Newton’s Iteration for Structured Matrices

V. Y. Pan†, Y. Rami‡, X. Wang‡
†Dept. Math. & Comp. Sci., Lehman College, Cuny
Bronx, NY 10468, USA
vpan@alpha.lehman.cuny.edu
‡Ph.D. Program in Math.
Graduate School, Cuny
NY, NY 10036, USA
ryoussef.geo@yahoo.com, xwang2@gc.cuny.edu

Keywords: Structured matrices, unified superfast algorithm.

Abstract

Newton’s iteration rapidly improves a crude initial approximation to matrix inverse. Every step amounts to two matrix multiplications, which are performed very fast for structured matrices, but special techniques are required to preserve the structure as the iteration goes on. For Toeplitz-like matrices, such techniques were developed and analyzed in 1992-93 as well as the techniques for choosing an initial approximation to the inverse. The inversion of various other structured matrices can be reduced to the Toeplitz-like case by transforms proposed in [17], but it is an interesting theoretical and practical challenge to extend Newton’s iteration to other classes of structured matrices directly in a unified way. This is done in the present paper. The extension involves several novel techniques of algorithm design and analysis and enables unified superfast approximate inversion of an $n \times n$ structured matrix $M$ (within error bound $\epsilon$) by using $O(n \log(\text{cond}(M)) \log(1/\epsilon))$ flops up to a factor $\log^d n$, $d \leq 2$.

1 Introduction

Among various matrix structures, the ones of Toeplitz and Hankel types, naturally associated with linear operators of displacement (shift), are most popular and most studied [11], [13]. The powerful concepts of displacement rank and displacement (shift) are most popular and most studied [11], but it is an interesting theoretical and practical challenge to extend Newton’s iteration to other classes of structured matrices directly in a unified way. This enabled extension of any successful algorithm, for computations with structured matrices (e.g., for the solution of a structured linear system of equations) of one of the four cited types to the three other types. (See [9] and [5] on a resulting practical breakthrough.)

Alternatively, one may exploit correlation to the operators to devise effective algorithms for various computations with structured matrices in a unified way. In this approach, for a fixed linear operator $L$, the class of $n \times n$ matrices $M$ is defined such that the image matrices $L(M)$ have small rank, $r \ll n$ or $r = O(1)$, and therefore, can be expressed nonuniquely via their short generators of length $r$, of the form

$$L(M) = GH^T,$$  \hfill (1.1)

$$G = (g_1, \ldots, g_r), \ H = (h_1, \ldots, h_r).$$  \hfill (1.2)

The generator matrices $G, H$ have only $2rn$ entries versus $n^2$ of $M$. The linear operator $L$ is chosen to be readily invertible, so that an input matrix $M$ can be easily recovered from its image matrix $L(M)$ or from its short $L$-generator $G, H$. Now, operations with $L$-generators $(G, H)$ instead of matrices $M$ can be performed much faster and with using less memory space (typically, by using $O(r^e n \log^d n)$ flops for $e, d \leq 2$ and $O(n)$ words of memory versus orders of $n^2$ or $n^3$ flops and $n^2$ words of memory for general matrices). In particular, unified algorithm of [22] yielded such computational cost bounds for solving a structured linear system of equations and for various other important computations with structured matrices, covering also the computations with singular input and ones in finite fields.

In the present paper we devise unified algorithms for rapid refinement of a crude initial approximation to the inverse of a

---

*Supported by NSF Grant CCR-9732206 and PSC CUNY Award 61393-0030.
structured matrix. Our basic tool is Newton’s iteration, which is reduced to a sequence of matrix multiplications. The operation is inexpensive for structured matrices, but the structure rapidly deteriorates in the process of iteration. We develop two general techniques for preserving both the structure and the superlinear convergence. For the two resulting versions of our Unified Newton-Structured Matrix Iteration, we estimate their convergence rate and the computational cost of their performance in flops. This analysis is ultimately reduced to the estimation of the norm of the inverse operator \( L^{-1} \), and we propose three distinct techniques for obtaining such estimates. We also show homotopy techniques for choosing an initial approximation to the inverse, which enable superfast inversion of any well-conditioned structured matrix.

Our study extends the previous work on Newton’s iteration for general matrices [26], Toeplitz-like matrices [18], [20], [21], [23], and Cauchy-like matrices [27], where in particular the important ideas of representation of a structured matrix \( M \) via the orthogonal \( L \)-generator of \( M \), produced by the SVD of \( L(M) \), and of numerical truncation of such a generator were introduced. We extend these and other known ideas but also introduce some novel techniques of independent interest, particularly for the estimation of the norm of \( L^{-1} \).

## 2 Operators and operator matrices

Typically, linear operators \( L \) associated with structured matrices take the Sylvester type form \( L = \nabla_{A,B} \) or the Stein type form \( L = \Delta_{A,B} \), where

\[
\nabla_{A,B}(M) = AM - MB, \quad \Delta_{A,B}(M) = M - AMB, \tag{2.1}
\]

\( A \) and \( B \) are operator matrices. Note that

\[
\nabla_{A,B} = A\Delta_{A^{-1},B}, \quad \nabla_{A,B} = -\Delta_{A,B^{-1}}B, \tag{2.2}
\]

assuming nonsingularity of \( A \) and \( B \), respectively.

Because of correlation (2.3), we may cover both types of operators by studying one of them, \( \nabla_{A,B} \).

### Example 2.1. Structured matrices.

\( T = (t_{i-j})_{i,j=0}^{n-1} \) (Toeplitz matrices),

\( H = (h_{i+j})_{i,j=0}^{n-1} \) (Hankel matrices),

\[ V(x) = \left( x_i \right)_{i,j=0}^{n-1}, \quad x = (x_i) \) (Vandermonde matrices),

\[ C(x,y) = \left( \frac{1}{x_i-y_j} \right)_{i,j=0}^{n-1}, \quad x = (x_i), y = (y_j) \) (Cauchy matrices).

### Example 2.2. Operator matrices.

\( D(v) = \text{diag} (v_i)_{i=0}^{n-1} \) (diagonal);

\[ Z_f = (z_{i,j})_{i,j=0}^{n-1}, \quad z_{i+1,i} = 1, \quad z_{0,n-1} = f, \quad f \text{ is any scalar}, \quad z_{i,j} = 0, \text{ for all other pairs } (i,j) \) (unit \( f \)-circulant);

\[ Z_f^T \) (\( W^T \) is the transpose of \( W \)).

### Example 2.3. Operators \( \nabla_{A,B} \) associated with matrices \( M \) and \( r = \text{rank} \nabla_{A,B}(M) \).

a) \( M = T, A = Z_c, B = Z_f \) or \( A = Z_e^T, B = Z_f^T \) for any pair \((e,f)\); \( r \leq 2 \),

b) \( M = H, A = Z_c, B = Z_f^T \) or \( A = Z_e^T, B = Z_f \); \( r \leq 2 \),

c) \( M = V(x), A = D(x), B = Z_f^T \) or \( A = D^{-1}(x), x_i \neq 0, \text{ for all } i, B = Z_i; r = 1 \),

d) \( M = C(x,y), A = D(x), B = D(y), r = 1 \).

A matrix \( M \) is of Toeplitz, Hankel, Vandermonde, or Cauchy type (or Toeplitz-, Hankel-, Vandermonde-, or Cauchy-like) if \( \nabla_{A,B}(M) \) is small relatively to the matrix size where \( \nabla_{A,B}(M) \) is an operator of parts a), b), c), d) of Example 2.3, respectively. A linear operator \( L \) is nonsingular if \( L(M) = 0 \) implies \( M = 0 \). Operators of Example 2.3 are nonsingular if \( e \neq f \) and if all \( 2n \) components of the vectors \( x \) and \( y \) are distinct.

## 3 Short generators for structured matrices

### Definition 3.1. A pair \((G,H)\) is an \( L \)-generator of length \( r \) for \( M \), as long as \((1.1), (1.2)\) hold (for some \( r \geq \text{rank } L(M) \)) and the operator \( L \) is nonsingular. The SVD of \( W = L(M) \) defines a unique orthogonal \( L \)-generator \((G,H)\) of the minimum length \( r \) for a matrix \( M \):

\[
W = U\Sigma^2V^T, \quad U^T U = V^T V = I_r, \tag{3.1}
\]

\[ \Sigma = \text{diag}(\sigma_i)_{i=1}^{r}, \quad r = \text{rank } W, \tag{3.2}
\]

\[ \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0, \tag{3.3}
\]

\[ G = U\Sigma, \quad H = V\Sigma. \tag{3.4}
\]

### Remark 3.1. Various other polices of the diagonal scaling of \( G \) and \( H \) may replace (3.4).

Such \( L \)-generators can be defined numerically, if we truncate (that is, set to zero) those \( \sigma_i \) that are smaller than a fixed tolerance \( \epsilon \).

### Theorem 3.1. [20] (cf. also [19, Proposition A.6]). Given a generator of length \( r \) for an \( n \times n \) matrix \( W \), where \( r_- = \text{rank } W \) is unknown, \( r_- \leq r \), it suffices to use \( O(r^2n + (r_- \log r_-) \log(\sigma_r^2/\delta)) \) flops to approximate within \( \delta \) all the entries of the matrices \( U, \Sigma^2, \) and \( V \) of (3.1), (3.2).

If we have a short \( L \)-generator, we may shorten it to the minimum length by applying Theorem 3.1 for \( \delta < \sigma_r^2 \) or obtain even a shorter \( L \)-generator for a nearby matrix by truncation of the smallest singular values in the SVD (up to a fixed \( \delta \)) [20].

## 4 Bilinear representation via \( L \)-operators

For nonsingular operators \( L \), we may recover \( M \) from its image \( L(M) \). Moreover, the customary operators are bilinearly invertible. That is, a matrix \( M \) is expressed bilinearly via the entries of its \( L \)-generator matrices \( G, H \) of (1.1), (1.2). The following are some simple and useful results (cf. [7], [29], [8], [14], [25]).
Theorem 4.1.

\[ M = A^kMB^k + \sum_{i=0}^{k-1} A^i \Delta_{A,B}(M) B^i, \forall k \geq 1. \]

Theorem 4.2. If the operator matrices A and B can be diagonalized by some similarity transformations, then

\[ M = P \left( \frac{(P^{-1}\nabla(M)Q)_{i,j}}{\alpha_i - \beta_j} \right) Q^{-1}, \text{ if } \alpha_i \neq \beta_j, \forall i, j; \]

\[ M = P \left( \frac{(P^{-1}\Delta(M)Q)_{i,j}}{1 - \alpha_i\beta_j} \right) Q^{-1}, \text{ if } \alpha_i\beta_j \neq 1, \forall i, j. \]

Theorem 4.3. Suppose we know the annihilation polynomials of A and B, that is, we have \( a_0 I + a_1 A + \cdots + a_{N-1} A^{N-1} + A^N = 0 \), and \( b_0 I + b_1 B + \cdots + b_{N-1} B^{N-1} + B^N = 0 \). Then \( MK = W \), where

\[
K = \sum_{j=1}^{N-1} \sum_{k=0}^{N-1-j} a_{k+j} b_k A^j + \sum_{j=1}^{N-1} \sum_{k=0}^{N-1-j} a_k b_{k+j} B^j + \sum_{k=0}^{N-1} a_k b_k - 1 \]

\[
W = \sum_{i=0}^{N-2} \sum_{j=1}^{N-1-i} \sum_{k=j}^{N-1-j} a_{k+j} b_k A^{i+j} \Delta_{A,B}(M) B^i + \sum_{i=0}^{N-2} \sum_{j=1}^{N-1-i} \sum_{k=j}^{N-1-j} a_k b_{k+j} A^{i+j} \Delta_{A,B}(M) B^i + \sum_{i=0}^{N-2} \sum_{k=i+1}^{N-1} a_k b_k A^i \Delta_{A,B}(M) B^i - \sum_{i=0}^{N-1} A^i \Delta_{A,B}(M) B^i.
\]

In particular, if \( A^p = a_0 I \) or \( B^q = b_0 I \) (that is, if A is an \( a_0 \) idempotent matrix of order \( p \) or B is an \( b_0 \) idempotent matrix of order \( q \)), then

\[ M = \sum_{i=0}^{p-1} A^i \Delta_{A,B}(M) B^i \]

or

\[ M = (I - b_0 A^q)^{-1} \sum_{i=0}^{q-1} A^i \Delta_{A,B}(M) B^i, \]

respectively.

Application of these results and (2.3) gives us simple bilinear expressions for structured matrices M via their L-generators for \( L = \nabla_{A,B} \) and \( L = \Delta_{A,B} \).

Example 4.1. For \( L = \nabla_{A,B} \), A, B of Example 2.3, and G, H of (1.1), (1.2), we have

\[ M = \frac{f}{e - f} \sum_{k=1}^{l} Z_c(g_k) Z_{1/f}^T(Z_f^T h_k) \]

for \( (A, B) = (Z_c, Z_f) \),

\[ M = \frac{f}{e} \sum_{k=1}^{l} Z_c(J g_k) Z_{1/f}^T(Z_f^T h_k) \]

for \( (A, B) = (Z_c^T, Z_f) \).

\[
M = \operatorname{diag} \left( \frac{f}{a_i^q - f} \right) \sum_{k=1}^{n-1} \sum_{i=0}^{l} D(g_k) V(x) Z_{1/f}^T(Z_f^T h_k)
\]

for \( (A, B) = (D(x), Z_f) \),

\[
M = \sum_{k=1}^{l} D(g_k) C(x, y) D(h_k)
\]

for \( (A, B) = (D(x), D(y)) \), where

\[ Z_u(v) = \sum_{i=0}^{n-1} v_i Z_u \text{ for } v = (v_i)_{i=1}^{n-1}. \]

Example 4.2. For \( L = \Delta_{A,B} \) and G, H, \( Z_u(v) \) as above, we have

\[
M = \frac{1}{1 - ef} \sum_{k=1}^{l} Z_c(g_k) Z_{1/f}^T(h_k)
\]

for \( (A, B) = (Z_c, Z_f) \),

\[
M = \frac{1}{1 - ef} \sum_{k=1}^{l} Z_c(g_k) J Z_f(J h_k)
\]

for \( (A, B) = (Z_c, Z_f^T) \),

\[
M = \frac{1}{1 - x_i y_j} \sum_{k=1}^{l} D(g_k) V(x) Z_{1/f}^T(h_k)
\]

for \( (A, B) = (D(x), Z_f) \),

\[
M = \sum_{k=1}^{l} D(g_k) \left[ \frac{1}{1 - x_i y_j} \right] D(h_k)
\]

for \( (A, B) = (D(x), D(y)) \).

There are similar expressions for structured matrices associated with other customary linear operators.

5 Operations with structured matrices in terms of their short L-generators

The presented results enables COMPRESS of a structured matrix M into its short L-generator \((G, H)\) and DECOMPRESS or RECOVERY of M from \((G, H)\). It is faster and memory space efficient to operate with short L-generators \((G, H)\) rather than with matrices M themselves, according to the informal rule: COMPRESS the input, OPERATE, RECOVER the output. For the OPERATE stage, let us extend matrix structure to the sums, products and inverses. We assume \( L = \nabla_{A,B} \) (see (2.3) or [25], [4] on some extensions to the case of \( L = \Delta_{A,B} \)).
Fact 5.1. For any linear operator $L$, we have $L(aM + bN) = aL(M) + bL(N)$.

Fact 5.2.

$\nabla_{A,C}(MN) = \nabla_{A,B}(M)N + M\nabla_{B,C}(N)$.

Fact 5.3.

$\nabla_{B,A}(M^{-1}) = -M\nabla_{A,B}(M)M^{-1}$.

Definition 5.1. $v(M), m(M,M_1)$, and $i(M)$ denote the minimum number of flops required for the multiplication of a matrix $M$ by a vector, by a matrix $M_1$, and the inversion of $M$ (assuming its nonsingularity), respectively. We write $v_{r,n}(L) = \max_{M} v(M), m_{r,r_1,n}(L, L_1) = \max_{M, M_1} m(M, M_1)$, and $i_{r,n}(L) = \max_{M} i(M)$ provided that

a) the maximization is over all $n \times n$ matrices $M$ and $M_1$ represented with their $L$-generators of length at most $r$ and $L_1$-generators of length at most $r_1$, respectively, and

b) the output matrices $M^{-1}$ and $M_1^{-1}$ are represented with their $L_-$ and $L_+$-generators of length $r + r_1$, respectively, where $L = \nabla_{A,B}$, $L_+ = \nabla_{B,A}, L_1 = \nabla_{B,C}, L_- = \nabla_{A,C}$.

For the structured matrices of Example 2.1, we have $v(M) = O(n \log n)$ for $M = T$ and $M = H$; $v(M) = O(n \log^2 n)$ for $M = V(x)$ and $M = C(x, y)$.

Based on (4.1)-(4.4), we extend these bounds to more general classes of structured matrices.

Definition 5.2. A linear operator $L$ is regularly compressible if $m_{r,n}(L) = O(v_{r,n}(L))$.

Theorem 5.1. $v_{r,n}(L) = O(rn \log n)$ for the operators $L = \nabla_{Z_1}z_1, L = \nabla_{Z_2}z_1, L = \nabla_{Z_1}z_2$, and $L = \nabla_{Z_2}z_2$; $v_{r,n}(L) = O(rn \log^2 n)$ for the operators $L = \nabla_{\Delta}(x), L = \nabla_{\Delta^{-1}}(x), L = \nabla_{\Delta}(x), L = \nabla_{\Delta^{-1}}(x,y)$. All of these operators are regularly compressible.

6 Some definitions

Definition 6.1. $\|M\|$ is any fixed operator norm of a matrix $M$. $\|M\|_l$ is the $l$-norm of $M$, $l = 1, 2, \ldots, \infty$. $\kappa(M) = \text{cond}_2(M) = \sigma_2^2(M)/\sigma_1^2(M)$ where $\sigma_1^2(M)$ is the $i$-th singular value of $M, r = \text{rank } M$.

Theorem 6.1. (Cf. [3], [6]) We have $\|M\|_2 = \sigma_2^2(M)$ for every matrix $M$ and $\kappa(M) = \|M\|_2/\|M\|_1^2$ for an $n \times n$ nonsingular matrix $M = (m_{i,j})$. Furthermore, we have $\|M\|_l/\sqrt{n} \leq \|M\|_2 \leq \|M\|_1 \sqrt{n}, l = 1, \infty; \|M\|_1 = \|MT\|_\infty = \max_j \sum_i |m_{i,j}|$. Moreover, if $M \in \mathbb{R}^{n \times n}$, that is, if the matrix $M$ has real entries, then $\|M\|_2 \leq \|M\|_1 \|M\|_\infty$.

Theorem 6.2. (Cf. [6].) Define $\sigma_i$ by (3.1) - (3.3) for $i \leq r$. Write $\sigma_q = 0$ for $q > r$. Then $\sigma_q^{2,1+} = \min_{\text{rank } V \leq q} \|W - V\|_2$.

Definition 6.2.

\[ \nu = \nu_r(L) = \sup_M (\|L(M)\|_1/\|M\|_1), \]  
(6.1)

\[ \nu^* = \nu_r(L^{-1}) = \sup_M (\|L(M)\|_1/\|L(M)\|_1), \]  
(6.2)

where the supremum is over all matrices $M$ having positive $L$-rank at most $r$.

7 Newton’s iteration for matrix inversion

Newton’s iteration rapidly improves a crude initial approximation $X_0$ to the inverse of a nonsingular matrix $M$:

\[ X_{i+1} = X_i(2I - MX_i), \quad i = 0, 1, \ldots \]  
(7.1)

Let us write

\[ E_i = M^{-1} - X_i, \quad e_i = \|E_i\|, \quad e_{1,i} = \|E_i\|_1, \]  
(7.2)

\[ R_i = M E_i = I - MX_i, \quad \rho_i = \|R_i\|, \quad \rho_{1,i} = \|R_i\|_1, \]  
(7.3)

for all $i$ and $l = 1, 2, \ldots, \infty$. (7.1) implies that

\[ R_i = R_{i-1}^2 - R_0^2, \quad \rho_i \leq \rho_0^2, \quad \rho_i, \rho_l \leq \rho_{l,0}^2, \]  
(7.4)

\[ e_i = R_0^2 \kappa(M), \quad e_{i,l} = \rho_{i,0}^2 \kappa(M). \]  
(7.5)

(7.4), (7.5) show quadratic convergence of $X_i$ to $M^{-1}$ provided that $\rho_0 < 1$ or $\rho_{l,0} < 1$.

8 Newton-Structured Matrix Iteration

For a general input matrix $M$, the computational cost of the iteration (7.1) as well as its known modifications is high because matrix products must be computed at each step. For structured matrices, matrix multiplication has a lower computational cost of $m_{r,n}(L)$ (see Theorem 5.1) when we operate with short $L$-generators. The length of the $L$-generators, however, is tripled in each step (7.1) (cf. Proposition 8.1 below). Thus, we modify Newton’s iteration where $M$ and $X_0$ are structured matrices.


Input: A positive integer $r$, two matrices $A$ and $B$, an $n \times n$ nonsingular structured matrix $M$ having $\nabla_{A,B}$-rank $r$ and defined by its $\nabla_{A,B}$-generator $(G, H)$ of length $r$, a sufficiently close initial approximation $X_0$ to $M^{-1}$ given with its $\nabla_{B,A}$-generator of length at most $r$, a bound $q$ on the number of iteration steps, and a Subroutine $\text{R}$ for the transition from a generator of length at most $3r$ for an $n \times n$ matrix approximating $M^{-1}$ to a $\nabla_{B,A}$-generator of length at most $r$ for another approximation of $M^{-1}$.

Output: A $\nabla_{B,A}$-generator of length at most $r$ for a matrix $Y_q$ approximating $M^{-1}$.

Computations: Recursively compute $\nabla_{B,A}$-generators of length $3r$ and $r$ for the matrices $X_1, Y_1, X_2, Y_2, \ldots, Y_q$, that is, of length at most $3r$ for the matrices

\[ X_{i+1} = Y_i(2I - MY_i), \quad i = 0, 1, \ldots, q - 1, \]  
(8.1)
and at most $r$ for the matrices $Y_i$ defined by Subroutine R applied to $X_i$.

**Proposition 8.1.** Under the assumptions of Algorithm 8.1, a $\nabla_{B,A}$-generator of length at most $3r$ for the matrix $X_{i+1}$ of (8.1) for any $i$ can be computed by using $O(m_{r,n}(\nabla_{B,A}))$ flops.

**Proof.** See [25].

**Remark 8.1.** We will present and analyse the modified iteration for nonsingular operators of Sylvester type. The extensions to the Stein type operators based on equations (2.3) and to certain singular operators are shown in [23] and [25].

9 Short $L$-generators by truncation of the smallest singular values

To complete the description of Algorithm 8.1, we must specify Subroutine R for the transition from $X_i$ to $Y_i$. We will propose two choices, which we denote by Subroutines R1 and R2.

**Subroutine R1.** Fix a small positive $\epsilon$, approximate within $\epsilon$ the orthogonal $L$-generator of $L(X_i)$ and decrease its length to $r$ by zeroing all the singular values $\sigma_i^2$ for $i \geq r + 1$. Output the resulting orthogonal $L$-generator $L(Y_i)$, which defines a unique matrix $Y_i$ if $L$ is a nonsingular operator.

By combining Theorems 3.1 and 6.1 and Proposition 8.1, we obtain

**Theorem 9.1.** Under the assumptions of Algorithm 8.1, it is sufficient to use $$O(m_{r,n}(\nabla_{B,A}) + r^2n + (r \log r) \log \log(\|L(M)\|_2/\epsilon))$$ flops in a single step (8.1) of this algorithm combined with Subroutine R1. If

$$\log(\|L(M)\|_2/\epsilon) = 2^{O(nr/\log r)},$$

then the cost bound turns into $O(m_{r,n}(\nabla_{B,A}) + r^2n)$.

By Theorem 6.2, $L(Y_i)$ is as close to $L(X_i)$ as possible assuming the 2-norm and the bound $L(Y_i) \leq r$. It follows that $Y_i \approx X_i$. Therefore, $Y_i \approx M^{-1}$ because $X_i \approx M^{-1}$. Formally, we deduce (cf. [18], [20], [25]):

$$\|M^{-1} - Y_i\|_2 \leq (1 + (\|A\|_2 + \|B\|_2)\nu^-)e_{2,i}$$

for $\nu^- = \nu_{2,r}(\nabla_{B,A}^{-1})$. By analyzing and summarizing our estimates, we obtain

**Corollary 9.1.** Under the assumptions of Algorithm 8.1, let

$$(1 + (\|A\|_2 + \|B\|_2)\nu^-)\rho_{2,0}^{1/\theta} \kappa(M) \leq 1 \quad (9.2)$$

for $\rho_{2,0}$ of (7.3), $\nu^- = \nu_{2,r}(\nabla_{B,A})$ of Definition 6.2, $\kappa(M)$ of Definition 6.1, and $0 < \theta \leq 1$. Let the algorithm be combined with Subroutine R1. Then

$$\|M^{-1} - Y_i\|_2 \leq \rho_{2,0}^{1+\theta}\|M^{-1}\|_2, \quad i = 1, 2, \ldots,$$

and therefore, the residual norm bound $\rho_{2,1} \leq \epsilon \kappa(M)$ is ensured in $q = \lfloor \log_{1+\theta}(\log \epsilon/\log \rho_{2,0}) \rfloor$ steps of the algorithm; under (9.1) these steps use $O(m_{r,n}(\nabla_{B,A}) + r^2n)q)$ flops, which is $O(nr^2\log^4 n)$ if the operators $\nabla_{A,B}$ and $\nabla_{B,A}$ are regularly compressible and if $\nu_{r,n}(L) = O(nr\log^4 n)$ for $L = \nabla_{A,B}$ and $L = \nabla_{B,A}$.

The estimates of Corollary 9.1 are obtained provided that the initial residual norm $\rho_{2,0}$ is bounded in terms of the norm $\nu^- = \nu_{2,r}(\nabla_{A,B}^{-1})$ (cf. (9.2)) and similarly with an alternative Subroutine R2 in the next section (cf. (10.2)). The missing estimates for the norm $\nu^-$ will be supplied in sections 11-13 and policy for choosing the initial matrix $X_0$ in section 14.

**Remark 9.1.** One may develop a dynamic version of Newton-Structured Matrix Iteration with Subroutine R1, where one may change with $i$ the level of the cut-off in zeroing the smallest singular values, to optimize the overall flops count (cf. [18] and [1] in the Toeplitz-like case). In particular, the recipe of [1] is to zero all the singular values of $\nabla_{B,A}(X_i)$ that are less than a fixed tolerance $\epsilon$. Because the residual norms $\rho_i \leq \rho_0$ are assumed to be small for all $i$, this recipe should normally lead to exactly the same matrix $Y_i$ as the one output by our Algorithm 8.1. The only exception is the matrices $X_i$ for which there exist approximation matrices $V$ satisfying simultaneously the two following inequalities: rank $\nabla_{B,A}(V) < r, \|\nabla_{B,A}(V - X_i)\|_2 \leq \epsilon$.

10 Short $L$-generators of approximate inverse by substitution

To obtain an alternative Subroutine R, to be denoted R2, write

$$G_i = -X_iG, \quad H_i^T = H_i^TX_i,$$

where $\nabla_{A,B}(M) = GH^T$, and observe that $X_i \approx M^{-1}$ and, therefore,

$$G_iH_i^T \approx M^{-1}GH^TM^{-1} = \nabla_{B,A}(M^{-1}).$$

**Subroutine R2.** Compute in $O(r\nu_{r,n}(\nabla_{B,A}))$ flops and output $\nabla_{B,A}$-generator $G_i, H_i$ of (10.1) of length $r$ for a matrix $X_i$ satisfying $\nabla_{B,A}(Y_i) = G_iH_i^T$.

**Theorem 10.1.** Write $C_i = \nu^- ||GH^T||(e_{i+2}||M^{-1}||)$, for $\nu_i$ of (7.2) and $\nu^-$ of Definition 6.2, write $C_{i} = ||Y_i - M^{-1}||$. Then $\epsilon_i \leq C_{i}e_{i}, e_{i} \leq \sqrt{C_{i+1}}\|Y_i\|, i = 1, 2, \ldots$.

The theorem is proved in [25] by using in particular the bound $e_{i+1} \leq ||M||e_i^2$, which follows because, by (7.1), we have $M^{-1} - X_{i+1} = (M^{-1} - Y_i)M(M^{-1} - Y_i)$.

Based on Theorem 10.1, we estimate the convergence rate and the computational cost of Algorithm 8.1, which uses the above subroutine (see the proofs in [25]).

**Theorem 10.2.** Let $\rho_0 \leq 1, e_i \leq ||M^{-1}||$, for all $i$,

$$\hat{e} = \rho_0\|Y_i\|(1 - \rho_0) \leq 1/\sqrt{||M||},$$

$$\hat{e}^{1-\theta}||M||^{1-\theta/2}C \leq 1,$$
for $0 < \theta \leq 1$ and $C = 3\nu^-||GHT|| \cdot ||Y_0||/(1 - \rho_0)$. Then
$c_{i+1} \leq (e^2||M||)^{(1+\theta)}, c_{i+1} \leq Cc_{i+1}, i = 0, 1, \ldots$

**Corollary 10.1.** Given positive $\epsilon$ and $\bar{\epsilon}$, write

$$k = \lfloor \log(\log(\epsilon/\log(e^2||M||))/\log(1 + \theta)) \rfloor,$$

$$l = \lfloor \log(\log(\bar{\epsilon}/\log(e^2||M||))/\log(1 + \theta)) \rfloor.$$

Then, under the assumptions of Theorem 10.2, we have $c_{k+1} \leq c_{l+1} \leq c_{i+1}$. For regularly compressible operators $\nabla_{A,B}$ and $\nabla_{B,A}$, the latter bounds are reached in $O(kv_{r,n}(\nabla_{B,A}))$ and $O(lv_{r,n}(\nabla_{B,A}))$ flops, respectively, which turn into $O(kv^2n \log^2 n)$ and $O(lv^2n \log^2 n)$ if $v_{r,n}(\nabla_{B,A}) = O(rv \log^2 n)$.

11 Bounds on the inverse operator norms by using orthogonal generators

Given an operator $L$ and a bilinear expression of a matrix $M$ via the orthogonal generator for $L(M)$, we may relate $||M||$ to the singular values of $L(M)$ and deduce the desired bound on $\nu^-$. This leads us to

**Theorem 11.1.** Let $s$ and $t$ be a pair of vectors of dimension $n$ filled with $2n$ distinct coordinates, $s_i$ and $t_j$, none of the $t_j$ being zero. Let $\nabla = \nabla_{A,B}$ and $\Delta = \Delta_{A,B}$ be nonsingular operators of (2.1), (2.2). Then we have the following bounds on the $l$-norm $\nu_{r,l}$, $l = 1, 2, 1 \leq r \leq n$, of the inverse operators $\nabla^(-1)$ and $\Delta^(-1)$ over the $n \times n$ complex matrices:

$$\nu_{r,l}(\Delta_{Z,Z^T}) = \nu_{r,l}(\Delta_{Z^T,Z}) \leq r\nu^{1.5}, \quad (11.1)$$

$$\nu_{r,l}(\nabla_{D,1}^{-1}(t),Z) \leq r\sqrt{n} ||D(t)V^T(t)||_l, \quad (11.2)$$

$$\nu_{r,l}(\nabla_{D,1}^{-1}(t),Z) \leq r\sqrt{n} ||D(t)V^T(t)||_l, \quad (11.3)$$

$$\nu_{r,l}(\nabla_{D,1}(s)) \leq r\sqrt{n} ||D(s)||C(s,t)||_l, \quad (11.4)$$

for $l = 1, 2, \infty$. For $l = 2$ over the real matrices these upper bounds are decreased by factor $\sqrt{n}$.

**Proof.** See [25].

**Remark 11.1.** The operators $\Delta$ and $\nabla$ above are associated with four matrix classes of Example 2.1.

12 Bounds on the inverse operator norm in the case of nilpotent and $f$-idempotent operator matrices

For the Stein type operators $\Delta_{A,B}$ where at least one of the operator matrices $C = A$ or $C = B$ is $f$-idempotent of rank $k$ for some positive integer $k$ and scalar $f$, that is, $C^k = fI$ (and $0$-idempotent is called nilpotent), the inverse operator norm can be bounded based on the following corollary of Theorem 4.3:

**Corollary 12.1.** Let $\Delta = \Delta_{A,B}$ be a nonsingular operator of Stein type (2.2), where $A^k = fI$ (or $B^k = fI$) for some scalar $f$ and positive integer $k$. Then (by Theorem 4.3) we have

$$\nu_{r,1} \leq ||(I - fB^k)^{-1}|| \cdot \sum_{i=0}^{k-1} ||A^i||B^i||$$

(or $\nu_{r,1} \leq ||(I - fA^k)^{-1}|| \cdot \sum_{i=0}^{k-1} ||A^i||B^i||$).

Corollary 12.1 enables us to estimate the inverse operator norm $\nu_{r,1}(\Delta_{A,B}^{-1})$ where $A \in \{Z_f, Z^T_f\}, B \in \{Z_f, Z^T_f, D(t)\}, A = D(t), B \in \{Z_f, Z^T_f\}$ for some scalar $f$ and vector $t$ (see [25]). This covers the operators associated with Chebyshev-Vandermonde-like matrices [12] and also enables us to improve our estimates of Theorem 11.1 for operators associated with Toeplitz-like and Hankel-like matrices: $\nu_{r,1}(\Delta_{A,B}^{-1}) = \nu_{r,1}(\Delta_{Z_f,Z^T_f}^{-1}) \leq n/2, \nu_{r,1}(\Delta_{Z,Z^T}^{-1}) \leq n/2, \nu_{r,1}(\Delta_{Z,Z^T}^{-1}) \leq n, \forall r \geq 1$.

13 Estimating the inverse operator norms based on the eigendecomposition of an operator matrix

In this section we will generalize our technique of section 12.

**Theorem 13.1.** Let $\Delta = \Delta_{A,B}$ be a nonsingular operator of the Stein type with $n \times n$ operator matrices $A$ and $B$. Let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of the matrix $A$. Let us write $A_{\lambda_i} = I, A_{\lambda_i} = A - \lambda_i I, B_{\lambda_i} = I - \lambda_i B, i = 1, \ldots, n$. Assume that $1/\lambda_i$ is not an eigenvalue of the matrix $B$. Then we have

$$M = \sum_{k=1}^{n} \prod_{i=1}^{k-1} A_{\lambda_i} \Delta(M) \left( \prod_{i=1}^{k} B_{\lambda_i}^{-1} \right) B_k^{-1},$$

and consequently,

$$\nu_{r,1} \leq \sum_{k=1}^{n} ||B_k^{-1}|| \left( \prod_{i=1}^{k-1} ||A_{\lambda_i}|| \right) \left( \prod_{j=1}^{k} ||B_{\lambda_j}|| \right).$$

**Proof.** See [25].

**Remark 11.1.** Theorem 13.1 enables us to estimate the norm of the inverse operators associated with Cauchy-like and Toeplitz+Hankel-like matrices [25].

14 Choices of an initial approximate inverse

Each of bounds (9.2) and (10.2) ensures rapid convergence of Algorithm 8.1. In many cases, a close initial approximation $X_0$ satisfying (9.2) and/or (10.2) is available, for instance, supplied by the preconditioned conjugate gradient method, which has linear convergence, or by a direct method performed with rounding errors. Otherwise, to fulfill (9.2) and/or (10.2), we may apply the homotopy techniques proposed and specified in some detail for Toeplitz-like matrices.
in [18]. The idea is to start with an easily invertible structured matrix $M_0$ from the same class as $M$ (such as the identity matrix $M_0 = I$ or, more generally, $M_0 = Z_k^f$ for $k \in \{0, 1, \ldots, n - 1\}$ and a scalar $f$, for a Toeplitz-like $M$, a Vandermonde matrix $M_0 = V(\kappa)$ for a Vandermonde-like $M$, and a Cauchy matrix $M_0 = C(x, y)$ for a Cauchy-like $M$), then apply Algorithm 8.1 first to approximate $M_0^{-1}$, and then, recursively, to invert numerically the matrices $M_k = M_0 + (M - M_0)\tau_k$, $k = 1, 2, \ldots, K$, by choosing $X_0M_{k-1}^{-1}$ for an initial approximation for the iterative inversion of $M_k$.

$\tau_1, \ldots, \tau_K$ form an increasing sequence of positive values where $\tau_1 = 1, M = M_k$, and $K$ is minimized provided that each $\tau_i$ is chosen close enough to $\tau_{i-1}$ to satisfy the assumptions (9.2) or (10.2) for the input matrices $M = M_i$ and $X_0 = M_{i-1}^{-1}$. A sample policy for such a choice for Toeplitz-like matrices $M$ is specified in [18], which results in the solution in $O((K + \log \log(1/\epsilon))r\nu_r,n(L))$ flops with $K = O(\log \kappa(M))$ and with the output error bound $\epsilon$.

References


