Symbolic-Numeric Computation
D. Wang and L. Zhi, Eds.
Trends in Mathematics, 85–97

Implementation of Fast Low Rank Approximation of a Sylvester Matrix

Bingyu Li, Zhuojun Liu and Lihong Zhi

Abstract. We describe and implement a fast algorithm for constructing structured low rank approximation of a Sylvester matrix. The fast algorithm is obtained by exploiting low displacement ranks of the involved structured matrices. We present detailed error analysis and experiments to show that the fast algorithm is stable.

Mathematics Subject Classification (2000). Primary 68W30; Secondary 65F05.

Keywords. Sylvester matrix, displacement rank, generalized Schur algorithm, structured total least norm.

1. Introduction

The authors in [11] described a fast algorithm based on structured total least norm (STLN) [16, 14] for constructing structured low rank approximation of a Sylvester matrix and obtaining the nearest perturbed polynomials with exact GCD of degree not less than a given positive integer. This algorithm is of complexity $O((2m + 2n - k + 3)^2)$, where m, n, k are degrees of input polynomials and a given positive integer. The increased efficiency is obtained by exploiting low displacement ranks of the involved structured matrices in [10, 11]. However, since coefficient matrices appeared in the STLN method have large condition numbers, it is necessary to reduce error by choosing a suitable generator matrix for the fast algorithm. In this paper, we present a new generator pair of the augmented matrix (3.6) in Sect. 3. In Sect. 4, we analyze the backward error and forward error of the fast algorithm. Experiments are given in Sect. 5 to show the stability of the fast algorithm.

The work is partially supported by the National Key Basic Research Project 2004CB318000 of China and the Chinese National Science Foundation under Grants 10371127 and 10401035.

2. Preliminaries

We are given two polynomials $a, b \in \mathbb{R}[x]$ with $a = a_m x^m + \cdots + a_1 x + a_0$ and $b = b_n x^n + \cdots + b_1 x + b_0$, $a_m \neq 0, b_n \neq 0$. S is the Sylvester matrix of a and b. The perturbations of a and b are denoted by $\Delta a = \Delta a_m x^m + \cdots + \Delta a_1 x + \Delta a_0$ and $\Delta b = \Delta b_n x^n + \cdots + \Delta b_1 x + \Delta b_0$ respectively. We consider the minimal perturbation problem: For a positive integer $k \leq \min(m, n)$, *minimize* $\|\Delta a\|_2^2 + \|\Delta b\|_2^2$ *preserving that* $a + \Delta a$ *and* $b + \Delta b$ *have an exact GCD of degree not less than* k.

Denote $S_k = [\mathbf{a}\, A_k] \in \mathbb{R}^{(m+n-k+1)\times(m+n-2k+2)}$ as the k-th Sylvester matrix,

$$S_k = \begin{bmatrix} a_m & 0 & \cdots & 0 & 0 & b_n & 0 & \cdots & 0 & 0 \\ a_{m-1} & a_m & \cdots & 0 & 0 & b_{n-1} & b_n & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & a_0 & a_1 & 0 & 0 & \cdots & b_0 & b_1 \\ 0 & 0 & \cdots & 0 & a_0 & 0 & 0 & \cdots & 0 & b_0 \end{bmatrix}, \tag{2.1}$$

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

where $\mathbf{a}$ is the first column of S_k and A_k consists of the last $m+n-2k+1$ columns of S_k.

The perturbations Δa and Δb are expressed by an $(m+n+2)$-dimensional vector $\mathbf{d}$,

$$\mathbf{d} = [d_1, d_2, \ldots, d_{m+n+1}, d_{m+n+2}]^T. \tag{2.2}$$

The k-th Sylvester structured perturbation of S_k is represented as $[\Delta\mathbf{a}\, D_k]$.

Theorem 1. [10, 11] *Given univariate polynomials* $a(x), b(x) \in \mathbb{R}[x]$ *with* $\deg(a) = m$ *and* $\deg(b) = n$. *Let* $S(a, b)$ *be the Sylvester matrix of* $a(x)$ *and* $b(x)$, S_k *be the* k*-th Sylvester matrix,* $1 \leq k \leq \min(m, n)$. *Then* $\deg(\gcd(a, b)) \geq k$ *if and only if* S_k *has rank deficiency at least* 1.

The minimal perturbation problem can be formulated as the following equality constrained least squares problem:

$$\min_{\mathbf{x},\mathbf{d}} \|\mathbf{d}\|_2, \text{ subject to } \mathbf{r} = 0, \tag{2.3}$$

where the structured residual $\mathbf{r}$ is given by

$$\mathbf{r} = \mathbf{a} + \Delta\mathbf{a} - (A_k + D_k)\mathbf{x}. \tag{2.4}$$

The STLN algorithm [1] initializes $\mathbf{x}$ as the unstructured least square solution $A_k\mathbf{x} \approx \mathbf{a}$ and sets $\Delta\mathbf{a} = \mathbf{d} = \mathbf{0}$, and then refines both $\mathbf{x}$ and $\mathbf{d}$ by the first order iterative update

$$\min_{\Delta\mathbf{x},\Delta\mathbf{d}} \left\| \begin{bmatrix} w(X_k - P_k) & w(A_k + D_k) \\ I_{m+n+2} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{d} \\ \Delta\mathbf{x} \end{bmatrix} + \begin{bmatrix} -w\mathbf{r} \\ \mathbf{d} \end{bmatrix} \right\|_2, \tag{2.5}$$

where w is a large penal value and I_{m+n+2} is an identity matrix of order $m+n+2$. The matrices P_k and X_k are introduced in [11, 10] such that

$$\Delta\mathbf{a} = P_k\, \mathbf{d},\ D_k\mathbf{x} = X_k\mathbf{d}. \tag{2.6}$$

Let us denote the coefficient matrix of the system in (2.5) by M,

$$M = \begin{bmatrix} w(X_k - P_k) & w(A_k + D_k) \\ I_{m+n+2} & \mathbf{0} \end{bmatrix}, \tag{2.7}$$

and denote $\mathbf{y} = \begin{bmatrix} \Delta\mathbf{d} \\ \Delta\mathbf{x} \end{bmatrix}$, $\mathbf{z} = \begin{bmatrix} w\mathbf{r} \\ -\mathbf{d} \end{bmatrix}$; the least squares problem (2.5) can be rewritten as

$$\min_{\mathbf{y}} \|M\mathbf{y} - \mathbf{z}\|_2 . \tag{2.8}$$

It has been shown in [11] that M is a Toeplitz-like structured matrix of displacement rank at most 4.

3. Fast Algorithm for Solving the Least Squares Problem

Fast algorithms based on QR decomposition for solving least squares problems with coefficient matrices being Toeplitz matrices have been considered in [5, 12, 4, 2, 6, 15, 17]. The stability properties of these algorithms are still not well understood and most of the algorithms may suffer from loss of accuracy when they are applied to ill-conditioned problems. Based on the method of corrected semi-normal equations, the algorithm derived in [13] can produce a more accurate R factor in the QR decomposition of a Toeplitz matrix, even for certain ill-conditioned matrices. Another fast and stable algorithm for solving the Toeplitz-like least squares problem was developed by Gu in [8]. The algorithm is based on the fast algorithm for solving Cauchy-like least squares problems. Although these algorithms [13, 8] can be used to solve least squares problems of significantly extended range from well-conditioned to certain ill-conditioned. It is still under investigation whether those algorithms can be used to solve the least squares problems (2.8) with coefficient matrices having many very small singular values. In [11], we propose to solve the least squares problem (2.8) fast by extending the fast algorithm described by Chandrasekaran et al. in [3] for solving systems of linear equations. Here, we show that their fast algorithm [3] can be generalized to solve (2.8). The numerical stability of the fast algorithm will be explained in next two sessions.

For the least squares problem (2.8), we denote its solution by $\mathbf{y}_{\mathrm{LS}}$, and the minimum residual vector by $\mathbf{r}_{LS} = M\mathbf{y}_{\mathrm{LS}} - \mathbf{z}$. Then $\mathbf{y}_{LS}$ solves the following linear system:

$$\begin{bmatrix} M^T M & M^T \\ M & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ -\mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{z} + \mathbf{r}_{LS} \end{bmatrix}. \tag{3.1}$$

Denote by Q the orthogonal matrix from the QR decomposition of M. We partition Q as: $Q = [Q_1, Q_2]$, where $Q_2 \in \mathbb{R}^{(2m+2n-k+3)\times k}$; then

$$\|\mathbf{r}_{LS}\|_2 = \|Q_2^T \mathbf{z}\|_2 = \left\| Q_2^T \begin{bmatrix} w\mathbf{r} \\ -\mathbf{d} \end{bmatrix} \right\|_2 . \tag{3.2}$$

Due to the heavy weight of the upper block $M(1..m+n-k+1,:)$ of M, the entries of the block $Q_2(m+n-k+2..2m+2n-k+3,:)$ are $O(1)$, the block $Q_2(1..m+n-k+1,:)$ consists of near zero elements. Therefore, derived from (3.2), $\|\mathbf{r}_{LS}\|_2$ is of much smaller size compared to $\|\mathbf{z}\|_2$, i.e.

$$\|\mathbf{r}_{LS}\|_2 \ll \|\mathbf{z}\|_2 . \tag{3.3}$$

The inequality (3.3) tells us that we can compute an approximate solution $\hat{\mathbf{y}}$ to (2.8) by omitting the term $\mathbf{r}_{LS}$ and solving the following augmented system proposed in [11]:

$$\begin{bmatrix} M^TM & M^T \\ M & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ -\mathbf{z} \end{bmatrix} \approx \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}. \tag{3.4}$$

We normalize the matrix M and the vector $\mathbf{z}$ as:

$$M := M/\|M\|_F, \quad \mathbf{z} := \mathbf{z}/\|M\|_F, \tag{3.5}$$

where $\|M\|_F$ is the Frobenius norm of M. Due to the large penal value w, after normalization, the lower left corner of M has very small diagonal elements. This causes the numerical rank deficiency of M^TM. Moreover, since M is not a square matrix, the coefficient matrix of the linear system (3.4) is rank deficient. In order to complete the generalized Schur algorithm successfully, we construct T [3, 11] as:

$$T = \begin{bmatrix} M^TM + \alpha I^{(1)} & M^T \\ M & -\beta I^{(2)} \end{bmatrix}, \tag{3.6}$$

where $\alpha I^{(1)}, \beta I^{(2)}$ are small multiples of identity matrices. Here the perturbed matrix $M^TM + \alpha I^{(1)}$ is positive definite, which ensures the positive steps complete successfully; The perturbation $\beta I^{(2)}$ is added to guarantee that the Schur complement of T with respect to $M^TM + \alpha I^{(1)}$ is negative definite, which ensures the negative steps complete successfully.

It has been shown in [11] that T is a structured matrix with displacement rank at most 10. We can construct a generator pair (G, J) for T such that

$$T - FTF^T = GJG^T,$$

where

$$F = \operatorname{diag}\left(Z_{m+1}, Z_{n+1}, Z_{n-k}, Z_{m-k+1}, Z_{m+n-k+1}, Z_{m+n+2}\right),$$

$J = \operatorname{diag}\left(I_4, -I_6\right)$, and G is a matrix with 10 columns. However, in [11], we expressed G by columns of T, for which some of entries are of order $1/w^2$, where w is the large penal value. It is undesirable for numerical stability.

In the following, we introduce a new generator matrix with columns which are only of order $1/w$, the same order as that of M. Define $t_1, \ldots, t_4$ as:

$$\begin{aligned} t_1 &= \|M(:,1)\|_2^2 + \alpha, & t_2 &= \|M(:,m+2)\|_2^2 + \alpha, \\ t_3 &= \|M(:,m+n+3)\|_2^2 + \alpha, & t_4 &= \|M(:,2n+m-k+3)\|_2^2 + \alpha. \end{aligned}$$

Let

$$\begin{aligned}
\mathbf{c}_1 &= \left[M^T(1,:)M,\ M^T(1,:)\right]^T + \alpha I(:,1),\\
\mathbf{c}_2 &= \left[M^T(m+2,:)M,\ M^T(m+2,:)\right]^T + \alpha I(:,m+2),\\
\mathbf{c}_3 &= \left[M^T(m+n+3,:)M,\ M^T(m+n+3,:)\right]^T + \alpha I(:,m+n+3),\\
\mathbf{c}_4 &= \left[M^T(2n+m-k+3,:)M,\ M^T(2n+m-k+3,:)\right]^T\\
&\quad +\alpha I(:,2n+m-k+3),
\end{aligned}$$

where I denotes the identity matrix of order $4m+4n-3k+6$. Then

$$\begin{aligned}
\mathbf{g}_1 &= \mathbf{c}_1/\sqrt{t_1},\\
\mathbf{g}_2 &= \mathbf{c}_2/\sqrt{t_2},\ \text{except that } \mathbf{g}_2[1]=0,\\
\mathbf{g}_3 &= \mathbf{c}_3/\sqrt{t_3},\ \text{except that } \mathbf{g}_3[1]=0,\ \mathbf{g}_3[m+2]=0,\\
\mathbf{g}_4 &= \mathbf{c}_4/\sqrt{t_4},\ \text{except that } \mathbf{g}_4[1]=0,\ \mathbf{g}_4[m+2]=0,\ \mathbf{g}_4[m+n+3]=0,\\
\mathbf{g}_5 &= \left[0, \mathbf{g}_1^T(2:4m+4n-3k+6)\right]^T,\\
\mathbf{g}_6 &= \left[\mathbf{g}_2^T(1:m+1), 0, \mathbf{g}_2^T(m+3:4m+4n-3k+6)\right]^T,\\
\mathbf{g}_7 &= \left[\mathbf{g}_3^T(1:m+n+2), 0, \mathbf{g}_3^T(m+n+4:4m+4n-3k+6)\right]^T,\\
\mathbf{g}_8 &= \left[\mathbf{g}_4^T(1:2n+m-k+2), 0, \mathbf{g}_4^T(2n+m-k+4:4m+4n-3k+6)\right]^T,\\
\mathbf{g}_9 &= [\ \underbrace{0,\ldots,0,0}_{2m+2n-2k+3}, \sqrt{\beta}, 0,\ldots,0\ ]^T,\\
\mathbf{g}_{10} &= [\ \underbrace{0,\ldots,0,0}_{3m+3n-3k+4}, \sqrt{\beta}, 0,\ldots,0\ ]^T.
\end{aligned}$$

Expanding the proof in [3] by combining with singular value decomposition (SVD) of M, we prove that after applying $2m+2n-2k+3$ positive steps and $2m+2n-k+3$ negative steps of the generalized Schur algorithm, which operates on the generator pair (G,J), we can obtain a backward stable factorization of T:

$$\begin{bmatrix} \hat{R}^T & 0 \\ \hat{Q} & \hat{D} \end{bmatrix} \begin{bmatrix} \hat{R} & \hat{Q}^T \\ 0 & -\hat{D}^T \end{bmatrix}, \tag{3.7}$$

where $\hat{R}$ is upper triangular and $\hat{D}$ is lower triangular. Furthermore, using the triangular factorization (3.7) we can solve the following augmented system

$$T \begin{bmatrix} \mathbf{y} \\ \xi \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \tag{3.8}$$

and get

$$(T+H)\begin{bmatrix}\hat{\mathbf{y}}\\ \hat{\xi}\end{bmatrix}=\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}, \tag{3.9}$$

where $\|H\|_2 = O(\epsilon)$ and ϵ is the machine precision. The solution $\begin{bmatrix}\hat{\mathbf{y}}\\ \hat{\xi}\end{bmatrix}$ is obtained through the following substitutions:

$$\begin{bmatrix}\hat{R} & \check{Q}^T\\ 0 & -\hat{D}^T\end{bmatrix}^{-1}\begin{bmatrix}\hat{R}^T & 0\\ \hat{Q} & \hat{D}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix},$$

and $\hat{\mathbf{y}}$ is computed by the expression

$$\hat{R}^{-1}\hat{Q}^T\hat{D}^{-T}\hat{D}^{-1}\mathbf{z}. \tag{3.10}$$

$\hat{\mathbf{y}}$ is regarded as an approximate solution to the least squares problem (2.8).

As mentioned in [11], the computation of $\hat{\mathbf{y}}$ is of quadratic complexity $O((2m+2n-k+3)^2)$.

4. Error Analysis for the Fast Algorithm

In [3], the authors derived a backward error bound to show that the approximate solution $\hat{\mathbf{y}}$ is a backward stable solution to the original linear system. For our case, however, it is still not clear how to prove that $\hat{\mathbf{y}}$ is a backward stable solution to the least squares problem (2.8).

In the following, by means of the formula derived in [7], we compute an alternative F-norm backward error bound $\hat{\varepsilon}(\hat{\mathbf{y}})$ that $\hat{\mathbf{y}}$ satisfies. As shown by the numerical tests in Sect. 5, the obtained solutions are backward stable. Besides, based on (3.9) we derive a relative forward error bound for the approximate solution $\hat{\mathbf{y}}$ in Sect. 4.2.

4.1. Computation of Backward Error

Let $\hat{\varepsilon}(\hat{\mathbf{y}})$ be an alternative F-norm bound on δM such that $\hat{\mathbf{y}}$ is an exact solution to the least squares problem below

$$\min_{\mathbf{y}}\|(M+\delta M)\mathbf{y}-\mathbf{z}\|_2. \tag{4.1}$$

$\hat{\varepsilon}(\hat{\mathbf{y}})$ differs from the smallest possible backward perturbation $\varepsilon(\hat{\mathbf{y}})$ derived in [18, 9] by at most a factor less than 2. Suppose $M = U\begin{bmatrix}\Sigma\\ \mathbf{0}\end{bmatrix}V^T$ is the singular value decomposition of M and $\hat{\mathbf{y}}\neq\mathbf{0}$. Let

$$\hat{\mathbf{r}}=\mathbf{z}-M\hat{\mathbf{y}}=U\begin{bmatrix}\hat{\mathbf{r}}_1\\ \hat{\mathbf{r}}_2\end{bmatrix} \tag{4.2}$$

for $\hat{\mathbf{r}}_1 \in \mathbb{R}^{(2m+2n-2k+3)\times 1}$ and $\hat{\mathbf{r}}_2 \in \mathbb{R}^{k\times 1}$. Then

$$\hat{\varepsilon}(\hat{\mathbf{y}}) = \min(\eta, \tilde{\sigma}), \tag{4.3}$$

where $\eta = \frac{\|\hat{\mathbf{r}}\|_2}{\|\hat{\mathbf{y}}\|_2}$, and

$$\tilde{\sigma} = \sqrt{\frac{\hat{\mathbf{r}}_1^T \Sigma^2 (\Sigma^2 + \eta^2 I)^{-1} \hat{\mathbf{r}}_1}{\|\hat{\mathbf{r}}_2\|_2^2/\eta^2 + \eta^2 \hat{\mathbf{r}}_1^T (\Sigma^2 + \eta^2 I)^{-2} \hat{\mathbf{r}}_1}}. \tag{4.4}$$

The detailed analysis of computations can be found in [7].

4.2. Forward Error Analysis

We derived a relative forward error bound for the approximate solution $\hat{\mathbf{y}}$ to the least squares problem (2.8). Hereafter, we use $\kappa(\cdot)$ to denote the 2-norm condition number of its argument. We define $u = m + n - k + 1, l = m + n - k + 2$. For any integer $i, 1 \le i \le u + l$, σ_i is the i-th largest singular value of M.

Lemma 2. *Denote by* $\mathbf{f}$ *a vector which satisfies the following linear system:*

$$\begin{bmatrix} M^T M + \alpha I^{(1)} & M^T \\ M & -\beta I^{(2)} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} \\ \hat{\xi} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} + \mathbf{f}; \tag{4.5}$$

we derive an upper bound for $\frac{\|\hat{\mathbf{y}} - \mathbf{y}_{LS}\|_2}{\|\mathbf{y}_{LS}\|_2}$ *which is roughly of the form:*

$$\frac{\beta + \alpha\beta\kappa^2(M)}{\|M\|_2^2 - \alpha\beta\kappa^2(M)} + \frac{\kappa(M) + \beta\kappa^2(M)}{\|M\|_2^2 - \alpha\beta\kappa^2(M)} \frac{\|\mathbf{f}\|_2}{\|\mathbf{y}_{LS}\|_2}. \tag{4.6}$$

Proof. Partition $\mathbf{f}$ as $\left[\mathbf{f}_1^T, \mathbf{f}_2^T\right]^T$, $\mathbf{f}_1 \in \mathbb{R}^{(2m+2n-2k+3)\times 1}$, $\mathbf{f}_2 \in \mathbb{R}^{(2m+2n-k+3)\times 1}$; from (4.5), we have

$$\begin{cases} \left(M^T M + \alpha I^{(1)}\right) \hat{\mathbf{y}} + M^T \hat{\xi} = \mathbf{f}_1, \\ M\hat{\mathbf{y}} - \beta\hat{\xi} = \mathbf{z} + \mathbf{f}_2. \end{cases}$$

Eliminating $\hat{\xi}$, we get

$$\beta \left(M^T M + \alpha I^{(1)}\right) \hat{\mathbf{y}} + M^T M \hat{\mathbf{y}} = M^T \mathbf{z} + M^T \mathbf{f}_2 + \beta \mathbf{f}_1.$$

Noting that

$$M^T \mathbf{z} = M^T M \mathbf{y}_{LS},$$

using elementary calculus we get

$$\left(\beta M^T M + \alpha\beta I^{(1)} + M^T M\right)(\mathbf{y}_{LS} - \hat{\mathbf{y}}) = \left(\beta M^T M + \alpha\beta I^{(1)}\right) \mathbf{y}_{LS} - M^T \mathbf{f}_2 - \beta \mathbf{f}_1,$$

and

$$\begin{aligned} \mathbf{y}_{LS} - \hat{\mathbf{y}} &= \left(\beta I^{(1)} + \alpha\beta (M^T M)^{-1} + I^{(1)}\right)^{-1} \left(\beta I^{(1)} + \alpha\beta (M^T M)^{-1}\right) \mathbf{y}_{LS} \\ &\quad - \left(\beta I^{(1)} + \alpha\beta (M^T M)^{-1} + I^{(1)}\right)^{-1} \left(M^\dagger \mathbf{f}_2 + \beta (M^T M)^{-1} \mathbf{f}_1\right), \end{aligned}$$

where $M^\dagger$ is the Moore-Penrose pseudoinverse of M. Since

$$\kappa(M) = \left\|M^\dagger\right\|_2 \|M\|_2, \quad \kappa^2(M) = \left\|(M^T M)^{-1}\right\|_2 \|M\|_2^2,$$

we derive an upper bound for $\frac{\|\mathbf{y}_{LS}-\hat{\mathbf{y}}\|_2}{\|\mathbf{y}_{LS}\|_2}$ which is of the form:

$$\begin{aligned} & \frac{\beta+\alpha\beta\left\|(M^TM)^{-1}\right\|_2}{1-\beta-\alpha\beta\left\|(M^TM)^{-1}\right\|_2}+\frac{\left\|M^{\dagger}\right\|_2+\beta\left\|(M^TM)^{-1}\right\|_2}{1-\beta-\alpha\beta\left\|(M^TM)^{-1}\right\|_2}\frac{\|\mathbf{f}\|_2}{\|\mathbf{y}_{LS}\|_2} \\ = & \frac{\beta\|M\|_2^2+\alpha\beta\kappa^2(M)}{(1-\beta)\|M\|_2^2-\alpha\beta\kappa^2(M)}+\frac{\kappa(M)\|M\|_2+\beta\kappa^2(M)}{(1-\beta)\|M\|_2^2-\alpha\beta\kappa^2(M)}\frac{\|\mathbf{f}\|_2}{\|\mathbf{y}_{LS}\|_2}. \end{aligned}$$

The final form of the inequality follows from $\|M\|_2 \leq 1$ and $\beta \ll 1$. □

Lemma 3. *Partition* $\mathbf{z}$ *as* $\left[\mathbf{z}_u^T, \mathbf{z}_{l+k}^T\right]^T$, $\mathbf{z}_u \in \mathbb{R}^{u\times 1}$, $\mathbf{z}_{l+k} \in \mathbb{R}^{(l+k)\times 1}$*; then for* T *defined as* (3.6) *we have*

$$\left\|T^{-1}\begin{bmatrix}\mathbf{0}\\ \mathbf{z}\end{bmatrix}\right\|_2 \leq \left(\frac{4}{\sigma_u}+\frac{2}{w\beta}+\frac{1}{w\sigma_{u+l}}\right)\|\mathbf{z}_u\|_2+\left(\frac{2}{\beta}+\frac{3}{\sigma_{u+l}}\right)\|\mathbf{z}_{l+k}\|_2. \quad (4.7)$$

Proof. Denote

$$T^{-1}=\begin{bmatrix}B_{11} & B_{12}\\ B_{12}^T & B_{22}\end{bmatrix},$$

where $B_{11} \in \mathbb{R}^{(u+l)\times(u+l)}$. We derive expressions for B_{11}, B_{12} and B_{22} in terms of the SVD of M.

Let $M = U\begin{bmatrix}\Sigma\\ \mathbf{0}_{k\times(u+l)}\end{bmatrix}V^T$ be the SVD of M, where U, V are orthogonal matrices and Σ is a diagonal matrix consisting of singular values of M. The diagonal matrix Σ can be written as

$$\Sigma=\begin{bmatrix}\Sigma_u & \\ & \Sigma_l\end{bmatrix},$$

where

$$\Sigma_u=\operatorname{diag}(\sigma_1,\ldots,\sigma_u),\ \Sigma_l=\operatorname{diag}(\sigma_{u+1},\ldots,\sigma_{u+l}).$$

We define diagonal matrices

$$\begin{aligned} \Lambda_u &= \left(\left[I_u+\alpha\Sigma_u^{-2}\right]^{-1}+\beta I_u\right)^{-1}, \\ \Lambda_u' &= \left(\Sigma_u+\alpha\Sigma_u^{-1}\right)^{-1}\Lambda_u, \\ \Lambda_l &= \left(\Sigma_l+\alpha\Sigma_l^{-1}\right)^{-1}\left(\left[I_l+\alpha\Sigma_l^{-2}\right]^{-1}+\beta I_l\right)^{-1}, \\ \Lambda_{l+k} &= \begin{bmatrix}\left(\left[I_l+\alpha\Sigma_l^{-2}\right]^{-1}+\beta I_l\right)^{-1} & \\ & \frac{1}{\beta}I_k\end{bmatrix}, \end{aligned}$$

where I_u, I_l, I_k denote identity matrices of dimensions u, l, k respectively. Then

$$B_{11} = V \begin{bmatrix} \beta\Sigma_u^{-1}\Lambda_u' & \\ & \beta\Sigma_l^{-1}\Lambda_l \end{bmatrix} V^T, \tag{4.8}$$

$$B_{12} = V \begin{bmatrix} \Lambda_u' & \mathbf{0}_{u\times k} \\ \Lambda_l & \mathbf{0}_{l\times k} \end{bmatrix} U^T, \tag{4.9}$$

$$B_{22} = -U \begin{bmatrix} \Lambda_u & \\ & \Lambda_{l+k} \end{bmatrix} U^T. \tag{4.10}$$

Note that the following inequalities hold:

$$\|\Lambda_u\|_2 \le 1 + 1/\sigma_u,\ \|\Lambda_u'\|_2 \le 1/\sigma_u,\ \|\Lambda_l\|_2 \le 1/\sigma_{u+l},\ \|\Lambda_{l+k}\|_2 \le 1 + 1/\beta; \tag{4.11}$$

for the last inequality the following assumption is used:

$$\alpha\beta < \sigma_{u+l}^2. \tag{4.12}$$

Meantime, for the partition of U:

$$U = \begin{bmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{bmatrix}, \text{ where } U_{11} \in \mathbb{R}^{u\times u},$$

due to the heavy weight ($O(w)$) of the submatrix $M(1..u, :)$, the following inequalities hold:

$$\|U_{12}\|_2,\ \|U_{21}\|_2 \le 1/w. \tag{4.13}$$

Finally, we derive that

$$\begin{aligned} \left\| T^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\|_2 &\le \|B_{12}\mathbf{z}\|_2 + \|B_{22}\mathbf{z}\|_2 \\ &\le \left(\frac{4}{\sigma_u} + \frac{2}{w\beta} + \frac{1}{w\sigma_{u+l}}\right)\|\mathbf{z}_u\|_2 + \left(\frac{2}{\beta} + \frac{3}{\sigma_{u+l}}\right)\|\mathbf{z}_{l+k}\|_2. \end{aligned}$$

□

Lemma 4. *Assume that H is a backward error matrix which satisfies* (3.9)*, and that $\gamma\|H\|_2 < 1$, where*

$$\gamma = \beta/\sigma_{u+l}^2 + 2/\sigma_{u+l} + 1/\beta + 1; \tag{4.14}$$

then for the vector $\mathbf{f}$ defined in Lemma 2*, we have*

$$\|\mathbf{f}\|_2 \le \frac{\|H\|_2}{1-\gamma\|H\|_2}\left[\left(\frac{4}{\sigma_u} + \frac{2}{w\beta} + \frac{1}{w\sigma_{u+l}}\right)\|\mathbf{z}_u\|_2 + \left(\frac{2}{\beta} + \frac{3}{\sigma_{u+l}}\right)\|\mathbf{z}_{l+k}\|_2\right]. \tag{4.15}$$

Proof. By the definitions of $\mathbf{f}$ and H, we get

$$\begin{aligned}\mathbf{f} &= -H \begin{bmatrix} \hat{\mathbf{y}} \\ \hat{\xi} \end{bmatrix} \\ &= -H\,(T+H)^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \\ &= -H\left(I + T^{-1}H\right)^{-1} T^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix}.\end{aligned}$$

So

$$\begin{aligned}\|\mathbf{f}\|_2 &\le \frac{\|H\|_2}{1 - \|T^{-1}H\|_2} \left\| T^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\|_2 \\ &\le \frac{\|H\|_2}{1 - \|T^{-1}\|_2 \|H\|_2} \left\| T^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{z} \end{bmatrix} \right\|_2.\end{aligned}$$

We use (4.8), (4.9), (4.10) and (4.11) to derive that

$$\begin{aligned}\left\|T^{-1}\right\|_2 &\le \|B_{11}\|_2 + \|B_{12}\|_2 + \|B_{22}\|_2 \\ &\le \beta/\sigma_{u+l}^2 + 2/\sigma_{u+l} + 1/\beta + 1.\end{aligned}$$

Based on Lemma 3 and the assumption of $\gamma\|H\|_2 < 1$, we get the final form of the inequality. □

Theorem 5. *Assume that $\hat{\mathbf{y}}$ is a vector which solves the linear system* (3.9)*; with assumptions of $\alpha\beta < \sigma_{u+l}^2$ and $\gamma\|H\|_2 < 1$, we can derive an upper bound for the relative forward error $\frac{\|\hat{\mathbf{y}} - \mathbf{y}_{LS}\|_2}{\|\mathbf{y}_{LS}\|_2}$ which is of the form:*

$$\frac{\beta + \alpha\beta\kappa^2(M)}{\|M\|_2^2 - \alpha\beta\kappa^2(M)} + \frac{\kappa(M) + \beta\kappa^2(M)}{\|M\|_2^2 - \alpha\beta\kappa^2(M)} \frac{\|H\|_2\ \delta}{(1 - \gamma\|H\|_2)\|\mathbf{y}_{LS}\|_2}, \tag{4.16}$$

where γ is defined in (4.14) *and*

$$\delta = \left(\frac{4}{\sigma_u} + \frac{2}{w\beta} + \frac{1}{w\sigma_{u+l}}\right)\|\mathbf{z}_u\|_2 + \left(\frac{2}{\beta} + \frac{3}{\sigma_{u+l}}\right)\|\mathbf{z}_{l+k}\|_2. \tag{4.17}$$

Proof. It follows immediately from Lemmas 2, 3 and 4. □

Remark 4.1. In order to complete the generalized Schur algorithm stably we take α, β that satisfy the lower bound derived in [3]. In essential, it means that α, β can be taken somewhat larger than the machine precision. Besides this, based on the error analysis in this section, α, β should be taken such that the upper bound (4.16) is as small as it could be. Hence the assumption $\alpha\beta < \sigma_{u+l}^2$ used in Theorem 5 is natural, noting that it is approximately equivalent to $\alpha\beta\kappa^2(M) < 1$, which is a necessary condition that the upper bound (4.16) is smaller than 1.

Remark 4.2. In practice, after normalization (3.5), the following inequalities hold:

$$\|\mathbf{z}_u\|_2 < 1, \quad \|\mathbf{z}_{l+k}\|_2 < 1/w. \tag{4.18}$$

Furthermore, when the given integer k (2.5) is taken as an upper bound of the degree of approximate GCD, we generally have $\delta < 1$.

5. Experiments

In Table 1, we show the performance of the fast version of the algorithm AppSylv-k [10]. The efficiency is gained by applying the fast algorithm described in Sect. 3 to solve the least squares problem (2.8) in each iteration. In the numerical tests, we compute minimal perturbations needed for two univariate polynomials having an exact GCD of degree not less than a given integer. All computations are done in Maple 10 under Windows for Digits = 15, $\alpha = \beta = 10^{-14}$.

TABLE 1. Algorithm performance on benchmarks

Ex	m,n	k	error (classic)	error (new fast)	for.err. (LS Prob.)	$\hat{\varepsilon}(\hat{\mathbf{y}})$ (LS Prob.)
1	2, 2	1	5.59933e–3	5.59933e–3	0.461e–3	0.238e–12
2	3, 3	2	1.07129e–2	1.07129e–2	0.434e–3	0.176e–13
3	5, 4	3	1.56146e–6	1.56146e–6	0.143e–2	0.143e–13
4	5, 5	3	1.34138e–8	1.34318e–8	0.664e–3	0.452e–13
5	6, 6	4	1.96333e–10	1.96333e–10	0.182e–3	0.448e–13
6	8, 7	4	1.98415e–16	1.98416e–16	0.322e–3	0.896e–14
7	10, 10	5	1.51551e–12	1.51552e–12	0.272e–2	0.598e–13
8	14, 13	7	2.61818e–4	2.61819e–4	0.163e–1	0.112e–12
9	28, 28	10	2.54575e–4	3.54600e–4	0.992e–1	0.512e–14
10	50, 50	30	9.35252e–6	9.40237e–6	0.134	0.168e–13

The sample polynomials are the same as those generated in [10]: For each example, we use 50 random cases for each (m, n), and report the average over all results. For each example, the prime parts and 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}.

In Table 1, m, n denote the degrees of polynomials a and b; k is a given integer; "error (classic)" and "error (new fast)" denote the minimal perturbations computed by the algorithms given in [10] and this report respectively; "for.err. (LS Prob.)" denotes the relative forward error of $\hat{\mathbf{y}}$ with respect to the solution given in [10]. In the last column we show backward perturbations $\hat{\varepsilon}(\hat{\mathbf{y}})$ to the least squares problem (2.8) computed according to method given in Sect. 4.1.

The small backward perturbation $\hat{\varepsilon}(\hat{\mathbf{y}})$ shown in Table 1 imply that $\hat{\mathbf{y}}$ is a stable solution to (2.8). The computed minimal polynomial perturbations have

the same magnitudes as those computed by the algorithm in [10]; Especially, the new results in the first eight examples even have several significant digits identical to that of the classic results [10]. However, from the last two examples, we can see that the accuracy of the new results could be affected by the large condition number of M.

Acknowledgment

The authors would like to thank Prof. Zhongzhi Bai for useful discussions.

References

[1] A.A. Anda and H. Park, *Fast plane with dynamic scaling*, SIAM J. Matrix Anal. Appl. **15**, pp. 162–174, 1994.

[2] A.W. Bojanczyk, R.P. Brent and F.R. de Hoog, *QR factorization of Toeplitz matrices*, Numer. Math. **49**, pp. 81–94, 1986.

[3] S. Chandrasekaran and A.H. Sayed, *A fast stable solver for nonsymmetric Toeplitz and quasi-Toeplitz systems of linear equations*, SIMAX **19**, pp. 107–139, 1998.

[4] J. Chun, T. Kailath and H. Lev-Ari, *Fast parallel algorithms for QR and triangular factorization*, SIAM J. Sci. Stat. Comput. **8**, pp. 899–913, 1987.

[5] G. Cybenko, *A general orthogonalization technique with applications to time series analysis and signal processing*, Math. Comp. **40**, pp. 323–336, 1983.

[6] G. Cybenko, *Fast Toeplitz orthogonalization using inner products*, SIAM J. Sci. Stat. Comput. **8**, pp. 734–740, 1987.

[7] M. Gu, *Backward perturbation bounds for linear least squares problems*, SIAM J. Matrix Anal. Appl. **20**, pp. 363–372, 1998.

[8] M. Gu, *New fast algorithms for structured linear least squares problems*, SIAM J. Matrix Anal. Appl. **20**, pp. 244–269, 1998.

[9] N.J. Higham, *Accuracy and stability of numerical algorithms*, Society for Industrial and Applied Mathematics, Philadelphia, 1996.

[10] E. Kaltofen, Z. Yang and L. Zhi, *Structured low rank approximation of a Sylvester matrix*. In Dongming Wang and Lihong Zhi, editors, International Workshop on Symbolic-Numeric Computation Proceedings, pp. 188–201, 2005.

[11] B. Li, Z. Yang and L. Zhi, *Fast low rank approximation of a Sylvester matrix by structured total least norm*, Journal of Japan Society for Symbolic and Algebraic Computation **11**, pp. 165–174, 2005.

[12] J.G. Nagy, *Fast inverse QR factorization for Toeplitz matrices*, SIAM J. Sci. Stat. Comput. **14**, pp. 1174–1183, 1993.

[13] H. Park and L. Eldén, *Stability analysis and fast algorithms for triangularization of Toeplitz matrices*, Numer. Math. **76**, pp. 383–402, 1997.

[14] H. Park, L. Zhang and J.B. Rosen, *Low rank approximation of a Hankel matrix by structured total least norm*, BIT **39**, pp. 757–779, 1999.

[15] S. Qiao, *Hybrid algorithm for fast Toeplitz orthogonalization*, Numer. Math. **53**, pp. 351–366, 1988.

[16] J.B. Rosen, H. Park and J. Glick, *Total least norm formulation and solution for structured problems*, SIAM J. Matrix Anal. Appl. **17**, pp. 110–128, 1996.

[17] D.R. Sweet, *Fast Toeplitz orthogonalization*, Numer. Math. **43**, pp. 1–21, 1984.

[18] B. Waldén, R. Karlson and J.-G. Sun, *Optimal backward perturbation bounds for the linear least square problem*, Numer. Lin. Alg. Appl. **2**, pp. 271–286, 1995.

Bingyu Li, Zhuojun Liu and Lihong Zhi
Key Laboratory of Mathematics Mechanization
AMSS, Chinese Academy of Sciences
Beijing 100080 China
e-mail: {liby,zliu,lzhi}@mmrc.iss.ac.cn