n-k+1) \times 1}$  and  $A_k \in \mathbb{R}^{(m+n-k+1) \times (m+n-2k+1)}$ , it is always possible to find a perturbation  $[\mathbf{h}_k, E_k]$  of  $k$ th Sylvester structure such that  $\mathbf{b}_k + \mathbf{h}_k \in \text{Range}(A_k + E_k)$ .

Thus the approximate GCD Problem 1.1 can be formulated as a structure-preserving total least norm problem

$$\min_{\mathbf{z} \in \mathbb{R}^{m+n+2}} \|\mathbf{z}\|_2 \quad \text{with } (A_k + E_k)\mathbf{x} = \mathbf{b}_k + \mathbf{h}_k \quad \text{for some vector } \mathbf{x}, \quad (4)$$

where the perturbation matrix  $[\mathbf{h}_k, E_k]$  of  $S_k$  has the form:

$$[\mathbf{h}_k, E_k] = \left[ \begin{array}{cccc} z_1 & & & z_{m+2} \\ z_2 & \ddots & & z_{m+3} & \ddots \\ \vdots & \ddots & & \vdots & \ddots & z_{m+2} \\ z_{m+1} & z_2 & z_{m+n+2} & z_{m+3} \\ & \ddots & \vdots & \ddots & \vdots \\ & & z_{m+1} & & z_{m+n+2} \end{array} \right], \quad (5)$$

$\underbrace{\hspace{10em}}_{n-k+1}$        $\underbrace{\hspace{10em}}_{m-k+1}$

which is parameterized via the vector  $\mathbf{z} \in \mathbb{R}^{m+n+2}$ :

$$\mathbf{z} = [z_1, z_2, \dots, z_{m+n+1}, z_{m+n+2}]^T.$$

In the following section, we illustrate how to find a minimum solution by generalizing and applying the STLS method derived in [25]. As pointed out in [19,28], the above STLN problem may not have a solution. Moreover, the optimal solution may not correspond to a nearest GCD pair since it may correspond to polynomials of smaller degrees which remain relatively prime. Although the STLS method is not guaranteed to converge to a global minimum, in our experiments,

under some assumptions, they compute the solutions found by the global method in [8].

### 3. A new fast STLS algorithm

By introducing the Lagrangian multipliers, the constrained minimization (4) can be transformed into an unconstrained optimization problem

$$L(\mathbf{z}, \mathbf{x}, \boldsymbol{\lambda}) = \frac{1}{2} \mathbf{z}^T \mathbf{z} - \boldsymbol{\lambda}^T (\mathbf{b}_k + \mathbf{h}_k - A_k \mathbf{x} - E_k \mathbf{x}) = \frac{1}{2} \mathbf{z}^T \mathbf{z} + \boldsymbol{\lambda}^T \mathbf{r}(\mathbf{z}, \mathbf{x}). \quad (6)$$

A Sylvester-structured matrix  $X_k$  is constructed such that

$$X_k \mathbf{z} = E_k \mathbf{x} - \mathbf{h}_k = [\mathbf{h}_k, E_k] \begin{bmatrix} -1 \\ \mathbf{x} \end{bmatrix}, \quad (7)$$

where

$$X_k = \left[ \begin{array}{cccc} -1 & & & x_{n+1-k} \\ x_1 & \ddots & & x_{n+2-k} & \ddots \\ \vdots & \ddots & -1 & \vdots & \ddots & x_{n+1-k} \\ x_{n-k} & x_1 & x_{m+n+1-2k} & & & x_{n+2-k} \\ & \ddots & \vdots & & \ddots & \vdots \\ & & x_{n-k} & & & x_{m+n+1-2k} \end{array} \right]. \quad (8)$$

$\underbrace{\hspace{10em}}_{m+1}$        $\underbrace{\hspace{10em}}_{n+1}$

We initialize  $\mathbf{x}$  as the unstructured least squares solution to  $A_k \mathbf{x} \approx \mathbf{b}_k$  and  $\mathbf{z} = \boldsymbol{\lambda} = \mathbf{0}$ . Since  $A_k$  is a structured matrix with displacement rank at most 2, we can find the solution  $\mathbf{x}$  efficiently using  $O((m+n-k+1)(m+n-2k+1))$  flops based on the normal equation method [14].

We refine  $\mathbf{x}, \mathbf{z}$  and  $\boldsymbol{\lambda}^T$  simultaneously by applying the Newton's method on the Lagrangian  $L$ , which yields:

$$M \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{x} \\ \Delta \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} I_{t_1} & \mathbf{0}_{t_1 \times t_2} & X_k^T \\ \mathbf{0}_{t_2 \times t_1} & \mathbf{0}_{t_2 \times t_2} & A_k^T + E_k^T \\ X_k & A_k + E_k & \mathbf{0}_{s \times s} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{z} \\ \Delta \mathbf{x} \\ \Delta \boldsymbol{\lambda} \end{bmatrix} = - \begin{bmatrix} \mathbf{g} + J^T \boldsymbol{\lambda} \\ \mathbf{r}(\mathbf{z}, \mathbf{x}) \end{bmatrix}, \quad (9)$$

where  $t_1 = m + n + 2$ ,  $t_2 = m + n - 2k + 1$ , and  $s = m + n - k + 1$ . Here  $J = [X_k, A_k + E_k]$ , which is the Jacobian of  $\mathbf{r}(\mathbf{z}, \mathbf{x})$  with respect to the vector  $[\mathbf{z}^T, \mathbf{x}^T]^T$ , and  $\mathbf{g} = [\mathbf{z}^T, \mathbf{0}^T]^T$ . The iterative update  $\mathbf{z} = \mathbf{z} + \Delta \mathbf{z}$ ,  $\mathbf{x} = \mathbf{x} + \Delta \mathbf{x}$ , and  $\boldsymbol{\lambda} = \boldsymbol{\lambda} + \Delta \boldsymbol{\lambda}$  is stopped

when  $\|\Delta\mathbf{x}\|_2$ ,  $\|\Delta\mathbf{z}\|_2$  and  $\|\Delta\lambda\|_2$  become smaller than a given tolerance.

It can be shown that  $M$  is a matrix with displacement structure. However, since  $M$  is not strongly regular, a permutation matrix  $P$  has been introduced in [25] to transform  $M$  into  $\tilde{M}$ :

$$\tilde{M} = PMP^T = \begin{bmatrix} I_{t_1} & X_k^T & \mathbf{0}_{t_1 \times t_2} \\ X_k & \mathbf{0}_{s \times s} & A_k + E_k \\ \mathbf{0}_{t_2 \times t_1} & A_k^T + E_k^T & \mathbf{0}_{t_2 \times t_2} \end{bmatrix}. \tag{10}$$

The matrix  $\tilde{M}$  in (10) is strongly regular if the matrices  $X_k$  and  $A_k^T + E_k^T$  are all of full row rank [29]. For a properly chosen positive integer  $k \geq \deg(\gcd(f, g))$ , by Theorems 2.1, 2.2,  $A_k$  is of full column rank. We initialize  $\mathbf{x}$  as the unstructured least squares solution to  $A_k\mathbf{x} \approx \mathbf{b}_k$ , the coprime polynomials  $u$  and  $v$ , formed from the vector  $[-1, \mathbf{x}]^T$  are

$$v = -x^{n-k} + x_1x^{n-k-1} + \dots + x_{n-k},$$

$$u = x_{n+1-k}x^{m-k} + x_{n+2-k}x^{m-k-1} + \dots + x_{m+n+1-2k}.$$

The matrix  $X_k$  can be obtained from the Sylvester matrix of the polynomials  $x^{k+1}v$  and  $x^{k+1}u$  by cutting off its last  $k + 1$  rows. Since these last  $k + 1$  rows only consist of zero entries, and  $u, v$  are coprime, by Theorem 2.1, the matrix  $X_k$  is of full row rank. So the matrix  $\tilde{M}$  can be initialized to be strongly regular if we choose  $k$  properly.

Let  $\hat{M}$  denote the Schur complement of  $\tilde{M}$  with respect to the block  $I_{t_1}$ ,

$$\hat{M} = \begin{bmatrix} -X_kX_k^T & A_k + E_k \\ A_k^T + E_k^T & \mathbf{0}_{t_2 \times t_2} \end{bmatrix}. \tag{11}$$

A partial  $LDL^T$  decomposition of  $\tilde{M}$  is of the form:

$$\tilde{M} = \begin{bmatrix} I & & \\ X_k & I & \\ \mathbf{0}_{t_2 \times t_1} & & I \end{bmatrix} \begin{bmatrix} I & \\ & \hat{M} \end{bmatrix} \begin{bmatrix} I & \\ X_k & I \\ \mathbf{0}_{t_2 \times t_1} & I \end{bmatrix}^T. \tag{12}$$

**Theorem 3.1.**  $\hat{M}$  is a structured matrix with the displacement rank at most 8. We have  $\hat{M} - F\hat{M}F^T = GJG^T$ , where

$$F = \begin{bmatrix} Z_{m+n-k+1} & \\ & Z_{m+n-2k+1} \end{bmatrix}, \quad J = \begin{bmatrix} I_4 & \\ & -I_4 \end{bmatrix}$$

and

$$G = \begin{bmatrix} \mathbf{c}_6 & \mathbf{c}_5 & \mathbf{c}_4 & \mathbf{c}_3 & \mathbf{c}_1 & \mathbf{c}_2 & \mathbf{c}_5 & \mathbf{c}_6 \\ \frac{1}{2}\mathbf{e}_{n-k+1} & \frac{1}{2}\mathbf{e}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\frac{1}{2}\mathbf{e}_1 & -\frac{1}{2}\mathbf{e}_{n-k+1} \end{bmatrix},$$

here

$$\begin{aligned} \mathbf{c}_1 &= X_k(:, 1), \\ \mathbf{c}_2 &= X_k(:, m + 2), \\ \mathbf{c}_3 &= Z_{m+n-k+1}X_k(:, m + 1), \\ \mathbf{c}_4 &= Z_{m+n-k+1}X_k(:, m + n + 2), \\ \mathbf{c}_5 &= A_k(:, 1) + E_k(:, 1), \\ \mathbf{c}_6 &= A_k(:, n - k + 1) + E_k(:, n - k + 1) \\ &\quad - Z_{m+n-k+1}A_k(:, n - k) \\ &\quad - Z_{m+n-k+1}E_k(:, n - k), \end{aligned}$$

and  $Z_i$  is a lower shift matrix of order  $i$ ,  $\mathbf{e}_i$  denotes the  $i$ th column of the identity matrix  $I_{m+n-2k+1}$ .

Applying the generalized Schur algorithm on the generator pair  $(G, J)$  of  $\hat{M}$  with  $m + n - k + 1$  negative steps and  $m + n - 2k + 1$  positive steps, we obtain a backward stable factorization of  $\hat{M}$  [26,30]:

$$\hat{M} = R^TDR,$$

where  $R$  is an upper-triangular matrix and  $D$  is a signature matrix. Thus we have

$$\tilde{M} = L_1L_2D_1L_2^TL_1^T,$$

where

$$L_1 = \begin{bmatrix} I & & \\ X_k & I & \\ \mathbf{0}_{t_2 \times t_1} & & I \end{bmatrix}, \quad L_2 = \begin{bmatrix} I & \\ & R^T \end{bmatrix},$$

$$D_1 = \begin{bmatrix} I & \\ & D \end{bmatrix}.$$

The computational cost of the generalized Schur algorithm is proportional to  $(2m + 2n - 3k + 2)^2$ .

In order to solve the linear system (9), we need to solve five linear systems with coefficient matrices  $L_1, L_2, D_1, L_2^T, L_1^T$ , respectively. The solution of the linear systems with coefficient matrix  $L_1$  and  $L_1^T$  can be computed in  $O((m - k + 1) \times (m + 1))$  flops that is dominated by the computational complexity of multiplying  $X_k^T$  by a vector, here we assume that  $m \geq n$ . The solution of the linear system with coefficient matrix  $D_1$  is obtained by changing the sign of the corresponding entries of the right vector. Furthermore, the solution of the linear systems with

coefficient matrix  $L_2$  and  $L_2^T$  can be computed in  $O((2m + 2n - 3k + 2)^2)$  flops.

**Remark 1.** Since the optimization (6) is non-convex, the local optimization algorithm can only produce a local minimum, moreover, the stationary point may be a maxima or a point of inflection [20,25,28,31]. However, in our experiments, we suppose that the input polynomials are within a relative error of no more than  $10^{-2}$  to the polynomials with an exact GCD of given degree  $k$ . Under this assumption, we demonstrate that our fast STLS algorithm converges quickly to the minimal approximate solutions. In some difficult cases, for example, the input polynomials are far from having an approximate GCD, our fast algorithm takes significantly many iterations of search before reaching a point of convergence. This point may still only be a local minima or maxima.

$$Y = \begin{bmatrix} \tilde{u}_{m-k} & \cdots & \tilde{u}_0 & & & \\ & \ddots & & \ddots & & \\ & & & & \tilde{u}_{m-k} & \cdots & \tilde{u}_0 \end{bmatrix}$$

**4. An approximate GCD algorithm and experiments**

The following algorithm is designed for finding an approximate solution to Problem 1.1:

**Algorithm FSyl-STLS.**

*Input:* A Sylvester matrix  $S$  generated by two univariate polynomials  $f, g \in \mathbb{R}[x]$  with  $\deg(f) = m, \deg(g) = n$ , an integer  $1 \leq k \leq \min(m, n)$ , a tolerance tol.

*Output:* Polynomials  $\tilde{f}$  and  $\tilde{g}$  with  $\dim \text{Nullspace}(S(\tilde{f}, \tilde{g})) \geq k$  and the Euclidean distance  $\|\tilde{f} - f\|_2^2 + \|\tilde{g} - g\|_2^2$  is reduced to a minimum:

- (1) Form the  $k$ th Sylvester matrix  $S_k$ , choose the first column of  $S_k$  as  $\mathbf{b}_k$ , and the remaining columns of  $S_k$  as  $A_k$ . Let  $E_k = \mathbf{0}, \mathbf{h}_k = \mathbf{0}, \boldsymbol{\lambda} = \mathbf{0}$ .
- (2) Compute  $\mathbf{x}$  from  $\min \|A_k \mathbf{x} - \mathbf{b}_k\|_2$  and set  $\mathbf{r} = A_k \mathbf{x} - \mathbf{b}_k$ . Form  $X_k$  as shown in (8).
- (3) Repeat
  - (a) Solve the linear system (9) by the fast algorithm introduced in Section 3.
  - (b) Set  $\mathbf{x} = \mathbf{x} + \Delta \mathbf{x}, \mathbf{z} = \mathbf{z} + \Delta \mathbf{z}, \boldsymbol{\lambda} = \boldsymbol{\lambda} + \Delta \boldsymbol{\lambda}$ .

- (c) Construct the matrix  $E_k$  and  $\mathbf{h}_k$  from  $\mathbf{z}$ , and  $X_k$  from  $\mathbf{x}$ . Set  $A_k = A_k + E_k, \mathbf{b}_k = \mathbf{b}_k + \mathbf{h}_k, \mathbf{r} = A_k \mathbf{x} - \mathbf{b}_k$ , until  $(\|\Delta \mathbf{x}\|_2 \leq \text{tol} \text{ and } \|\Delta \mathbf{z}\|_2 \leq \text{tol} \text{ and } \|\Delta \boldsymbol{\lambda}\|_2 \leq \text{tol})$ .
- (4) Output polynomials  $\tilde{f}$  and  $\tilde{g}$  formed from  $\mathbf{b}_k$  and  $A_k$ .

Suppose  $\tilde{S}_k = [\tilde{\mathbf{b}}_k, \tilde{A}_k]$  is the nearest singular  $k$ th Sylvester matrix computed by the algorithm FSyl-STLS and  $\tilde{A}_k \mathbf{x} = \tilde{\mathbf{b}}_k$ . We form the polynomials  $\tilde{u}$  and  $\tilde{v}$  from the vector  $[-1, \mathbf{x}^T]^T$ , where  $\tilde{u}, \tilde{v}$  satisfy that

$$\tilde{v}\tilde{f} + \tilde{u}\tilde{g} = 0.$$

Let  $\tilde{u} = \tilde{u}_{m-k}x^{m-k} + \cdots + \tilde{u}_1x + \tilde{u}_0$  and  $\tilde{v} = \tilde{v}_{n-k}x^{n-k} + \cdots + \tilde{v}_1x + \tilde{v}_0$ . Suppose  $k = \deg(\gcd(\tilde{f}, \tilde{g}))$ , an approximate GCD of  $f$  and  $g$  can be formed from the vector  $\mathbf{d}$  which solves the least squares problem

$$\min_{\mathbf{d}} \|Y \mathbf{d} - \mathbf{c}\|_2, \tag{13}$$

where  $\mathbf{c} = [\tilde{f}_m, \dots, \tilde{f}_0, \tilde{g}_n, \dots, \tilde{g}_0]^T$  and

$$\begin{bmatrix} -\tilde{v}_{n-k} & \cdots & -\tilde{v}_0 & & & \\ & \ddots & & \ddots & & \\ & & & & -\tilde{v}_{n-k} & \cdots & -\tilde{v}_0 \end{bmatrix}^T \in \mathbb{R}^{(m+n+2) \times (k+1)}.$$

Since the coefficient matrix  $Y$  is a sparse Toeplitz matrix, and  $Y^T Y$  is a positive definite Toeplitz matrix with displacement rank at most 2, the linear system (13) can be solved using  $O((m + n + 2)k)$  flops [20,25,28]. When  $k$  is smaller than  $\deg(\gcd(\tilde{f}, \tilde{g}))$ , as suggested in [32], we may increase  $k$  by  $k + 1$  and run the algorithm FSyl-STLS on  $\tilde{f}, \tilde{g}$  again until we find the correct  $k$ .

Below we present an example to show that the performance of the fast algorithm derived in [24] can be affected by the large condition number of the involved least squares problem.

**Example 1.** We construct the polynomials  $f$  and  $g$  according to the following way:

$$f = ph + 10^{-7} \delta_1 \frac{\|ph\|_2}{\|\delta_1\|_2}, \quad g = qh + 10^{-7} \delta_2 \frac{\|qh\|_2}{\|\delta_2\|_2},$$

where the prime parts  $p, q$  and the common divisor  $h$  have degrees 50, 40 and 10, respectively; for noises the polynomials  $\delta_1$  and  $\delta_2$  have degrees 60 and 50, respectively, with random integer coefficients in  $[-10^7, 10^7]$ . After scaled, the noises have sizes

0.000283 and 0.000317, respectively.

$$\begin{aligned}
 f &= 25.2137x^{60} - 2.03761x^{59} - 49.5383x^{58} \\
 &\quad + \dots - 14.3085, \\
 g &= -12.3988x^{50} - 35.6049x^{49} - 49.3511x^{48} \\
 &\quad + \dots + 7.61565.
 \end{aligned}$$

Let  $\text{tol} = 10^{-5}$ ,  $k = 10$ , applying the algorithm FSyl-STLS, after three iterations, we obtain an approximate GCD of  $f, g$  of degree 10 with the minimal perturbation  $2.265 \times 10^{-8}$ . However, since the condition number of the least squares problem solved by the fast STLN algorithm in [24] is about  $3.7 \times 10^{14}$ , the minimal polynomial perturbation  $6.645 \times 10^{-7}$  is obtained by the fast STLN algorithm after 10 iterations, and it is larger than the input perturbation  $1.807 \times 10^{-7} (= 0.000283^2 + 0.000317^2)$ .

Here  $m, n$  denote the degrees of polynomials  $f$  and  $g$ ;  $k$  is the degree of the approximate GCD of  $f$  and  $g$ ;  $\text{error}(\text{STLN})$ ,  $\text{error}(F_{\text{STLN}})$  and  $\text{error}(F_{\text{STLS}})$  denote the minimal perturbation  $\|\tilde{f} - f\|_2^2 + \|\tilde{g} - g\|_2^2$  computed by the algorithms in [19,24,33] and the FSyl-STLS algorithm;  $\text{it.num}(F_{\text{STLN}})$  and  $\text{it.num}(F_{\text{STLS}})$  denote the average numbers of iterations performed by the fast STLN algorithm [24] and the FSyl-STLS algorithm respectively. As shown in the table, the two fast algorithms both converge quickly to the minimal approximate solutions, needing no more than five iterations. When the degrees of polynomials become large, we note that the FSyl-STLS algorithm still converges very quickly while the fast STLN algorithm requires more number of iterations.

Ex	$m, n$	$k$	error (STLN)	it.num ( $F_{\text{STLN}}$ )	error ( $F_{\text{STLN}}$ )	it.num ( $F_{\text{STLS}}$ )	error ( $F_{\text{STLS}}$ )
1	2, 2	1	5.59933e-3	2.20	5.59933e-3	2.25	5.59933e-3
2	3, 3	2	1.07129e-2	3.05	1.07129e-2	3.15	1.07129e-2
3	5, 4	3	1.56146e-6	1.20	1.56146e-6	1.20	1.56146e-6
4	5, 5	3	1.34138e-8	1.05	1.34318e-8	1.05	1.34318e-8
5	6, 6	4	1.96333e-10	1	1.96333e-10	1	1.96333e-10
6	8, 7	4	1.98415e-16	1	1.98415e-16	1	1.98415e-16
7	10, 10	5	1.51551e-12	2.15	1.51551e-12	1	1.51551e-12
8	14, 13	7	2.61818e-4	2	2.61819e-4	2	2.61819e-4
9	28, 28	10	3.54575e-4	2.75	3.54575e-4	2	3.54575e-4
10	50, 50	30	9.35252e-6	3.25	9.35274e-6	1.4	9.35250e-6

In the table, we compare the accuracy of the new fast algorithm with the algorithms in [19,24,33]. We use Maple 9 and take  $\text{Digits} = 15$ . The sample polynomials are generated as described in [19]: we use 50 random cases for each  $(m, n)$ , and report the average over all results. For each example, the prime parts and the GCD of two polynomials are constructed by choosing polynomials with random integer coefficients in the range  $-10 \leq c \leq 10$ , and then adding a perturbation. For noise we choose a relative tolerance  $10^{-e}$ , then randomly choose a polynomial that has the same degree as the product, with coefficients in  $[-10^e, 10^e]$ . Finally, we scale the perturbation so that the relative error is  $10^{-e}$ .

### 5. Concluding remarks

In this paper, we develop a fast STLS algorithm for computing an approximate GCD of two univariate polynomials. For this iterative algorithm, each single iteration has quadratic computational complexity in the degrees of the given polynomials. Experiments show that the new algorithm can produce satisfactory computational results and it may require less number of iterations than the fast algorithm derived in [24]. Our new algorithm can also be generalized to compute the approximate GCDs of several polynomials with linearly constrained coefficients.

## Acknowledgments

The authors wish to thank Dr. Ivan Markovsky and anonymous referees for their valuable comments and suggestions that helped improve the paper.

## References

- [1] M. Agrawal, P. Stoica, P. Åhgren, Common factor estimation and two applications in signal process., *Signal Processing* 84 (2004) 421–429.
- [2] T. Söderström, P. Stoica, B. Friedlander, An indirect prediction error method, *Automatica* 27 (1991) 183–188.
- [3] P. Stoica, T. Söderström, Common factor detection and estimation, *Automatica* 33 (1997) 985–989.
- [4] A. Schönhage, Quasi-GCD computations, *J. Complexity* 1 (1985) 118–137.
- [5] M.T. Noda, T. Sasaki, Approximate GCD and its application to ill-conditioned algebraic equations, *J. Comput. Appl. Math.* 38 (1991) 335–351.
- [6] R.M. Corless, P.M. Gianni, B.M. Trager, S.M. Watt, The singular value decomposition for polynomial systems, in: A.H.M. Levelt (Ed.), *Proceedings of the 1995 International Symposium on Symbolic and Algebraic Computation (ISSAC'95)*, ACM Press, New York, NY, 1995, pp. 96–103.
- [7] I.Z. Emiris, A. Galligo, H. Lombardi, Certified approximate univariate GCDs, *J. Pure Appl. Algebra* 117 & 118 (1996) 229–251 (Special Issue on Algorithms for Algebra).
- [8] N. Karmarkar, Y.N. Lakshman, Approximate polynomial greatest common divisors and nearest singular polynomials, in: Y.N. Lakshman (Ed.), *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation (ISSAC'96)*, ACM Press, New York, NY, 1996, pp. 35–42.
- [9] V. Hribernic, H.J. Stetter, Detection and validation of clusters of polynomials zeros, *J. Symbolic Comput.* 24 (1997) 667–681.
- [10] B. Beckermann, G. Labahn, When are two polynomials relatively prime?, *J. Symbolic Comput.* 26 (1998) 677–689.
- [11] C.J. Zarowski, X. Ma, F.W. Fairman, A QR-factorization method for computing the greatest common divisor of polynomials with real-valued coefficients, *IEEE Trans. Signal Process.* 48 (2000) 3042–3051.
- [12] V.Y. Pan, Numerical computation of a polynomial GCD and extensions, *Inform. Comput.* 167 (2001) 71–85.
- [13] M. Sasaki, T. Sasaki, Polynomial remainder sequence and approximate GCD, *ACM SIGSAM Bull.* 31 (2001) 4–10.
- [14] L. Zhi, Displacement structure in computing approximate GCD of univariate polynomials, in: Z. Li, W. Sit (Eds.), *Proceedings of the Sixth Asian Symposium on Computer Mathematics (ASCM 2003)*, Lecture Notes Series on Computing, vol. 10, World Scientific, Singapore, 2003, pp. 288–298.
- [15] R.M. Corless, S.M. Watt, L. Zhi, QR factoring to compute the GCD of univariate approximate polynomials, *IEEE Trans. Signal Process.* 52 (2004) 3394–3402.
- [16] Z. Zeng, The approximate GCD of inexact polynomials. Part I: a univariate algorithm, manuscript, 2004.
- [17] Z. Zeng, Computing multiple roots of inexact polynomials, *Math. Comput.* 74 (2005) 869–903.
- [18] B. Botting, M. Giesbrecht, J. May, Using Riemannian SVD for problems in approximate algebra, in: D. Wang, L. Zhi (Eds.), *Proceedings of the 2005 International Workshop on Symbolic–Numeric Computation*, Distributed at the Workshop in Xi'an, China, pp. 209–219.
- [19] E. Kaltofen, Z. Yang, L. Zhi, Structured low rank approximation of a Sylvester matrix, in: D. Wang, L. Zhi (Eds.), *Preliminary Version in SNC 2005 Proceedings*, in: *Symbolic-Numeric Computation, Trends in Mathematics*, Birkhäuser Verlag, Basel, Switzerland, 2007, pp. 69–83.
- [20] I. Markovsky, S.V. Huffel, An algorithm for approximate common divisor computation, in: *Proceedings of the 17th Symposium on Mathematical Theory of Networks and Systems (MTNS'06)*, Kyoto, Japan, July 2006, pp. 274–279.
- [21] J.B. Rosen, H. Park, J. Glick, Total least norm formulation and solution for structured problems, *SIAM J. Matrix Anal. Appl.* 17 (1) (1996) 110–128.
- [22] H. Park, L. Zhang, J.B. Rosen, Low rank approximation of a Hankel matrix by structured total least norm, *BIT* 39 (4) (1999) 757–779.
- [23] B. Li, Z. Yang, L. Zhi, Fast low rank approximation of a Sylvester matrix by structured total least norm, *J. Jpn Soc. Symbolic Algebraic Comput.* 11 (3,4) (2005) 165–174.
- [24] B. Li, Z. Liu, L. Zhi, Implementation of fast low rank approximation of a Sylvester matrix, in: D. Wang, L. Zhi (Eds.), *Symbolic-Numeric Computation, Trends in Mathematics*, Birkhäuser Verlag, Basel, Switzerland, 2007, pp. 85–97.
- [25] P. Lemmerling, N. Mastronardi, S. Van Huffel, Fast algorithm for solving the Hankel/Toeplitz structured total least squares problem, *Numer. Algorithms* 23 (2000) 371–392.
- [26] T. Kailath, A.H. Sayed, Displacement structure: theory and applications, *SIAM Rev.* 37 (3) (1995) 297–386.
- [27] S. Gao, E. Kaltofen, J.P. May, Z. Yang, L. Zhi, Approximate factorization of multivariate polynomials via differential equations, in: J. Gutierrez (Ed.), *Proceedings of the 2004 International Symposium Symbolic and Algebraic Computation (ISSAC'04)*, ACM Press, New York, NY, 2004, pp. 167–174.
- [28] P. Lemmerling, *Structured total least squares: analysis, algorithms and applications*, Dissertation, Katholieke Universiteit Leuven, Belgium, 1999.
- [29] G.H. Golub, C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 1996.
- [30] S. Chandrasekaran, A. Sayed, A fast stable solver for nonsymmetric Toeplitz and quasi-Toeplitz systems of linear equations, *SIAM J. Matrix Anal. Appl.* 19 (1) (1998) 107–139.
- [31] Y. Yuan, W. Sun, *Optimization Theories and Algorithms (in Chinese)*, Science Press, 1997.
- [32] E. Kaltofen, Z. Yang, L. Zhi, Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials, manuscript, December 2006, 20pp.
- [33] E. Kaltofen, Z. Yang, L. Zhi, Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials, in: J.-G. Dumas (Ed.), *Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation (ISSAC'06)*, ACM Press, New York, NY, 2006, pp. 169–176.