2004

TR-2004014: A Homotopic/Factorization Process for Toeplitz-like Matrices with Newton's/Conjugate Gradient Stages

Victor Y. Pan

A Homotopic/Factorization Process for Toeplitz-like Matrices with Newton's/Conjugate Gradient Stages *

Victor Y. Pan
Department of Mathematics and Computer Science
Lehman College of CUNY, Bronx, NY 10468, USA
vpan@lehman.cuny.edu

September 13, 2004

Summary We modify our earlier homotopic process for iterative inversion of Toeplitz and Toeplitz-like matrices to improve the choice of the initial approximate inverses at every homotopic step. This enables us to control the condition of the auxiliary matrices and to accelerate convergence substantially. The algorithm extends our older approach where the input matrix was factorized into the product of better conditioned factors, which are more readily invertible with the conjugate gradient algorithm. Now we study a similar factorization as a homotopic process and use the option of inverting all or some factors by applying Newton's iteration where initial approximate inverses are readily available.

2000 Math. Subject Classification: 65F10, 65B99

Key Words: Toeplitz matrices, matrix inversion, Newton's iteration, homotopic (continuation) processes

*This research was supported by NSF Grant CCR 9732206 and PSC CUNY Awards 65393-0034 and 66437-0035.

0 Introduction

Homotopic or continuation techniques are a customary means for supplying initial approximations for iterative processes. In [P92], [P01, Section 6.9], [P01a], and [PKRK04], these techniques were studied for iterative inversion of general and structured matrices M. The algorithms in these papers rely on Newton's iteration and have superlinear local convergence. For a structured input matrix M, it is usually desired to obtain a closer initial approximation to its inverse to counter the potentially negative effect on convergence caused by the compression of the approximate inverses updated in the iterative process. For Toeplitz input matrices M and a certain class of initial approximate inverses X_0, however, this effect is frequently positive, that is, the compression yields autocorrection of the approximations, according to the extensive experiments reported in [P01, Table 6.21].

Unfortunately, the initial approximate inverses X_0 computed in [P01], [P01a], and [PKRK04] are not in the latter class. Furthermore, they do not support the convergence acceleration via scaling proposed in [PS91]. This means that we lose our chance to achieve the autocorrection as well as the acceleration by means of scaling.

In our present paper we modify our homotopic processes to supply the initial inverses X_0 free of these deficiencies. Our algorithm extends the algorithms in [PS92], [PZDH95]. In these papers a Toeplitz-like matrix T is factorized as the product $T_0T_1\cdots T_h$ where T_i are Toeplitz-like matrices and any number h and any condition numbers $\mathrm{cond}_2(T_i)$, $i = 0, 1, \ldots, h$, can be selected such that $\mathrm{cond}_2(T) = \prod_{i=0}^{h} \mathrm{cond}_2(T_i)$. This enables accelerated inversion of the matrix T based on the application of the Conjugate Gradient algorithm to the factors T_i rather than to their product T. Now we view the inversion of such factors as a homotopic process of the inversion of the matrix T, observe that the approximate inverses of $T_0, \ldots, T_h$ are readily available, and apply Newton's iteration to invert all or some of them.

Our approach can be extended to various other classes of structured matrices.

We begin with an extensive background section and then present and analyze our approach in Sections 2–7.

1 Background on Iterative Matrix Inversion

1.1 Newton's iteration for general matrix inversion

Consider Newton's iteration for matrix inversion

$$X_{i+1} = X_i(2I - MX_i),\ i = 0, 1, \dots. \tag{1.1}$$

It squares the residuals $I - X_iM$ and $I - MX_i$ in every step and is strongly stable numerically.

Hereafter, except for Theorems 4.1 and 5.1, let us assume that M is an $n \times n$ Hermitian (or real symmetric) and positive definite matrix with the eigenvalues $\lambda_1, \dots, \lambda_n$ such that

$$\lambda^+ \geq \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq \lambda_- > 0 \tag{1.2}$$

(see Section 7 on the extensions of our study). Let us follow [PS91] to the end of this subsection. Choose

$$X_0 = I/||M||_F \tag{1.3}$$

and write

$$\kappa_2 = \mathrm{cond}_2(M) = \lambda_1/\lambda_n,$$

$$R_i = I - X_iM = I - MX_i = R_{i-1}^2 = R_0^{2^i},\ i = 0, 1, \dots.$$

Deduce that

$$||R_0||_2 \leq 1 - 1/(n^{1/2}\kappa_2)$$

and, consequently,

$$||R_i||_2 < e^{-g}$$

for $e = 2.7182812\dots$, any positive g, and every $i \geq 1 + \ln(n^{1/2}\kappa_2) + \log_2 g$.

We need roughly by twice fewer Newton's steps if we modify the choice (1.3) and iteration (1.1) to have

$$R_i = C_{2^i}(\gamma M + \delta I)/C_{2^i}(\delta),\ i = 0, 1, \dots.$$

Here

$$\gamma = 2/(\lambda_1 - \lambda_n),\ \delta = -(\lambda_1 + \lambda_n)/(\lambda_1 - \lambda_n), \tag{1.4}$$

and $C_h(y)$ denotes the h-th degree Chebyshev polynomial $\cos(h \arccos y)$. We arrive at this modification by choosing

$$X_1 = (8/\sigma)(-M + (\lambda_1 + \lambda_n)I),\ \sigma = 4\lambda_1\lambda_n + (\lambda_1 + \lambda_n)^2, \tag{1.5}$$

$$X_{i+1} = \alpha_{i+1} X_i (2I - MX_i), \tag{1.6}$$

$$\alpha_{i+1} = (1 + \beta_{i+1})/\beta_{i+1},\ \beta_{i+1} = C_{2^{i+1}}(\delta) = 2\beta_i^2 - 1 \tag{1.7}$$

for $i = 0, 1, \ldots$; δ in (1.4), and

$$\beta_0 = -\frac{\lambda_i + \lambda_n}{\lambda_i - \lambda_n}. \tag{1.8}$$

Practically, one may apply the power or (better) Lanczos method to compute a close upper bound λ^+ on λ_1 and then apply the same method to the matrix $\lambda^+ I - M$ to compute $\overline{\lambda}_n$ such that $0 \le \overline{\lambda}_n - \lambda_n \le \epsilon \lambda_1$ for a small ϵ defined by the computer precision. By substituting λ^+ for λ_1 and $\overline{\lambda}_n$ for λ_n in equations (1.4)–(1.8), we change the residual norms $||R_i||_2$ by $\rho_i(\epsilon) = O(\epsilon)$; for quite a few first iterates the changes are not significant and little affect the convergence.

1.2 The Toeplitz and Toeplitz-like cases

For the general matrices M, Newton's steps (1.1)–(1.3) and (1.5)–(1.8) are quite expensive; each of them uses $2n^3 + O(n^2)$ flops. This situation dramatically changes where M has structure of Toeplitz type or, as we say, is a *Toeplitz-like* matrix. By this we mean that M is given with its displacement generator (G, H) of length $r = O(1)$ where G and H are $n \times r$ matrices, $GH^T = L(M)$ for a fixed displacement operator L associated with Toeplitz structure, and some fixed bilinear expressions define M via G and H. We fix the Sylvester type operator $L = \nabla_{Z_1, Z_{-1}} L(M) = Z_1 M - M Z_{-1}$, where Z_1 and Z_{-1} are the unit circulant and skew-circulant shift matrices, respectively, and we write

$$Z_f = \begin{pmatrix} 0 & & & f \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{pmatrix}$$

for any real or complex scalar f. We have $L_-(M^{-1}) = Z_{-1} M^{-1} - M^{-1} Z_1 = -(M^{-1}G)H^T M^{-1}$. For Toeplitz matrices M, we have rank $L(M) \le 2$.

Our choice of the operator L can be modified (e.g., by replacing Z_1 by Z_e and Z_{-1} by Z_f for any pair of real or complex e and f such that $e \ne f$ or by using the Stein type operators $M \longrightarrow M - Z_e M Z_f^T$, $ef \ne 1$, instead of $L = \nabla_{Z_1, Z_{-1}}$), but hereafter we stay with our operators L and L_- (above) and we say "L-generator", "L-length", "L_--generator", and "L_--length", thus replacing the word "displacement" by "L-" or "L_--". We refer the reader to [P01, Chapters 1 and 4] and the bibliography therein on the definitions and basic properties of Toeplitz-like matrices and displacements.

In particular, we need the fact that for X_{i+1} in (1.1)–(1.3) or (1.5)–(1.8) an L_--generator of length of at most $r + 2r_i$ can be computed by using $O((r + r_i)^2 n \log n)$ flops provided an L_--generator for X_i of length r_i and an L-generator for M of length r are available. (The length r can be tripled if $r_i = r$.) Furthermore, for any pair of scalars α and β, we immediately obtain an L_--generator of length of at most $r+1$ for the matrix $\alpha I + \beta M$, in particular for X_1 in (1.5). (This length bound decreases to 2 if M is a Toeplitz matrix.) Therefore, the first Newton's steps (1.1)–(1.3) or (1.5)–(1.8) can be performed by using $O(n \log n)$ flops if $r = O(1)$.

1.3 The effects of compressing the displacements

The computed L_--generators of X_i may have their length tripled in every Newton's step, thus implying rapid increase of the number of flops per step. To save flops, we apply Newton's step (1.1) with X_i compressed into a matrix Y_i which lies nearby and has a shorter L_--generator (e.g., of length of at most r, $2r$, or $3r$). Three distinct compression techniques can be found in [P01, Chapter 6], [PRW02], [PVWC04]. In all cases, the transition to Y_i does not destroy the approximation to M^{-1}, and the number of flops used for the compression is dominated by the cost of performing steps (1.1)–(1.3) or (1.5)–(1.8). On the negative side, all these techniques

a) generally destroy the power of scaling (1.5)–(1.8) and

b) may increase the residual norm $||R_i||_2$ by some factor $f > 1$.

Recall that $R_{i+1} = R_i^2$ under (1.1). Therefore, with the compression we have $||R_{i+1}|| \leq f^2 \ ||R_i||_2^2$. Consequently

$$||R_{i+k}||_2 \leq \theta^{2^k} \ ||R_i||_2$$

provided $\theta \geq f^2 ||R_i||_2$, $||R_i||_2 \leq \theta / f^2$. The convergence is implied if $\theta < 1$, $||R_0||_2 < 1/f^2$; the smaller $||R_0||_2$ and θ, the faster convergence.

Hereafter, let b^+ denote a value such that Newton's steps (1.1) with a selected policy of compression rapidly converge provided

$$||R_0||_2 \leq 1/b^+. \tag{1.9}$$

We have $f = O(||M||_2\ n\ \kappa_2)$ according to the estimates in [P01, Chapter 6], [PRW02], and [PW03], which extend the earlier works [P92], [P93], and [PBRZ99]. Therefore, rapid convergence of steps (1.1) with compression can be ensured already if (1.9) holds for some b^+ in $O((||M||_2\ n\kappa_2)^2)$. Extensive experiments with Toeplitz matrices M and X_0 in (1.3) have been performed in the City University of New York by M. Kunin. He assumed the compression policy via truncation of a fixed number of the smallest singular values of the displacement with no scaling [P01, Chapter 6]. According to these experiments, the latter worst case bound is overly pessimistic; in particular rapid convergence was maintained quite steadily for $b^+ = 1.1$. Moreover, for about 25% of the input matrices M, the compression implied convergence for b^+ equal to or even slightly smaller than 1 (see [P01, Table 6.21]). This phenomenon is called *autocorrection in compression* in [CPWa], [PKRK04].

1.4 Linearly convergent preprocessing

To yield a crude initial approximation X_0, we approximate an L_--generator (G_-, H_-) for M^{-1} where

$$MG_- = -G,\ \ M^T H_- = H, \tag{1.10}$$

$L(M) = \nabla_{Z_1,Z_{-1}}(M) = GH^T$, $L_-(M^{-1}) = \nabla_{Z_{-1},Z_1}(M^{-1}) = G_-H_-^T$. We may apply any iterative process for solving linear systems with the coefficient matrices M and M^T. A simple process for a linear system $M\mathbf{x} = \mathbf{g}$ is given by

$$\begin{gathered} \mathbf{x}_0 = \mathbf{0},\ \ \mathbf{r}_0 = \mathbf{g}, \\ \boldsymbol{\Delta}_i = \mathbf{x}_{i+1} - \mathbf{x}_i = X_0\mathbf{r}_i,\ \ \mathbf{r}_{i+1} = \mathbf{r}_i - M(\mathbf{x}_{i+1} - \mathbf{x}_i), \\ i = 0, 1, \ldots, s, \end{gathered} \tag{1.11}$$

$\mathbf{x} \approx \mathbf{x}_s = \Sigma_{i=0}^s \boldsymbol{\Delta}_i$.

These equations imply linear convergence to $\mathbf{x}$:

$$\mathbf{x}_s - \mathbf{x} = R_0(\mathbf{x}_{s-1} - \mathbf{x}) = R_0^s(\mathbf{x}_0 - \mathbf{x}),$$

$$||\mathbf{x}_s - \mathbf{x}||_2 \le ||R_0||_2^s||\mathbf{x}_0 - \mathbf{x}||_2.$$

To ensure bound (1.9) for b^+ of the order of $(||M||_2 n\kappa_2)^2$, we choose s of the order of $\log(||M||_2 n\kappa_2)$; this also enables us to exclude the error propagation in the transition from an approximate solution of the system of equations (1.10) to an approximate inverse of M [PW03].

Alternatively we may apply the (preconditioned) congugate gradient algorithm to yield an approximate solution to (1.10) provided that M is well conditioned. Hereafter we use the abbreviation "CG" for "(preconditioned) congugate gradient." In s CG steps, the M-norm of the initial error vector $\mathbf{x} - \mathbf{x}_1$ decreases by the factor of $2(\frac{\sqrt{\kappa_2}-1}{\sqrt{\kappa_2}+1})^s$ (see [GL96, Theorem 10.2.6]) where M-norm of a vector $\mathbf{v}$ is $(\mathbf{v}^T M \mathbf{v})^{1/2}$.

One may continue using the CG algorithm or process (1.11) to improve the approximation to M^{-1} beyond (1.9). The convergence is linear (that is, slower than with (1.1)) but no compression is involved (which avoids a potential source of problems).

1.5 The homotopic (continuation) approach

For ill conditioned input matrices M, the residual norms $||R_0||_2$ for X_0 in (1.3) and $||R_1||_2$ for X_1 in (1.5) are close to 1; decreasing them, say below $1/e$, $e = 2.718281828\ldots$, takes many Newton's steps, even with scaling. The computed L_--generators of the approximations X_i become long, and Newton's steps involve many flops. To avoid this problem, homotopic (continuation) techniques were proposed in [P92] and further studied in [P01a], [P01, Section 6.9], and [PKRK04]. In this approach the inversion of a given matrix M is replaced by the inversion of a sequence of matrices $M_0, M_1, \ldots, M_{l+1} = M$. Here M_0 is readily invertible (e.g., $M_0 = M + aI$ for a larger scalar a and the identity matrix I), $M_k = M + a_k M_0$, $k = 1, 2, \ldots$, with a_k decreasing to 0 as k grows, and we invert each M_k by applying Newton's iteration with the initial approximation $X_0 = M_{k-1}^{-1}$. This enables us to ensure the initial residual norm of at most $1/b$ (for any fixed $b > 1$) for each of the $l+2$ matrix inversions in the homotopic process and simultaneously the upper bound of $\lceil \log(1 + b\kappa_2^+)/\log(1 + 1/(b-1)) \rceil$ on the number $l+1$ of homotopic steps [PKRK04]. Here $\kappa_2^+ = \frac{\lambda^+}{\lambda_n}$ for a fixed $\lambda^+ \geq \lambda_1$ (cf. (1.2)). We can see that the closer b to 1, the fewer homotopic steps are needed but the larger initial residual norms must be handled.

To a certain disadvantage of this approach in [P01, Section 6.9] and [PKRK04], the latter choice of the initial approximate inverses M_{k-1}^{-1} in the k-th homotopic step is distinct from (1.3) and does not support autocorrection in compression. Moreover, this choice does not allow us to apply the techniques of the initialization and scaling from [PS91] (see (1.3), (1.5)–(1.8)), which save about 50% of Newton's steps. Another deficiency of the algorithm in [PKRK04] is that we must invert matrices M_k whose condition numbers can be almost as large as $\kappa_2 = \text{cond}_2(M)$. This does not welcome using the CG algorithm.

2 Our Approach (the Basic Idea) and the Organization of the Rest of the Paper

Let us modify the homotopic process to counter the latter deficiencies. As in [PKRK04], we first invert $M_0 = M + aI$ for a larger $a = a_0$, but then, instead of the matrices $M_1, M_2, \ldots, M_{l+1}$, we recursively invert the matrices $P_0, P_1, \ldots, P_{l-1}$, and finally $N_l = M_l^{-1}M$ where $P_k = M_{k+1}M_k^{-1}$ for all k, $M_l = P_{l-1} \cdots P_0 M_0$. This technique is similar to the preconditioning techniques in [PS92], [PZDH95]. Bounds on the ratios a_k/a_{k+1} imply some bounds on the condition numbers of the matrices P_k and consequently on the number of the CG steps for the inversion of these matrices. For the alternative inversion with Newton's iteration we may use the identity matrix I as the initial approximate inverse. The initial residual norm $||R_0|| = ||I - \tilde{P}_k||$ can be bounded in terms of the ratio a_k/a_{k+1}, and in Corollary 5.4 we estimate the overall computational cost of the inversion of M based on this approach.

We organize the rest of our paper as follows. In the next section we relate our homotopic process to scaling and compression in Newton's interation. In Section 4, we describe our generic algorithm for general input matrices. In Section 5, we first express the spectra of the auxiliary matrices via the extremal eigenvalues of M and the auxiliary parameters $a_0, \ldots, a_l$; then, in the symmetric positive definite and Toeplitz–like case, we estimate their residual norms and condition numbers as well as the overall number of Newton's steps. In Section 6, we estimate the length of the displacement generators of the auxiliary matrices. In Section 7, we discuss some extensions and modifications and estimate from below the overall number of homotopic steps.

3 Two-Stage Initialization of Newton's Iteration

The initial bounds (1.9) for $R_0 = I - P_k$ and sufficiently large b^+ enable rapid inversion of the auxiliary matrices P_k with Newton's iteration. Achieving (1.9) could require too many homotopic steps, but we may apply a two-stage adaptive approach. We first fix $b \leq b^+$, write $a = a_0$, and select the parameters $l, a_0, a_1, \ldots, a_{l-1}$ to ensure that

$$||I - P_k||_2 \leq 1/b \tag{3.1}$$

for $k = 0, 1, \ldots, l-1$ (see Corollary 5.3). The respective values of l, a_0, a_1, …, a_{l-1} can be computed adaptively, which is particularly attractive for computing l and a few a_i with the largest i. Our explicit estimates in the next sections give us an alternative and may also serve as the initial landmarks in the adaptive computations.

For the transition from the norm bound (3.1) to the bound (1.9) for $R_0 = I - P_k$, we may use scaled Newton's steps (1.5)–(1.8). According to [PS91], $N(b, b^+) \approx \log_4(\log_b\ b^+)$ such steps are sufficient. Alternatively, we may seek transition to (1.9) by using unscaled Newton's iteration, hoping to achieve a speed up due to the autocorrection in compression. Another alternative is to apply linearly convergent processes such as (1.11) or the CG algorithm.

4 The Basic Theorem and the Generic Algorithm

We begin with the case of a general matrix M and a generalized homotopic process, and we impose further restrictions in the next sections.

Theorem 4.1. *Let M and S be a pair of n-by-n matrices. Let $a = a_0 \neq 0$, $b_k > 1$, and $a_{k+1} = a_k(1-1/b_k) = a\prod_{j=0}^{k}(1-1/b_j)$ be scalars for $k = 0, 1, \ldots$. Write $M_k = M + a_k S$, $P_k = I - a_k S(b_k M_k)^{-1}$, $k = 0, 1, \ldots$. Suppose that the matrices M_k are nonsingular for all k. Then we have*

a) $M_{k+1} = P_k M_k = P_k P_{k-1} \cdots P_0 M_0$,

b) $M = P_{k-1} \cdots P_0 M_0 N_k$ *where* $N_k = M_k^{-1} M$, *and* *c)* $I - N_k = a_k M_k^{-1} S$

for $k = 0, 1, \ldots$.

Theorem 4.1 enables us to compute recursively the matrices M_0, M_0^{-1}, P_k, P_k^{-1}, M_{k+1}, M_{k+1}^{-1}, for $k = 0, 1, \ldots, l-1$ and a fixed l, and $N_l = M_l^{-1} M$. Then if M is nonsingular, we compute the inverses N_l^{-1} and

$$M^{-1} = N_l^{-1} M_l^{-1} = N_l^{-1} M_0^{-1} P_0^{-1} \cdots P_l^{-1}.$$

This is our generic algorithm. Besides the inversion, the latter factorization and the one in Theorem 4.1b may have other applications, e.g., to computing $\det M$.

5 Residual Norms, Condition Numbers, and the Overall Number of Newton's Steps

The upper estimates for the complexity of the inversion of the matrices M_0, P_0, ..., P_{l-1}, and N_l depend on the initial residual norm if we use Newton's iteration and on the condition number of these matrices if we apply the *CG* algorithm. Let us estimate the latter quantities.

Theorem 5.1. *Let M have the spectrum $\Lambda(M) = \{\lambda_1, \ldots, \lambda_n\}$ and let $S = I$. Then under the assumptions of Theorem 4.1, we have*

$$\begin{array}{rcll} \Lambda(I - M_0/a) & = & \{-\lambda_s/a\}_{s=1}^n, & a = a_0, \\ \Lambda(I - P_k) & = & \{(a_k/b_k)/(a_k + \lambda_s)\}_{s=1}^n, & a_{k+1} = a_k(1 - 1/b_k), \\ & & & k = 0, 1, \ldots, l-1, \\ \Lambda(I - N_l) & = & \{a_l/(\lambda_s + a_l)\}_{s=1}^n. & \end{array}$$

Corollary 5.2. *Under the assumptions of Theorem 5.1, let M be a Hermitian (or real symmetric) and positive definite matrix and let*

$$\lambda^+ \geq \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq \lambda_- > 0,$$

$$a = a_0 > 0, \quad b_k > 1, \quad a_{k+1} = a_k(1 - 1/b_k), \quad k = 0, 1, \ldots.$$

Then

a) the matrices M_0, $M_0/a - I$, P_k, $I - P_k$ for $k = 0, 1, \ldots, l-1$, N_l and $I - N_l$ are Hermitian (or real symmetric) and positive definite, *b)* $||I - M_0/a||_2 \leq \lambda_1/a \leq \lambda^+/a$, $||I - P_k||_2 < 1/b_k$, $k = 0, 1, \ldots, l-1$,

$||I - N_l||_2 \leq 1/(1 + \lambda_n/a_l) \leq 1/(1 + \lambda_-/a_l)$,

c) $\mathrm{cond}_2(M_0) = 1 + \frac{\lambda_1 - \lambda_n}{a + \lambda_n} < 1 + \lambda_1/a \leq 1 + \lambda^+/a$,

$\mathrm{cond}_2(P_k) = \frac{1-(a_k/b_k)/(a_k+\lambda_1)}{1-(a_k/b_k)/(a_k+\lambda_n)} < 1/(1 - 1/b_k) = a_k/a_{k+1}$,

$k = 0, 1, \ldots, l-1$,

$\mathrm{cond}_2(N_l) = \frac{1-a_l/(\lambda_1+a_l)}{1-a_l/(\lambda_n+a_l)} < 1 + a_l/\lambda_n \leq 1 + a_l/\lambda_-$.

By relying on Corollary 5.2b, let us estimate the overall complexity of the inversion of a matrix M where we apply Newton's iteration to invert the matrices M_0, P_0, …, P_{l-1}, and N_l.

Corollary 5.3. *For a scalar $b > 1$, write $\kappa^+ = \lambda^+/\lambda_-$, $a = \lambda^+ b$, $l = l(b) = 1 + \lceil \log((b-1)b\kappa^+)/\log(1 + 1/(b-1)) \rceil$, $b_k = b$, $k = 0, 1, \ldots, l-1$. Then none of the residual norms in Corollary 5.2b exceeds $1/b$.*

Corollary 5.4. *Under the asssumptions of Corollary 5.3, $N_+(M) = (l+1)N(b, b^+)$ Newton's steps (1.5)–(1.8) and $N(M) = (l+1)N(b^+)$ Newton's steps (1.1)–(1.3) with compression are sufficient to invert M provided $N(b, b^+) \approx \log_4(\log_b\ b^+)$, Newton's steps (1.5)–(1.8) are sufficient to reach the norm bound (1.9) provided we initially have the bound (3.1), whereas $N(b^+)$ Newton's steps (1.1)–(1.3) with compression are sufficient for the convergence where the initial residual norm is $1/b^+$ or less. In particular,*

$$
\begin{array}{lcll}
l+1 & = & O(\log(\kappa^+/\nu)/\log(\nu+1)) & \text{for } \nu = 1/(b^+ - 1), \\
l+1 & \leq & \lceil 2 + \log_2 \kappa^+ \rceil & \text{for } b^+ = 2, \\
l+1 & \leq & \lceil 2 + \log_{11}(0.1\kappa^+) \rceil & \text{for } b^+ = 1.1, \\
l+1 & \leq & \lceil 2 + \log_{10^d+1}(10^{-d}\kappa^+) \rceil & \text{for } b^+ = 1 + 10^{-d} \text{ and any real } d.
\end{array}
$$

Corollary 5.4 supplies some upper bounds on the number of Newton's steps and guides us in choosing the parameters $a, l, b_0, \ldots, b_{l-1}$. These bounds and guidance should be revised for practical computations because of the helpful effect of the autocorrection in compression. Further correction is in order if one chooses to invert all or some of the factors M_0, P_0, …, P_{l-1}, and N_l by applying the *CG* algorithm. The relevant upper estimates for the complexity of the latter approach and the guidance for the choice of the parameters may rely on Corollary 5.2c or can be extracted from [PS92], [PZDH95]. (These upper estimates should be revised if the convergence of the CG algorithm is accelerated with preconditioning.) The following equation is basic for the study along this line:

$$\mathrm{cond}_2(M) = \mathrm{cond}_2(M_0)\,\mathrm{cond}_2(N_l)\prod_{k=0}^{l-1}\mathrm{cond}_2(P_k).$$

This equation holds for any matrix S in Theorem 4.1 (not only for $S = I$) and implies the following lower bound on the number l of homotopic steps:

$$l + 1 > \log_\kappa \mathrm{cond}_2(M)$$

for every κ exceeding the condition numbers of the matrices M_0, P_0, …, P_{l-1}, and N_l.

6 Some Bounds on the Length of Displacement Generators

Corollary 5.4 shows substantial effect of decreasing the ratio $(b-1)/(b^+-1)$ on decreasing the number of homotopic steps. The limiting factor, however, is the resulting increase of the L_--length of the approximants X_i in the process (1.5)–(1.8). The next simple theorem (immediately implied by the basic results in [P01, Chapter 1]) bounds the length of the displacement generators of the auxiliary matrices in our homotopic process.

Theorem 6.1. *Under the assumptions of Theorem 5.1, let M be a Toeplitz-like matrix given with its L-generator of length r for the Sylvester type operator $L = \nabla_{Z_1,Z_{-1}}$. Then L- and L_--generators for the matrices M_k and P_k for $k = 0, 1, \ldots,$ having length of at most $r+1$ and $r+2$, respectively, can be computed in $O(r^2 n \log n)$ flops. If M is a Toeplitz matrix, then M_k and P_k have displacement ranks of at most 2 and 3, respectively, and $O(n \log n)$ flops are sufficient to compute the displacement generators of the minimal length 2 or 3 for these matrices.*

Remark 6.2. If $S \neq cI$ for a scalar c, the displacement rank and length bounds of Theorem 6.1 generally grow; e.g., for $S = cI - M$ and a nonzero scalar c, the displacement rank of P_k may grow to $r+4$; moreover, for this S computing P_k requires an extra matrix multiplicaton of S by M_k^{-1} for every k. These drawbacks overweigh the advantage of the immediate inversion of the matrix $M_0 = cI$.

7 Modifications and Extensions

Our factorization/homotopy approach reduces the inversion of a Toeplitz-like matrices to the inversion of any number l of its Toeplitz-like factors. With the growth of l these factors become progressively better conditioned, and their closer approximate inverses become available. This gives us an option of effective inversion of the factors with both the CG and Newton's algorithms as well as other algorithms such as the generalized Schur and the MBA algorithms [KS95], [P01, Chapter 5]. Further theoretical and experimental study should provide some guidance for the choice among all these options as well as for the parameters a, l, $b_0, \ldots, b_{l-1}$. Surely, the inversion algorithm does not have to be invariant for all factors. We have specified our algorithms in the case where the input matrix is Hermitian (or real symmetric), positive definite, and Toeplitz-like. The extension to Hankel-like and Toeplitz + Hankel-like matrices is immediate [P01]. For the extension to the structures of other types (such as Cauchy-Pick's and Vandermonde's), one may consider the general method of displacement tranformation [P90] or the more direct approaches in [P01, Section 6.9] and [PKRK04, Section 10], covering also Hermitian (or real symmetric) indefinite and nonHermitian (unsymmetric) inputs.

References

[CPWa] G. Codevico, V. Y. Pan, M. Van Barel. Newton-like Iteration Based on Cubic Polynomials for Structured Matrices, accepted by *Numerical Algorithms*.

[GL96] G. H. Golub, C. F. Van Loan. *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1996.

[KS95] T. Kailath, A. H. Sayed. Displacement Structure: Theory and Applications, *SIAM Review*, **37, 3**, 297–386, 1995.

[P90] V. Y. Pan. On Computations with Dense Structured Matrices, *Mathematics of Computation*, **55, 191**, 179–190, 1990. Proceedings version: *Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'89)*, 34–42, ACM Press, New York, 1989.

[P92] V. Y. Pan. Parallel Solution of Toeplitz-like Linear Systems, *Journal of Complexity*, **8**, 1–21, 1992.

[P93] V. Y. Pan. Decreasing the Displacement Rank of a Matrix, *SIAM Journal on Matrix Analysis and Applications*, **14, 1**, 118–121, 1993.

[P01] V. Y. Pan. *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser/Springer, Boston/New York, 2001.

[P01a] V. Y. Pan. A Homotopic Residual Correction Process, *Proceedings of the Second Conference on Numerical Analysis and Applications* (P. Yalamov, editor), *Lecture Notes in Computer Science*, **1988**, 644–649, Springer, Berlin, 2001.

[PBRZ99] V. Y. Pan, S. Branham, R. Rosholt, A. Zheng. Newton's Iteration for Structured Matrices and Linear Systems of Equations, *SIAM Volume on Fast Reliable Algorithms for Matrices with Structure* (T. Kailath and A. H. Sayed, editors), 189–210, SIAM Publications, Philadelphia, 1999.

[PKRK04] V. Y. Pan, M. Kunin, R. Rosholt, H. Kodal. Homotopic Residual Correction Processes, Technical Report TR 2004, *Ph.D. Program in Computer Science, Graduate Center, City University of New York*, New York, 2004.

[PRW02] V. Y. Pan, Y. Rami, X. Wang. Structured Matrices and Newton's Iteration: Unified Approach, *Linear Algebra and Its Applications*, **343–344**, 233–265, 2002.

[PS91] V. Y. Pan, R. Schreiber. An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications, *SIAM Journal on Scientific and Statistical Computing*, **12, 5**, 1109–1131, 1991.

[PS92] V. Y. Pan, R. Schreiber. A Fast, Preconditioned Conjugate Gradient Solver, *Computers and Math. (with Applics.)*, **24, 7**, 17–24, 1992.

[PVWC04] V. Y. Pan, M. Van Barel, X. Wang, G. Codevico. Iterative Inversion of Structured Matrices, *Theoretical Computer Science*, **315**, **2**–**3**, 581–592, 2004.

[PW03] V. Y. Pan and X. Wang. Inversion of Displacement Operators. *SIAM Journal on Matrix Analysis and Applications*, **24, 3**, 660–677, 2003.

[PZDH95] V. Y. Pan, A. L. Zheng, O. Dias, X. H. Huang. A Fast, Preconditioned Conjugate Gradient Toeplitz and Toeplitz-Like Solvers, *Computers and Math. (with Applics.)*, **30, 8**, 57–63, 1995.