

HOMOTOPIC RESIDUAL CORRECTION PROCESSES

V. Y. PAN, M. KUNIN, R. E. ROSHOLT, AND H. KODAL

ABSTRACT. We present and analyze homotopic (continuation) residual correction algorithms for the computation of matrix inverses. For complex indefinite Hermitian input matrices, our homotopic methods substantially accelerate the known nonhomotopic algorithms. Unlike the nonhomotopic case our algorithms require no pre-estimation of the smallest singular value of an input matrix. Furthermore, we guarantee rapid convergence to the inverses of well-conditioned structured matrices even where no good initial approximation is available. In particular we yield the inverse of a well-conditioned $n \times n$ matrix with a structure of Toeplitz/Hankel type in $O(n \log^3 n)$ flops. For a large class of input matrices, our methods can be extended to computing numerically the generalized inverses. Our numerical experiments confirm the validity of our analysis and the efficiency of the presented algorithms for well-conditioned input matrices and furnished us with the proper values of the parameters that define our algorithms.

1. INTRODUCTION

Newton's iteration and other residual correction processes rapidly improve crude initial approximations to the inverse or Moore–Penrose generalized inverse of a matrix. Hereafter we write “RC” for residual correction. We study the RC processes, which are strongly numerically stable, allow effective parallel implementation, and converge at least quadratically right from the beginning as long as a reasonably close initial approximation to the inverse is available. The requirement of having a good initial approximation becomes even more critical in the highly important case in which the input matrix is structured, e.g., a Toeplitz, Hankel, Vandermonde, Cauchy, or Pick matrix.

In this paper we solve the initialization problem by applying homotopic (continuation) RC processes. Hereafter we refer to them as HRC processes. We first specify and then analyze and optimize them for Hermitian (or real symmetric) matrices, both positive definite and indefinite. Then we comment on the special case where

Received by the editor December 20, 2001 and, in revised form, March 10, 2004.

2000 *Mathematics Subject Classification.* Primary 65F10, 65F30.

Key words and phrases. Residual correction, Newton's iteration, homotopic (continuation) algorithms, (generalized) inverse matrix.

This work was supported by NSF Grant CCR9732206, PSC CUNY Awards 63383-0032 and 66406-0033, and a Grant from the CUNY Institute for Software Design and Development (CISDD).

The results of this paper were presented at the Second Conference on Numerical Analysis and Applications, Rousse, Bulgaria, in June 2000, and at the AMS/IMS/SIAM Summer Research Conference on Fast Algorithms in Mathematics, Computer Science, and Engineering in South Hadley, Massachusetts, in August 2001.

©2005 American Mathematical Society
Reverts to public domain 28 years from publication

the input matrix is structured, and we outline extensions to numerical computation of the Moore–Penrose generalized inverse (with some limitations in the case of structured matrices).

For Hermitian matrices (both positive definite and indefinite) our HRC processes require about $0.5 \log_2 \kappa(M)$ RC steps (performing matrix multiplication twice per step), where $\kappa(M)$ is the condition number of the input matrix M . This is by twice less than with the known RC processes in the complex indefinite case and about as many as known in the positive definite case.

Furthermore, unlike their nonhomotopic RC counterparts, the HRC processes do not require pre-estimation of the smallest singular values of the input matrices.

In the case of structured matrices M , each matrix multiplication is reduced to multiplication of a small number of structured matrices by vectors (that is, to $O(n \log n)$ flops for an $n \times n$ Toeplitz matrix M) [KKM79], [KS99], [P01a]. To preserve structure during an HRC process, one may have to increase the number of RC steps by the factor $F = O(\log(n \kappa(M)))$, which is an overly pessimistic bound according to our experimental tests. The resulting overall arithmetic cost bound for Toeplitz matrix inversion is $O(F^2 \log \kappa(M) n \log n)$ flops. In the case of well-conditioned Toeplitz matrices M for which $\kappa(M) = O(1)$, this yields the inversion cost in $O(n \log^3 n)$ flops, which is supported by no alternative iterative methods. Some additional practical (nonasymptotic) acceleration can be achieved by combining our homotopic algorithms with linearly convergent iterations for linear systems. The flop estimates in $O(n \log^3 n)$ are also supported and even slightly improved by the superfast divide-and-conquer direct algorithms [P01a, Chapter 5], but they have weaker numerical stability (cf. [B85] and [PS91]) and do not handle the computation of numerical generalized inverses.

Our numerical tests have confirmed the results of our analysis for general and Toeplitz input matrices and furnished us with appropriate values of the parameters of the HRC processes.

The approach was first proposed in [P92] in a cruder form; our present progress was partly announced in [P01b] and [P01a, Chapter 6].

Our theoretical estimates for the number of HRC steps grow in proportion to $\log \kappa(M)$. For ill-conditioned Toeplitz matrices these estimates are inferior to the bound of $O(n^2)$ flops for computing the inverses by means of fast and stable direct solution methods. Likewise the SVD based methods for generalized inverses of ill-conditioned matrices get the upper hand.

The phenomenon called *autocorrection in compression* and experimentally observed for the RC processes for Toeplitz matrices shows, however, a certain increase of the efficiency of these processes not explained by the theory. This leaves room for further experimental study, which may eventually change the above pessimistic conclusions about the limitation of the power of HRC processes. This would require extensive additional study.

Our paper is organized as follows. In Section 2 we recall some known RC processes. In Sections 3–5 we describe and analyze the HRC processes for Hermitian positive definite matrices, and in Section 6 for Hermitian indefinite ones. In Section 7 we extend our straightforward initialization policy. In Section 8 we comment on the specialization of the RC and HRC processes to structured matrices. In Section 9 we report the results of some numerical tests with real symmetric Toeplitz matrices. Section 9 and the experimental work for Table 5.1 are coauthored by all

authors, with the main responsibility for organizing and running the experiments covered in Section 9 and Table 5.1 by the second and the third authors, respectively; all other sections are due to the first author. In Section 10 we outline extensions to computing numerically the Moore–Penrose generalized inverses. In Section 11 we summarize our study of HRC processes and compare them with alternative algorithms.

Acknowledgment. We thank the referee for very helpful comments.

2. RESIDUAL CORRECTION PROCESSES

Hereafter, M^T and M^* denote the transpose and the Hermitian (conjugate) transpose of a matrix M , respectively. $\lceil x \rceil$ is the smallest among the integers not exceeded by a real x .

2.1. A basic RC process. A close initial approximation X_0 to the inverse of a nonsingular matrix M can be rapidly improved by means of a scaled RC process

$$(2.1) \quad \Delta_i = \Delta(X_i) = X_{i+1} - c_{i+1}X_i = c_{i+1} \sum_{k=1}^{p-1} R_i^k X_i, \quad i = 0, 1, \dots$$

[IK66, pages 86-88], where we write

$$(2.2) \quad R_i = R(M, X_i) = I - X_i M = R_{i-1} - M(X_{i-1} - X_i).$$

For $p = 2$, $c_{i+1} = 1$ for all i , we arrive at Newton’s iteration [S33]. For $p = 2^h$, we may compute $\sum_{k=0}^{p-1} R_i^k$ as $\prod_{j=0}^{h-1} (I + R_i^{2^j})$ by using fewer additions.

In the special unscaled case, where

$$(2.3) \quad c_i = 1 \text{ for all } i,$$

(2.1) and (2.2) imply that

$$R_i = (R_0)^{p^i}, \quad \|R_i\| \leq \|R_0\|^{p^i}, \quad i = 1, 2, \dots,$$

for any fixed matrix norm. That is, the unscaled RC process (2.1), (2.3) converges with the order of p to the matrix M^{-1} provided that $\|R_0\|_2 \leq \theta < 1$.

2.2. An initial approximation. Suppose a positive lower bound σ_- and an upper bound σ_+ on the singular values of M are available. Then, for an initial approximation X_0 to the matrix M^{-1} , we may choose

$$(2.4) \quad X_0 = c_0 M^*, \quad c_0 = \frac{2}{\sigma_+^2 + \sigma_-^2},$$

to yield that

$$\|R_0\|_2 \leq 1 - \frac{2}{1 + \kappa_+^2}, \quad \kappa_+ = \kappa_+(M) = \sigma_+/\sigma_-$$

(see [SS74]). Now the first $c = 2 \log_p \kappa_+ + O(1)$ RC steps (2.1), (2.3) (we call them *critical RC steps*) decrease the residual norm $\|R_i\|_2$ below $1/e = 1/2.718281 \dots = 0.367819 \dots$; the next $\rho = \lceil \log_p \ln(1/\delta) \rceil$ RC steps (we call them *refinement RC steps*) decrease the norm below a positive $\delta \leq 1/e$ [SS74].

Let $\kappa(M)$ denote the condition number M , that is, the largest ratio of two singular values of M , $\kappa(M) < \kappa_+$. Then the simpler initial choice of

$$X_0 = M^*/(\|M\|_1 \|M\|_\infty), \quad \|R_0\|_2 \leq 1 - 1/(n\kappa^2(M)),$$

proposed in [B-I66] implies the bound $i_- = \log_2(n\kappa^2(M)) + O(1)$ on the number of critical RC steps. For a Hermitian (or real symmetric) and positive definite matrix M , one may further decrease the number of critical RC steps roughly by twice [PS91] because we have

$$\|R_0\|_2 \leq 1 - \frac{1}{\sqrt{n\kappa(M)}} \quad \text{for } X_0 = I/\|M\|_F,$$

where $\|M\|_F = \text{trace}(M^*M)$ denotes the Frobenius norm of the matrix M . Furthermore, we may reach the bound $\|R_0\|_2 \leq 1 - 8\kappa_+ / (\kappa_+^2 + 6\kappa_+ + 1)$ by following [PS91] and choosing

$$(2.5) \quad X_0 = (8/\sigma)((\sigma_+ + \sigma_-)I - M), \quad \sigma = \sigma_+^2 + \sigma_-^2 + 6\sigma_+\sigma_-.$$

2.3. Scaled Newton's iteration. The choice of the scalars c_{i+1} in (2.1) was optimized in [PS91] in the case of RC processes (2.1) for $p = 2$ and X_0 in (2.4) or (2.5):

$$(2.6) \quad X_0 = c_0M^*, X_{i+1} = c_{i+1}(I + R_i)X_i = c_{i+1}(2X_i - X_iMX_i).$$

Under this choice, the number of critical steps decreases roughly by twice versus policy (2.3) and reaches the level of $i = \log_2 \kappa_+(M) + O(1/\kappa_+^2(M))$ for general M and roughly one half of that for a positive definite M . In other words, the impact of the optimal scaling is equivalent to increasing the order of convergence from 2 to 4.

Optimization of scaling as well as the initialization in (2.4) and (2.5) involve the lower and upper bounds σ_- and σ_+ on the singular values of M . Let ϵ denote the (relative) computer precision. An upper bound σ_+ can be obtained within relative errors of the order of ϵ based on the power or (better) Lanczos methods [GL96] applied to the matrices M^*M or MM^* . Application of the same methods to the matrices $\sigma_+I - M^*M$ or $\sigma_+I - MM^*$ yields σ_- with absolute errors of the order of $\epsilon\sigma_+$. These errors may exceed the smallest singular value of M but their impact on the initial residual norm remains within at most the order of ϵ . More precisely, under the cited policies of initializing and scaling from [PS91], the residuals are Hermitian or real symmetric matrices $I - C_{\sigma_-^2, \sigma_+^2}(MM^*)$ or (in the positive definite case) $I - C_{\sigma_-, \sigma_+}(M)$, where $C_{a,b}(y)$ is the scaled linear or (in the positive definite case) quadratic Chebyshev polynomial of the first kind in the range $[a, b]$. The residual norms are equal to $N_- = \max_{j=1, \dots, n} |1 - C_{\sigma_-^2, \sigma_+^2}(\sigma_j^2)|$ or (in the positive definite case) $H_- = \max_{j=1, \dots, n} |1 - C_{\sigma_-, \sigma_+}(\sigma_j)|$, where $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$, and $\sigma_1, \dots, \sigma_n$ denotes the singular values of M . The cited estimates for N_- and H_- from [PS91] hold if $\sigma_+ \geq \sigma_1$, $\sigma_n \geq \sigma_- > 0$.

Based on the power or Lanczos methods, we compute σ_- and σ_+ satisfying

$$(2.7) \quad (1 + \epsilon)\sigma_1 \geq \sigma_+ \geq \sigma_1, \quad \epsilon\sigma_1 \geq \sigma_- \geq \sigma_n.$$

Here we slightly abuse notation by keeping σ_- for a close upper bound on σ_n .

With these σ_+ and σ_- , the bounds N_- and H_- may increase at most to $N_+ = |1 - C_{\sigma_-^2, \sigma_+^2}(\sigma_n^2)|$ and $H_+ = |1 - C_{\sigma_-, \sigma_+}(\sigma_n)|$, respectively. Let us bound the initial residual norms in terms of $\kappa = \sigma_1/\sigma_n$, assuming (2.7). We have the bounds $\max\{N_-, N_+\}$ and $\max\{H_-, H_+\}$, where

$$N_+ = 1 - \frac{2\sigma_n^2}{\sigma_+^2 + \sigma_-^2}, \quad H_+ = 1 - \frac{8\sigma_n(\sigma_+ + \sigma_- - \sigma_n)}{\sigma_+^2 + \sigma_-^2 + 6\sigma_+\sigma_-}.$$

By applying (2.7), we deduce that

$$\begin{aligned} (\sigma_+^2 + \sigma_-^2)/\sigma_n^2 &\leq ((1 + \epsilon)^2 + \epsilon^2)\kappa^2 = (1 + 2\epsilon + 2\epsilon^2)\kappa^2, \\ \sigma_+ + \sigma_- - \sigma_n &\geq \sigma_1, \quad \sigma_+\sigma_-/\sigma_n^2 \leq (1 + \epsilon)\epsilon\kappa^2, \end{aligned}$$

and therefore,

$$N_+ \leq 1 - \frac{2}{(1 + 2\epsilon + 2\epsilon^2)\kappa^2}, \quad H_+ \leq 1 - \frac{8}{(1 + 8\epsilon + 8\epsilon^2)\kappa}.$$

In the RC processes with structured matrices, scaling is effective only initially because its effect is not compatible with the compression of structured matrices (see Section 8).

2.4. Bounding the precision of computing. It was proved in [PS91] that both scaled and unscaled Newton’s processes are numerically stable. Process (2.1), however, involves the expression

$$c_{i+1}\left(I + \sum_{k=0}^{p-1} R_i^k\right)X_i,$$

whose representation for a smaller residual norm $\|R_i\|$ requires the p -fold precision versus the single precision for representation of M and X_i . The precision growth in the RC process (2.1) can be avoided based on using *modular arithmetic in the real field* [P92b], [EPY98].

For the task of solving a linear system $M\mathbf{x} = \mathbf{b}$, the precision can be controlled if we apply the iterative improvement process

$$(2.8) \quad \mathbf{x}_1 = X_0\mathbf{b}, \quad \mathbf{r}_1 = \mathbf{b} - M\mathbf{x}_1,$$

$$(2.9) \quad \Delta_i = \mathbf{x}_{i+1} - \mathbf{x}_i = X_0\mathbf{r}_i, \quad \mathbf{r}_{i+1} = \mathbf{r}_i - M(\mathbf{x}_{i+1} - \mathbf{x}_i), \quad i = 1, \dots, s-1.$$

Unlike processes (2.1), the computations can be performed with a single/double precision where the output vector

$$(2.10) \quad \mathbf{x}_s = \mathbf{x}_1 + \sum_{i=1}^{s-1} \Delta_i$$

is represented by the sequence $\mathbf{x}, \Delta_1, \dots, \Delta_{s-1}$. This process involves neither operations with residual matrices R_i nor higher precision approximations X_i to M^{-1} . It converges linearly; the approximation error norm $\|\mathbf{x} - \mathbf{x}_{i-1}\|$ decreases by the factor of $\|R_0\| = \|I - X_0M\|$ in each iteration step because

$$\mathbf{x} - \mathbf{x}_i = (I - X_0M)(\mathbf{x} - \mathbf{x}_{i-1}) = (I - X_0M)^i(\mathbf{x} - \mathbf{x}_0).$$

3. A HOMOTOPIC RESIDUAL CORRECTION (HRC) ALGORITHM FOR A POSITIVE DEFINITE MATRIX

A reliable solution of the initialization problem for the RC processes is given by *homotopic RC processes*. We referred to them as *HRC processes* and study them next for a positive definite input matrix M , with $\text{spectrum}(M) = \{\lambda_1, \dots, \lambda_n\}$, where for some pair of λ_1^+ and λ_n^- we have

$$(3.1) \quad \lambda_1^+ \geq \lambda_1 = \|M\|_2 \geq \lambda_2 \geq \dots \geq \lambda_n \geq \lambda_n^- > 0.$$

Algorithm 3.1. A homotopic RC process for a positive definite matrix.

Input: an $n \times n$ Hermitian positive definite matrix, a nonnegative ϵ , a positive λ_1^+ satisfying (3.1), and a (scaled or unscaled) black box RC process (2.1), with a fixed stopping criterion.

Initialization: Fix some values $\nu_h > 0$ and θ_h , $0 < \theta_h < 1$, $h = 0, 1, \dots$, and write

$$(3.2) \quad M_0 = M + t_0 I, \quad t_0 = \lambda_1^+ / \theta_0, \quad X_0 = t_0^{-1} I,$$

$$(3.3) \quad M_{h+1} = t_{h+1} I + M = M_h - \Delta_h I, \quad \Delta_h = t_h - t_{h+1} > 0, \quad h = 0, 1, \dots$$

Apply the selected black box RC process (2.1) for $M = M_0$ and X_0 chosen as in (3.2). Stop where an approximation \tilde{X}_0 to M_0^{-1} is computed such that

$$\|R(M_0, \tilde{X}_0)\|_2 \leq \nu_0.$$

Computations: Stage h , $h = 0, 1, \dots$. Compute a lower bound η_h on the value

$$(3.4) \quad \|M_h^{-1}\|_2^{-1} = t_h + \lambda_n$$

(see Remark 3.1). Compute

$$(3.5) \quad \Delta_h = \theta_h \eta_h.$$

If $t_{h+1} > 0$, apply the black box RC process (with $X_0 = \tilde{X}_h$ and M replaced by M_{h+1}) to compute a matrix \tilde{X}_{h+1} such that

$$(3.6) \quad \|R(M_{h+1}, \tilde{X}_{h+1})\|_2 \leq \nu_{h+1}.$$

If $t_{h+1} \leq 0$, write $H = h + 1$, $X_0 = \tilde{X}_h$, $\nu_H = \epsilon$, use M instead of M_{h+1} , and apply the RC process to compute a matrix \tilde{X}_{h+1} satisfying (3.6) for $h + 1 = H$, $M_H = M$, $\nu_H = \epsilon$.

Output: The matrix \tilde{X}_H (approximating M^{-1}).

The algorithm is completely defined as soon as we fix an RC process (2.1) (including its stopping criterion (3.6) and the values $\nu_1, \dots, \nu_{H-1}, \nu_H = \epsilon$), the values $\theta_0, \dots, \theta_{H-1}$, and an algorithm for approximating $\|M_h^{-1}\|_2$ from above (see Remark 3.1).

Specific choices of the bounds $\nu = \nu_h$ can be guided by the following simple estimate.

Proposition 3.1. Let $\|I - \tilde{X}_h M_h\| \leq \nu_h$, $\|I - M_h^{-1} M_{h+1}\| \leq \theta_h$ for any fixed matrix norm. Then

$$\|I - \tilde{X}_h M_{h+1}\| \leq (1 + \nu_h) \theta_h + \nu_h.$$

Proof.

$$\begin{aligned} \|I - \tilde{X}_h M_{h+1}\| &\leq \nu_h + \|\tilde{X}_h M_h - \tilde{X}_h M_{h+1}\| \\ &\leq \nu_h + \|\tilde{X}_h M_h\| \|I - M_h^{-1} M_{h+1}\| \\ &\leq \nu_h + (1 + \nu_h) \theta_h. \quad \square \end{aligned}$$

Let us show the *correctness* of Algorithm 3.1. For the scalar t_0 , the matrix M_0 of (3.2) and the residual R_0 of (2.2), we have

$$R(M_0, t_0^{-1} I) = I - t_0^{-1} M_0 = -t_0^{-1} M,$$

$$(3.7) \quad r_0 = \|R(M_0, t_0^{-1} I)\|_2 = \lambda_1 / t_0 \leq \theta_0.$$

Further, deduce from (3.3) that

$$(3.8) \quad \begin{aligned} R(M_{h+1}, M_h^{-1}) &= \Delta_h M_h^{-1}, \\ r_{h+1} &= \|R(M_{h+1}, M_h^{-1})\|_2 = \Delta_h \|M_h^{-1}\|_2 = \Delta_h / (t_h + \lambda_n). \end{aligned}$$

Deduce from the bound $\eta_h \leq 1/\|M_h^{-1}\|_2$ and (3.4) that

$$(3.9) \quad \eta_h - t_h \leq 1/\|M_h^{-1}\|_2 - t_h = \lambda_n.$$

Finally, (3.5), (3.8) and (3.9) together imply that

$$(3.10) \quad r_{h+1} \leq \theta_h \text{ for all } h.$$

Remark 3.1. We can obtain a good lower bound η_h on $t_h + \lambda_n$ in (3.4) for $h < H$ by applying the power or Lanczos methods to yield a tight upper bound on $\|\tilde{X}_h\|_2$. Combine the equations (3.4), (3.6), and $M_h^{-1} = \tilde{X}_h + R(M_h, \tilde{X}_h)M_h^{-1}$ to deduce that $\|\tilde{X}_h\|_2/(1 + \nu_h) \leq \|M_h^{-1}\|_2 \leq \|\tilde{X}_h\|_2/(1 - \nu_h)$. Therefore we may choose

$$(3.11) \quad \eta_h = \|\tilde{X}_h\|_2^{-1}(1 - \nu_h) \leq \|M_h^{-1}\|_2^{-1} \leq \|\tilde{X}_h\|_2^{-1}(1 + \nu_h).$$

Furthermore, (3.9) implies that $\|M_h^{-1}\|_2^{-1} = \|M_i^{-1}\|_2^{-1} + t_h - t_i$ for all $i > 0$. Therefore, for smaller $h > 1$, we may alternatively choose

$$\eta_h = \max_{i < h} (\eta_i + t_h - t_i)$$

and avoid estimating $\|M_h^{-1}\|_2^{-1}$. We may repeat this trick periodically. In all cases, as an advantage versus the accelerated nonhomotopic RC processes, we do not have to estimate the smallest singular value λ_n of M except for the acceleration by scaling (see Remark 3.4). Equation (3.11) implies that

$$\|\tilde{X}_h\|_2/(1 + \nu_h) - t_h \leq \lambda_n \leq \|\tilde{X}_h\|_2/(1 - \nu_h) - t_h,$$

which enables tight bounds on λ_n provided ν_h is small and the norm $\|\tilde{X}_h\|_2$ is computed accurately.

Remark 3.2. The homotopic process of (3.2), (3.3) has the trajectory $M(t) = M + tI$ which for $t > 0$ is better conditioned than the input matrix M ; that is, one may easily verify that

$$(3.12) \quad \kappa(M(t)) \leq \kappa(M) \text{ for } t \leq 0.$$

The same inequality can be easily verified for the modification of the homotopic process in Section 6 in the indefinite Hermitian case.

Remark 3.3. The approach allows variations. For instance, instead of process (3.2), (3.3), we may apply the homotopic process $M_h = (1 - t_h)M + t_h M_0$, $t_0 = 1$, $t_h \rightarrow 0$, $h = 0, 1, \dots$, or the dual process

$$M_{h+1} = I + t_{h+1}M = M_h + (t_{h+1} - t_h)M, \quad t_0 = 1, \quad h = 0, 1, \dots,$$

where t_h grows large and at the end a single step (3.3) or a few steps (3.3) are applied. The resulting computations can be analyzed similarly to process (3.3).

Remark 3.4. To accelerate the RC processes by scaling, we use the initial approximation X_0 in (2.4) or (2.5). This is distinct from the \hat{X}_0 in Algorithm 3.1, but one may still yield this acceleration, by computing $M_{h+1}^{-1} = M_h^{-1}P_h^{-1}$ for $P_h = M_{h+1}M_h^{-1}$. Using the matrices $\hat{X}_0 = aP_h^*$ or $\hat{X}_0 = aI + bP_h$ for appropriate scalars a and b (defined similarly to (2.4), (2.5) for P_h replacing M) supports the

desired acceleration by scaling (see [Pa]). In particular, the bounds in Table 6.1 for the HRC processes include this acceleration. Computation of the scalars a and b above requires approximation of the smallest singular values of the matrices P_h . This is not a problem initially, for smaller h , but generally becomes harder for larger h , and then one may choose to shift to the slower unscaled RC processes with $X_0 = P^*/(\|P\|_1\|P\|_\infty)$ or $X_0 = I/\|P\|_F$ or to extend the remedies as at the end of subsection 2.3.

4. THE NUMBER OF HOMOTOPIC STEPS

To simplify our analysis, we next assume that the values η_h are defined by (3.11). Then, by virtue of (3.3)–(3.5), (3.9), and Remark 3.1, we have

$$\begin{aligned} t_{h+1} + \lambda_n &= t_h + \lambda_n - \Delta_h = t_h + \lambda_n - \theta_h \eta_h \leq (1 - \theta_h(1 - \nu_h))(t_h + \lambda_n), \\ &h = 0, 1, \dots, H - 1. \end{aligned}$$

Therefore,

$$\begin{aligned} t_{h+1} + \lambda_n &\leq (t_0 + \lambda_n) \prod_{i=0}^h (1 - \theta_i(1 - \nu_i)), \quad h = 0, 1, \dots, H - 1, \\ t_H &\leq 0 \text{ if } \lambda_n \geq (t_0 + \lambda_n) \prod_{h=0}^{H-1} (1 - \theta_h(1 - \nu_h)). \end{aligned}$$

To simplify our analysis further, assume ν_h and θ_h invariant in h ; that is, let $\nu_h = \nu$ and $\theta_h = \theta > \nu$ for all h . Substitute $t_0 = \lambda_1^+/\theta$ of (3.2) and rewrite the latter inequality as

$$\frac{1}{(1 - \theta(1 - \nu))^H} \geq \lambda_1^+ / (\theta \lambda_n) + 1, \quad H \geq \frac{-\log(1 + \lambda_1^+ / (\theta \lambda_n))}{\log(1 - \theta(1 - \nu))}.$$

Therefore, H homotopic steps are sufficient for the minimum integer H satisfying this bound; that is,

$$(4.1) \quad H = \left\lceil \frac{\log(1 + \lambda_1^+ / (\theta \lambda_n))}{\log(1 / (1 - \theta(1 - \nu)))} \right\rceil.$$

5. THE OVERALL NUMBER OF THE RESIDUAL CORRECTION STEPS

At each homotopic step, the number of RC steps depends on the bound θ on the initial residual norm (we assume it is invariant in all homotopic steps), the order q of convergence of the selected RC process, and the stopping criterion for this process. For simplicity, we assume a fixed order q for each process (2.1) with fixed unstructured matrix M and scalars p and c_{i+1} , $i = 0, 1, \dots$, ignoring possible variation in the transition from the scaled to the unscaled RC processes (cf. Remark 3.4). In particular, $q = p$ for unscaled processes (2.1) and (2.3).

Recall our stopping criterion (3.6), assume that $\nu = \nu_h$ for a fixed $\nu < \theta$, and estimate that we need

$$(5.1) \quad c = c(q, \theta, \nu) = \lceil \log_q \log_{\tilde{\theta}} \nu \rceil$$

RC steps at the h -th homotopic step for $h < H$ and

$$c(q, \theta, \epsilon) = \lceil (\log_q \log_{\tilde{\theta}} \epsilon) \rceil$$

RC steps at the (last) H -th homotopic step. Here

$$(5.2) \quad \tilde{\theta} = \theta + (1 + \theta)\nu$$

is chosen based on Proposition 3.1, and ϵ bounds the output residual norm. Overall we use at most

$$(5.3) \quad P = (H - 1)c + \lceil \log_q \log_{\tilde{\theta}} \epsilon \rceil$$

RC steps. Since $\log_q \log_{\tilde{\theta}} \epsilon = \log_q \log_{\tilde{\theta}} \nu + \log_q \log_{\nu} \epsilon$, we obtain that P or $P + 1$ equals $cH + \lceil \log_q \log_{\nu} \epsilon \rceil$. It remains to choose θ and ν to minimize P , for a fixed $\epsilon > 0$.

We immediately observe that the bound in (4.1) on the number of homotopic steps H grows at least proportionally to $1/\theta$ as $\theta \rightarrow 0$, whereas c in (5.1) decreases proportionally to $\log(1/\theta)$. This suggests choosing larger values of θ .

Let us assume that $x = 1 - \theta > 0$ is small, y is a parameter of our choice, $0 < y < 1$, and that we write $\nu = xy/(1 + \theta) = xy/(2 - x)$. Then we have

$$\begin{aligned} \tilde{\theta} &= 1 - (1 - y)x, \quad \ln(1/\tilde{\theta}) \approx (1 - y)x, \\ \ln(1/\nu) &= \ln((2 - x)/(xy)), \\ c &\approx \log_q \left(\left(\ln \frac{2 - x}{xy} \right) / ((1 - y)x) \right) = \log_q \frac{1}{x} + \log_q \left(\frac{1}{1 - y} \ln \frac{2 - x}{xy} \right). \end{aligned}$$

Substitute $y = 1/4$ (say) and then obtain that

$$c \approx \log_q \left(\frac{1}{x} \left(\ln \frac{4}{3} \right) \left(\ln \frac{8}{x} \right) \right),$$

whereas for smaller x and H in (4.1), we have

$$H \approx \log_q(1 + \lambda_1^+/\lambda_n) / \log_q(1/x).$$

Therefore,

$$P \approx cH + \lceil \log_q \log_{\nu} \epsilon \rceil \approx \log_q(1 + \lambda_1^+/\lambda_n) + \lceil \log_q \log_{\nu} \epsilon \rceil.$$

Assume that the latter term is small and obtain that

$$P \approx \log_q(1 + \lambda_1^+/\lambda_n).$$

This bound matches the estimate for nonhomotopic processes but now it is supported without pre-estimation of λ_n^- . The bound is θ -independent; that is, the overall number P of RC steps changes little as θ varies near 1.

Let us also try to optimize the choice of y for θ near 1 and $0 < x = 1 - \theta$. In this case, we minimize c in (5.1) by choosing $z = 1/y$ such that

$$(5.4) \quad 0 < z - 1 = \ln \left(\frac{2 - x}{x} z \right).$$

In Table 5.1, in the columns marked by T, we show the variation of the value P in (5.3) depending on θ and $\kappa = \lambda_1/\lambda_n$ provided that we have c in (5.1), $\tilde{\theta}$ in (5.2), H in (4.1), $\epsilon = 10^{-9}$, $\nu = (x/z)(2 - x)$, z satisfying (5.4), for $\theta > 0.7$; $\nu = \theta/10$, for $\theta \leq 0.7$, and $q = 2$ (for $q = 4$, P decreases roughly by 50% according to (5.1) and (5.3)). In the columns of Table 5.1 marked by E, we show the values P optimized experimentally based on the following platform:

- Operating System: Microsoft Windows XP Professional Version 2002 Service Pack 1
- Programming Environment: MATLAB Version 5.1.0.421
- Processor: Pentium(R) 4, 2.00 GHz
- System Memory: 1 GB

TABLE 5.1. The overall number P of RC steps for values of $\theta < 1$ and integral values of $\log_{10} \kappa \leq 10$ (Theoretical values of P vs. Experimental values of P).

θ	0.2		0.4		0.6		0.8		0.9		0.95		0.99	
$\log_{10} \kappa$	T	E	T	E	T	E	T	E	T	E	T	E	T	E
1	39	41	24	19	16	17	13	16	15	17	10	16	13	19
2	59	62	36	28	22	25	23	22	21	23	18	22	23	24
3	81	83	51	37	31	33	28	28	27	29	26	28	23	29
4	101	104	63	46	37	41	33	35	33	35	34	33	33	35
5	121	125	78	55	46	49	43	41	39	41	34	38	33	39
6	143	146	90	64	52	57	48	46	45	48	42	43	43	45
7	163	167	105	73	61	65	58	53	51	54	50	49	43	48
8	183	188	117	82	67	73	63	59	57	60	58	55	53	55
9	205	209	132	91	76	82	73	66	63	66	58	60	53	58
10	225	229	144	100	82	89	78	71	69	71	66	64	63	62

To define the input matrices M in these experiments, we first used a random number generator in MATLAB to generate random 100×100 real symmetric indefinite Toeplitz matrices A , then we computed their extremal eigenvalues (again by using MATLAB) and produced $M = A + aI$ for an appropriate scalar a to have M positive definite with a selected condition number. We ran 100 tests for each pair of θ and κ , κ denoting the condition number of the input matrix M .

Table 5.1 displays the average numbers of RC steps (rounded to integers) for each pair θ and κ . These values of P observed experimentally and displayed in Table 5.1 in the columns marked by E are reasonably close to our estimates for them which are displayed in the columns marked by T, taking into account the rounding errors, our stiff assumptions that the inequalities in (3.11) and Proposition 3.1 turn into equations, and our heuristic choice of ν , particularly of $\nu = \theta/10$ for $\theta \leq 0.7$.

Table 5.1 shows that the number P of RC steps in our tests was minimized for $\theta = 0.95$ and $\nu = 0.0039 \dots$ based on z from (5.4), except for the matrices having $\log_{10} \kappa \geq 9$. In the cases where $\log_{10} \kappa \geq 9$, P slightly decreased (by less than 4%) for $\theta = 0.99$.

6. INVERSION OF INDEFINITE HERMITIAN MATRICES

We may extend our HRC algorithm of Section 3 to the inversion of any nonsingular matrix M based on the equations

$$(6.1) \quad M^{-1} = M^*(MM^*)^{-1} = (M^*M)^{-1}M^*$$

because the matrices MM^* and M^*M are Hermitian (or real symmetric) and positive definite. This standard symmetrization, however, squares the condition number and, consequently, slows down the HRC algorithm. A known remedy is to reduce the inversion of M to the inversion of the indefinite Hermitian (or real symmetric)

matrix

$$(6.2) \quad N = \begin{pmatrix} 0 & M \\ M^* & 0 \end{pmatrix},$$

where

$$N^{-1} = \begin{pmatrix} 0 & (M^*)^{-1} \\ M^{-1} & 0 \end{pmatrix}, \quad \kappa(N) = \kappa(M).$$

Let us next extend our HRC algorithm to the inversion of Hermitian matrices.

Let λ^- and λ^+ be two fixed positive values such that

$$\lambda^- \leq |\lambda| \leq \lambda^+$$

for every eigenvalue λ of M . Then for any fixed sequence of real θ_h , $0 < \theta_h < 1$, $h = 0, 1, \dots$, we define an HRC process by (3.2)–(3.6), for η_h still denoting an upper bound on the norm $\|M_h^{-1}\|_2$ but with the matrix I replaced by the complex matrix $I\sqrt{-1}$. That is, our HRC algorithm (which can be applied to any Hermitian input matrix M) is now defined by the equations

$$(6.3) \quad M_0 = M + t_0 I\sqrt{-1}, \quad t_0 = \lambda^+/\theta_0,$$

$$(6.4) \quad X_0 = -t_0^{-1} I\sqrt{-1}$$

(replacing (3.2)), and

$$(6.5) \quad \begin{aligned} M_{h+1} &= t_{h+1} I\sqrt{-1} + M = M_h - \Delta_h I\sqrt{-1}, \\ \Delta_h &= t_h - t_{h+1} > 0, \quad h = 0, 1, \dots \end{aligned}$$

(replacing (3.3)). Equations (6.3)–(6.5) immediately imply bounds (3.7) on r_0 and (3.10) on r_{h+1} for $\eta_h \geq \|M_h^{-1}\|_2$ and Δ_h of (3.5), and we may easily extend (3.12) as well. Using the complex matrix M_0 complicates the computations if M is real symmetric but is painless for complex matrices M .

Let us extend our analysis presented in Sections 4 and 5. First note that the equation

$$\|M_h^{-1}\|_2^{-1} = ((t_h^2 + (\lambda^-)^2)^{1/2}) \quad \text{for all } h$$

replaces (3.4). Then again let us simplify the analysis, similarly to Sections 4 and 5. Assume for simplicity that $\eta_h = (t_h^2 + (\lambda^-)^2)^{1/2}$ (cf. Remark 3.1) and $\theta_h = \theta$ for all h . It follows that

$$t_{h+1} = t_h - \Delta_h = t_h - (t_h^2 + (\lambda^-)^2)^{1/2}\theta < t_h - \theta \max\{t_h, \lambda^-\}, \quad h = 0, 1, \dots$$

Therefore, $t_{h+1} < 0$, where $(1 - \theta)^h t_0 \leq \theta \lambda^-$. Substitute $t_0 = \lambda^+/\theta$ and obtain that $t_H \leq 0$, where

$$(6.6) \quad H - 1 = \left\lceil \frac{\log(\lambda^+ / (\theta^2 \lambda^-))}{\log(1/(1 - \theta))} \right\rceil.$$

The latter bound is within the term $\gamma = 1 + \lceil (\log(1/\theta)) / \log(1/(1 - \theta)) \rceil$ from bound (4.1) for $\lambda_1^+ = \lambda^+$ and $\lambda_n^- = \lambda^-$. This term is at most 2 for $\theta \geq 1/2$. On the other hand, our estimates in Section 5 for the numbers of critical and refinement steps performed in each homotopic step remain unchanged (these estimates are completely defined by the parameters ϵ, ν , and θ). Therefore, up to replacing λ_n^- by λ^- and λ_1^+ by λ^+ and performing at most $a = \gamma \lceil \log_q((\log \nu) / \log \theta) \rceil$ additional RC steps, the estimates of Sections 4 and 5 apply to the Hermitian indefinite case as well. The latter bound a is relatively small, and we ignore it in Table 6.1, which summarizes our estimates for the overall numbers of RC steps in the HRC

TABLE 6.1. Numbers of RC steps required for numerical inversion of Hermitian matrices M for $q = 4$, ρ in Section 2.2, and $\theta \rightarrow 1$.

M	RC processes	HRC processes
indefinite $\kappa_+(M) = \lambda^+/\lambda^-$	$\log_2 \kappa_+(M) + \rho + O(1)$	$0.5 \log_2 \kappa_+(M) + O(1)$
positive definite $\kappa_+(M) = \lambda_1^+/\lambda_n^-$	$0.5 \log_2 \kappa_+(M) + \rho + O(1)$	$0.5 \log_2 \kappa_+(M) + O(1)$

processes and nonhomotopic RC processes applied to the same general Hermitian matrix M . We use the recipe of Remark 3.4 to incorporate the scaled RC processes in subsections 2.2 and 2.3 into our HRC processes. By Table 6.1, the HRC processes use roughly as many RC steps as nonhomotopic RC processes for the inversion of a Hermitian positive definite input matrix M and roughly by twice fewer RC steps where M is Hermitian indefinite.

7. A HOMOTOPIC RC PROCESS WITH A GENERALIZED INITIALIZATION RULE

Motivated by the applications to the inversion of structured matrices (see Section 6.9.3 in [P01a]), let us extend our homotopic processes and their analysis by allowing a more general choice of the initial matrix M_0 .

First assume that M and M_0 is any fixed pair of positive definite matrices, where M_0 is readily invertible, $\text{spectrum}(M_0) = \{\mu_1, \dots, \mu_n\}$,

$$(7.1) \quad \mu_1^+ \geq \mu_1 \geq \mu_2 \geq \dots \geq \mu_n \geq \mu_n^- > 0,$$

and the values μ_1^+ and μ_n^- are available. Now recursively define some scalars t_1, \dots, t_{H-1} and the matrices

$$(7.2) \quad M_{h+1} = t_{h+1}M_0 + M = M_h + (t_{h+1} - t_h)M_0, \quad h = 0, 1, \dots, H-1,$$

where $t_1 > t_2 > \dots > t_{H-1} > t_H = 0$.

One may rewrite (7.2) as $M_{h+1} = M_0(t_h I + M_0^{-1}M)$ and apply our previous study to the inversion of the indefinite matrix $M_0^{-1}M$. In our next alternative HRC process, we avoid shifting to this matrix directly. We deduce that

$$\|I - (t_1 M_0)^{-1}M_1\|_2 \leq \|M_0^{-1}M/t_1\|_2 \leq \|M_0^{-1}\|_2 \|M\|_2/t_1 \leq \lambda_1^+/(t_1 \mu_n^-),$$

for λ_1^+ of (3.1) and choose

$$(7.3) \quad t_1 = \lambda_1^+ / (\theta_0 \mu_n^-).$$

Then $\|I - (t_1 M_0)^{-1}M_1\|_2 \leq \theta_0$. Invert M_1 by applying a process (2.1) for $X_0 = t_1 M_0$. Now deduce from (7.2) that

$$(7.4) \quad \begin{aligned} I - M_h^{-1}M_{h+1} &= (t_h - t_{h+1})M_h^{-1}M_0, \\ \|I - M_h^{-1}M_{h+1}\|_2 &\leq (t_h - t_{h+1})\|M_h^{-1}\|_2 \|M_0\|_2. \end{aligned}$$

Substitute the bound

$$\|M_0\|_2 \leq \mu_1^+$$

and obtain that $\|I - M_h^{-1}M_{h+1}\|_2 \leq \theta_h$ if $(t_h - t_{h+1})\mu_1^+ \|M_h^{-1}\|_2 \leq \theta_h$ or, equivalently, if $t_{h+1} \geq t_h - \theta_h/(\mu_1^+ \|M_h^{-1}\|_2)$. Recall that, clearly,

$$\|M_h^{-1}\|_2 \leq 1/(t_h\mu_n^- + \lambda_n^-)$$

for all h and for λ_n^- of (3.1) (see [Par80, p.191]), write

$$(7.5) \quad t_{h+1} = t_h - (t_h\mu_n^- + \lambda_n^-)\theta_h/\mu_1^+,$$

and deduce (3.10). Invert the matrices M_{h+1} by applying processes (2.1) for $X_0 = M_h^{-1}$ and for $h = 1, 2, \dots, H - 2$, until the value t_{h+1} becomes nonpositive for $h = H - 1$. Then at the last homotopic step, invert M instead of M_H .

Clearly, the estimates of Section 5 for the number of RC steps at each homotopic step apply to the above generalized HRC process as well.

Let us next estimate the number of homotopic steps H , in terms of the parameters $t_1, \theta_h, \kappa^+ = \mu_1^+/\mu_n^-$, and the lower bounds λ_n^- and μ_n^- on the eigenvalues of the matrices M and M_0 . Substitute the expression $\kappa^+ = \mu_1^+/\mu_n^-$ into (7.5) for $h = 0, 1, \dots, H - 1$ and obtain that

$$(7.6) \quad \begin{aligned} t_{h+1} &= t_h(1 - \theta_h/\kappa^+) - \theta_h\lambda_n^-/\mu_1^+, \\ t_{h+1} + \kappa^+\lambda_n^-/\mu_n^- &= (t_h + \kappa^+\lambda_n^-/\mu_1^+)(1 - \theta_h/\kappa^+) \\ &= (t_1 + \kappa^+\lambda_n^-/\mu_n^-) \prod_{i=0}^h (1 - \theta_i/\kappa^+). \end{aligned}$$

Therefore, we have $t_{h+1} \leq 0$ if

$$(t_1 + \kappa^+\lambda_n^-/\mu_n^-) \prod_{i=0}^h (1 - \theta_i/\kappa^+) \geq \kappa^+\lambda_n^-/\mu_n^-;$$

that is, if

$$1 + t_1\mu_n^-/(\lambda_n^-\kappa^+) \geq 1/\prod_{i=0}^h (1 - \theta_i/\kappa^+).$$

Assuming that $\theta_h = \theta$ is invariant in h , we arrive at $t_H \leq 0$ for

$$(7.7) \quad H = 1 + \left\lceil \frac{(\log(1 + t_1\mu_n^-/(\lambda_n^-\kappa^+)))}{(\log(1 - \theta/\kappa^+))^{-1}} \right\rceil$$

and t_1 of (7.3).

Finally, if M is any nonsingular matrix, we may apply symmetrization recipes (6.1) or (6.2) to extend algorithm of this section. In particular, recipe (6.2) reduces the problem to the case where M is an indefinite Hermitian (or real symmetric) matrix. Then we may extend HRC process (7.2)–(7.5) where we keep equations (7.2)–(7.3), choose the matrix M_0 equal to $\hat{M}\sqrt{-1}$ for a fixed positive definite matrix \hat{M} , and modify (7.4)–(7.5) to ensure that $\|I - M_h^{-1}M_{h+1}\|_2 \leq \theta_h$ for all h .

Let us complete the description of this extended homotopic process. Assume that bounds (7.1) still hold where $\{\mu_1, \dots, \mu_n\} = \text{spectrum}(\hat{M})$ and each eigenvalue λ of the input matrix M satisfies the bounds

$$(7.8) \quad 0 < \lambda^- \leq |\lambda| \leq \lambda^+$$

for two fixed positive values λ^- and λ^+ . Now write

$$(7.9) \quad t_{h+1} = t_h - (\theta_h/\mu_1^+)((\lambda^-/\kappa^+)^2 + (t_h\mu_n^-)^2)^{1/2}, \quad \kappa^+ = \mu_1^+/\mu_n^-,$$

where $h = 0, 1, \dots, H - 1$.

Let us deduce bounds (3.10). Recall the following well-known theorem (see [Par80, proof of Theorem 15-3-3]).

Theorem 7.1. *Let M and \hat{M} be two Hermitian matrices. Let the matrix \hat{M} be positive definite, such that*

$$(7.10) \quad \hat{M} = U\Sigma^2U^*$$

for a unitary matrix U , $U^*U = UU^* = I_n$, and a diagonal matrix

$$\Sigma = \text{diag}(\sigma_i)_{i=1}^n, \quad \mu_1^+ \geq \sigma_1^2 \geq \sigma_2^2 \geq \dots \geq \sigma_n^2 \geq \mu_n^- > 0.$$

Then there exists a unitary matrix V , $V^*V = VV^* = I_n$, such that

$$(7.11) \quad D = V^*\Sigma^{-1}U^*MU\Sigma^{-1}V$$

is a real diagonal matrix.

Corollary 7.1. *Under the notation of (7.1), (7.8), and Theorem 7.1, we have*

$$\|M_h^{-1}\|_2^2 \leq (\mu_n^-)^{-2}((\lambda^-/\mu_n^+)^2 + t_h^2)^{-1} = ((\lambda^-/\kappa^+)^2 + (t_h\mu_n^-)^2)^{-1}$$

for $h = 1, 2, \dots$ where $\kappa^+ = \mu_1^+/\mu_n^-$.

Proof. By combining (7.10) and (7.11), we obtain that

$$M_h = M + t_h\sqrt{-1}\hat{M} = U\Sigma V(D + t_hI\sqrt{-1})V^*\Sigma U^*,$$

$$M_h^{-1} = U\Sigma^{-1}V(D + t_hI\sqrt{-1})^{-1}V^*\Sigma^{-1}U^*.$$

Therefore,

$$\|M_h^{-1}\|_2 \leq \|\Sigma^{-2}\|_2 \|(D + t_hI\sqrt{-1})^{-1}\|_2 \leq \left(\frac{1}{\mu_n^-}\right)^2 \left(\frac{1}{\|D^{-1}\|_2^2} + t_h^2\right)^{-0.5}.$$

On the other hand, we deduce from (7.1), (7.8), and (7.11) that

$$\|D^{-1}\|_2 \leq \|\Sigma^2\|_2 \|M^{-1}\|_2 \leq \mu_1^+/\lambda^-.$$

By substituting the latter bound into our estimate for the norm $\|M_h^{-1}\|_2$, we obtain

$$\|M_h^{-1}\|_2^2 \leq (\mu_n^-)^{-2}((\lambda^-/\mu_1^+)^2 + t_h^2)^{-1} = ((\lambda^-/\kappa^+)^2 + (t_h\mu_n^-)^2)^{-1}. \quad \square$$

Relations (7.1), (7.2), (7.4), (7.9), and Corollary 7.1 together immediately imply (3.10). Let us compare the estimate of Corollary 7.1 and the bound $\|M_h^{-1}\|_2 \leq 1/(t_h\mu_n^- + \lambda_n^-)$. The two estimates are close to one another provided that the terms λ_n^- and λ^-/κ^+ are dominated by the term $t_h\mu_n^-$. If the term λ^-/κ^+ dominates, the bound of Corollary 7.1 may be larger by roughly the factor of $\kappa^+\lambda_n^-/\lambda^-$.

Equation (7.9) implies the crude bounds

$$t_{h+1} \leq t_h - (\theta_h/\mu_1^+)(\lambda^-/\kappa^+ + t_h\mu_n^-), \quad h = 1, 2, \dots$$

Consequently,

$$t_{h+1} + \lambda^-/\mu_1^+ \leq (1 - \theta_h/\kappa^+)(t_h + \lambda^-/\mu_1^+) \leq \dots \leq (t_1 + \lambda^-/\mu_1^+) \prod_{i=1}^h (1 - \theta_i/\kappa^+).$$

The latter inequality implies that the value t_H is nonpositive for

$$H \leq 1 + \lceil (\log(1 + t_1\mu_1^+/\lambda^-))/\log(1 - \theta/\kappa^+)^{-1} \rceil,$$

provided that $\theta_h = \theta$ for all h .

8. SPECIALIZATION TO STRUCTURED MATRICES

8.1. Fast computations with structured matrices. The power of RC and HRC processes increases dramatically where M and X_0 are structured matrices (such as Toeplitz, Hankel, Vandermonde, Cauchy, and Pick matrices or the matrices with structures of similar type). Such matrices are represented in *compressed form*, with their displacements $L(M)$, where L is a linear displacement operator associated with a selected class of structured matrices (see, e.g., (8.1)). For each class, the associated displacement operators L ensure that $\text{dr}(M) = \text{rank}(L(M))$ is small, and thus the displacements $L(M)$ can be represented with $(2n)\text{dr}(M)$ parameters rather than n^2 entries.

The *displacement rank* of M is denoted by $\text{dr}(M)$. This is the basic parameter in computations with structured matrices, introduced in [KKM79]. We have $\text{dr}(M) \leq 2$ for Toeplitz, Hankel, Sylvester, and Frobenius matrices M and appropriate operators L , while we have $\text{dr}(M) \leq 1$ for Vandermonde and Cauchy matrices M and their associated operators L , and similar bounds are known for many other classes of structured matrices [P01a]. The linear inverse operators L^{-1} define a simple transition back from $L(M)$ to M , which enables us to decompress $L(M)$ [P01a, Sections 4.3–4.5], [PW03].

To give an example, first choose two scalars e and $f \neq e$. Let \mathbf{e}_{i-1} denote the i -th column of the identity matrix I , $i = 1, \dots, n$, and write

$$Z_f = \begin{pmatrix} 0 & & & f \\ 1 & \ddots & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{pmatrix}, \quad J = \begin{pmatrix} & & & 1 \\ & \ddots & & \\ & & \ddots & \\ 1 & & & \end{pmatrix}, \quad T = (t_{i-j})_{i,j=0}^{n-1},$$

$$\mathbf{t} = (t_i)_{i=0}^{n-1}, \quad \mathbf{t}_- = (t_{-i})_{i=0}^{n-1}, \quad \mathbf{v} = (v_i)_{i=0}^{n-1}, \quad Z_e(\mathbf{v}) = \sum_{i=0}^{n-1} v_i Z_e^i.$$

Note that $Z_1\mathbf{v} = (v_{i-1})_{i=0}^{n-1}$, $J\mathbf{v} = (v_{h-1-i})_{i=0}^{n-1}$ for $v_{-1} = v_{n-1}$, that is Z_1 cyclically shifts the coordinates of the vector \mathbf{v} , whereas J reverses their order. Also observe that $Z_e(\mathbf{v})$ denotes the classes of circulant, skew-circulant, and upper triangular Toeplitz matrices for $e = 1$, $e = -1$, and $e = 0$, respectively.

Now we may compress a Toeplitz matrix $T = (t_{i-j})_{i,j=0}^{n-1}$ by representing it with its displacement $L(M)$ for the Sylvester displacement operator L such that

$$(8.1) \quad L(T) = Z_e T - T Z_f = \mathbf{g}_1 \mathbf{h}_1^T + \mathbf{g}_2 \mathbf{h}_2^T,$$

where

$$(8.2) \quad \mathbf{g}_2 = \mathbf{e}, \quad \mathbf{h}_2 = eJ\mathbf{t} - Z_0^T \mathbf{t}_-, \quad \mathbf{g}_1 = Z_0 J \mathbf{t}_- - f\mathbf{t}, \quad \mathbf{h}_1 = \mathbf{e}_{n-1}$$

(cf. [P01a, equation (4.2.1), page 120]). We may decompress T from $L(T)$ and, more generally, we may reconstruct a matrix M from its displacement

$L(M) = Z_e M - M Z_f = \sum_{j=1}^l \mathbf{g}_j \mathbf{h}_j^T$, $e \neq f$ as follows (cf. [P01a, Example 4.4.2 on page 126]),

$$(8.3) \quad (e - f)M = \sum_{j=1}^l Z_e^T(\mathbf{g}_j) Z_f(J\mathbf{h}_j).$$

If M is structured and nonsingular, then M^{-1} is also structured and can be recovered by decompressing its displacement. For example, if M is a nonsingular matrix, and $L_{A,B}(M) = AM - MB = \sum_{j=1}^l \mathbf{g}_j \mathbf{h}_j^T$, then

$$(8.4) \quad \begin{aligned} L_{B,A}(M^{-1}) &= BM^{-1} - M^{-1}A = -M^{-1}L_{A,B}(M)M^{-1} \\ &= \sum_{j=1}^l (-M^{-1}\mathbf{g}_j)(\mathbf{h}_j^T M^{-1}). \end{aligned}$$

8.2. Structured RC processes. We recall (see [P01a, Chapter 6]) that every RC step only requires linear memory space $O(\rho n)$ and $O(\rho^2 v)$ flops provided that $\text{dr}(X_0) \leq \rho = \text{dr}(M^{-1})$, the structure is preserved for the computed approximations X_i to M^{-1} as i increases, and

$$(8.5) \quad v = v_{M, X_0}$$

flops are sufficient to multiply by a vector a basic $n \times n$ matrix of the class represented by M and X_0 . (For Toeplitz and Hankel matrices M and X_0 we have $v = O(n \log n)$, for Vandermonde and Cauchy matrices $v = O(n \log^2 n)$.)

The reader is referred to [P01a, Chapter 6], [PRW02], [PVBWC04], and the bibliographies therein on three compression techniques that preserve the structure and on the analysis of the resulting RC and HRC processes. In particular, the first and best known approach to the compression (proposed in [P92] and [P93]) is to truncate the smallest singular values of the displacement of the computed approximation X_k to M^{-1} and to continue the iteration with the resulting matrix \tilde{X}_k instead of X_k . This policy is hereafter referred to as TSSV.

The number s of untruncated singular values of the displacement is the *parameter of compression*. We may fix s explicitly (e.g., $s = 2$ at the final RC steps for the inversion of a Toeplitz matrix) or control the truncation numerically by keeping only the singular values which exceed $\epsilon \sigma_1$ for a fixed small positive ϵ (which we may keep invariant or decrease as the residual norm decreases) and σ_1 denoting the largest singular values of the displacement.

We have to account for the potential side effect of the compression. The TSSV as well as other compression policies may move X_k away from M^{-1} . How much can this increase the error and residual norms? According to [P92], [P93], [P01a], [PRW02], the increase is by the factors f and F , respectively, such that

$$(8.6) \quad f = O(\|L^{-1}\|),$$

$$(8.7) \quad F = O(\|L^{-1}\| \kappa(M))$$

where $\|L^{-1}\|$ denotes the norm of the inverse displacement operator. For the operators L associated with the Toeplitz/Hankel structure, we have $\|L^{-1}\| = O(n)$ (see the estimates for these and other inverse displacement operators L^{-1} in [P01a, Section 6.4], [PW03]).

Now assume an RC process with the TSSV having the order q of convergence. Let r_k denote the residual norm in the k -th RC step (with the TSSV) and write $\tilde{r}_k = r_k F^{1/(q-1)}$. Then we have $\tilde{r}_k \leq \tilde{r}_{k-1}^q, k = 1, 2, \dots$, and easily deduce that $\tilde{r}_k = \tilde{r}_0^{q^k}, k = 1, 2, \dots$. To keep the convergence order q , it is sufficient to compute an initial approximation X_0 such that

$$(8.8) \quad \tilde{r}_0 = r_0 F^{1/(q-1)} \leq \gamma < 1,$$

for a constant γ . It is not easy to guarantee this bound without using homotopic processes; e.g., the methods in subsection 2.2 are by far too weak to support (8.8) for F of the order of $n\kappa(M)$. Fortunately, (8.6) and (8.7) are only the upper but not the lower bounds. In our extensive experiments with the processes (2.1), (2.3) initialized by (2.4) or (2.5) and applied to Toeplitz input matrices M , we observed that the TSSV policy moves the computed approximations away from M^{-1} much less than such upper bounds would have implied (see the next section and [P01a, Tables 6.3-6.21]). In fact, in more than 20% of our tests for these processes, the compression substantially improved the approximations at some RC steps, forcing divergent processes to converge [P01a, Table 6.21]. We have no theoretical explanation for this experimental observation and just call this effect the *autocorrection in compression* (cf. [PVBWC04]). This effect is peculiar to processes (2.1), (2.3) with $X_0 = cM^*$ or $X_0 = cI$ (in the positive definite case) for appropriate scalars c and apparently is not compatible with the scaling in subsection 2.3. For homotopic processes, the latter initialization requires using the modification in [Pa] (see Remark 3.4).

The level of truncation, measured by the numbers s_k of untruncated singular values in the displacement $L(\tilde{X}_k)$, can be chosen anywhere from $\text{dr}(M^{-1})$ and up. The smaller s_k , the more structured \tilde{X}_k and the simpler to operate with it, but the farther $L(\tilde{X}_k)$ from $L(X_k)$ and possibly (although not necessarily) the farther \tilde{X}_k from M^{-1} .

We recall that the number of flops involved in the k -th RC step grows proportionally to $s_k^2 v$ for v in (8.5). For a fixed class of $n \times n$ structured matrices M and X_0 and for a fixed n , we have v fixed and measure the overall time cost of an RC process by the sum $t = \sum_k s_k^2$ where the summation is over all k representing all RC steps (in all homotopic steps for the homotopic processes).

8.3. HRC processes for structured matrices. In [P92] the bound

$$(8.9) \quad t = O((\log^2 F) \log \kappa), \quad \kappa = \kappa(M),$$

was proved for a homotopic process assuming the compression factor F , which in the worst case has the order of $n\kappa$ (cf. (8.7)), so that $\log^2 F$ is in $O(\log^2(n\kappa))$. This implies the bound $vt = O(n \log^3 n)$ for $n \times n$ well-conditioned Toeplitz matrices M and $v = O(n \log n)$ in (8.5). Let us briefly recall the basic idea of the algorithm supporting (8.9). Every homotopic step h was initialized to achieve a residual norm $r_0 \leq \theta$ for a fixed constant $\theta < 1$, and the next $k = O(\log \log F)$ RC steps were performed with no compression. This decreased the residual norms sufficiently much to yield the desired bound (8.8), and the computations were continued (if necessary) with compression, until they converged to M_h^{-1} . Then the $(h + 1)$ -st homotopic step was initialized with $r_0 \leq \theta$. Observe that the linearly growing displacement rank of X_k stayed in $O(\log F)$ provided that $\text{dr}(X_0) + \text{dr}(M) = O(1)$. The cost bound proportional to s_k^2 at the k -th RC step and the bound $O(\log \kappa)$ on

the number of homotopic steps for a fixed constant θ (implied by (4.1)) together give us (8.9).

The delay of the compression in the above algorithm simplifies our analysis, but the earlier compression would enable us to decrease the overall computational cost $t = \sum_k s_k^2$. It is not easy to optimize the choice of the parameters s_k or ϵ_k , however. To simplify our task, we keep the assumption that the parameters θ_k and ν_k defined in Sections 3–5 remain invariant in k ; that is, $\theta_k = \theta$, $\nu_k = \nu$ for all k , and we also extend this assumption to s_k and ϵ_k . The task of the optimization of these parameters is still computationally harder than the approximation of the inverse M^{-1} , and our analysis in the previous sections does not apply under the compression policy. In particular, we have no general estimates for the impact of the compression on the residual norms, which would decrease the bound (8.9). Thus we proceed heuristically, by choosing the experimental values of the triple θ , ν , ϵ for which the values of the overall time cost t stay reasonably close to the optimum (see the next section).

Remark 8.1. The nonhomotopic RC processes may converge efficiently even to the inverses of ill-conditioned structured matrices if the effect of autocorrection in compression turns out to overcome rapidly the difficulty at the initial stage where r_0 is close to 1. To exploit this effect for the homotopic RC processes, we would have to shift to their modification in [Pa] and then to elaborate experimentally upon the choice of the parameters θ_k , ν_k , and s_k (or ϵ_k). We leave this challenge to our future work.

Remark 8.2. Let us recall a simple way of limited decrease of θ . This way is slow but completely preserves matrix structure and may be useful where even a minor decrease of θ is important. Namely, if the residual norm is already small enough, say, equals 3/4 or 1/2, then we may further improve the approximation to M^{-1} based on linearly convergent processes such as (2.8)–(2.10). These processes only require that we multiply M and X_0 by vectors and, therefore, we do not need compression. In a smaller number of iteration steps, they converge linearly to the solutions to the linear systems that define a (displacement generator for) compressed representation of M^{-1} [P01a], [PKRC02, Remark 3.2] and thus linearly decrease the residual norms. When the norm becomes small enough, one may more safely shift to RC processes (2.1) with compression.

9. NUMERICAL EXPERIMENTS WITH REAL SYMMETRIC TOEPLITZ MATRICES

The experiments have been performed in Brooklyn College and the Graduate Center of CUNY by M. Kunin (in collaboration with the other coauthors). For the tests we used the following computational facilities:

- Operating System: Redhat Linux 9
- Programming Environment: g++ (GCC) 3.3.2, with LAPACK 3.0
- Processor: Pentium(R) III (Coppermine), 1 GHz
- System Memory: 256 MB

We randomly generated the entries of the $n \times n$ real symmetric indefinite Toeplitz matrices by using the `drand42()` function from the GCC C library. We computed the matrix norms by applying the SVD subroutine from LAPACK to produce the condition numbers of these matrices. In other cases we relied on the power method to speed up the computations. In our limited applications this choice made no

dramatic affect on the accuracy of our tests. By adding the properly scaled identity matrix, we turned our random indefinite matrices into positive definite with fixed condition numbers. Tables 9.1–9.4 show the results of our experiments with such matrices, grouped according to their sizes $n \times n$ for $n = 32, 64, 128$. For each n we tested 100 matrices of the size $n \times n$ with the condition numbers in the range from 10 to 500,000.

In the experiments we applied the HRC processes (3.2)–(3.6) based on the RC processes (2.1), (2.3) for $p = 2$, that is, on Newton’s unscaled iteration.

We represented the matrices with their displacements by using the Sylvester displacement operators L : $L(A) = Z_{-1}A - AZ_1$ for $A = M$ and $A = M_h$ and $L(B) = Z_1B - BZ_{-1}$ for $B = X_0$, $B = \tilde{X}_h$, and $B = M_h^{-1}$, for all h (see (8.1)–(8.4)).

For the compression of the displacements we used the TSSV policy truncating the singular values below $\epsilon\sigma_1$ for a fixed positive ϵ . At the last homotopic steps we stopped the RC process where it converged. At other homotopic steps we stopped where the residual norm decreased below a fixed positive η (compare Section 3). We chose the step sizes t_h according to (3.2)–(3.6) for a fixed pair of η and θ .

For each matrix dimension n , we measured the overall time cost C of the inversion of a Toeplitz matrix by the sum of the squares of the $\text{dr}(\tilde{X}_h)$ over all h and all the RC steps involved (dropping the factor v in (8.5), which was constant for this HRC process and for a fixed dimension n). We let the three parameters θ , η , and ϵ be invariant in h and first optimized them experimentally, for every input matrix M as follows.

For each matrix M we applied our HRC process for each of 270 triples (θ, η, ϵ) where θ , η , and ϵ ranged in the sets $i/10, i = 1, \dots, 9$; $0.5, 0.2, 0.1, 0.01, 0.001, 0.0001$, and $0.5, 0.2, 0.1, 0.01, 0.001$, respectively, and selected a triple which minimized the overall time cost C . Table 9.1 displays the average values + standard deviations of the parameters θ , η , and ϵ in these triples and the cost C for each n . The average was taken over all 100 matrices M for each fixed n . Of course, the cost of the computation of the average optimal triple substantially exceeded the cost of the matrix inversion.

The HRC algorithm applied to the same matrices M and to the latter triples diverged in 20% of the cases for $n = 128$, in 6% of the cases for $n = 64$, and never for $n = 32$. In the cases of convergence, however, the overall time cost was always within 10% of its optimal value.

To avoid divergence, we reapplied the algorithm for a distinct triple θ, η, ϵ for each n . This triple was again the average of the optimal triples but over the more narrow set, obtained by removing from the original set of 270 triples all triples for which the algorithm diverged at least once. Table 9.2 shows these average triples, which we call safe. By replacing $\eta = 0.099$ by $\eta = 0.1$ for $n = 64$, we unified the triples for $n = 32, 64, 128$ and arrived at the single safe triple $(\theta, \eta, \epsilon) = (0.4, 0.1, 0.5)$. For each n and for every matrix M , we reapplied our algorithm to compute the cost values C for this triple and then calculated the average of these values (over 100 matrices M for each n) as well as the standard deviations. Table 9.3 shows these results. The overall time cost increased roughly by twice versus its optimal values in Table 9.1, but no divergence was observed in all 300 tests. This suggests the triple $(0.4, 0.1, 0.5)$ as the basic candidate choice for the users.

TABLE 9.1. Optimal time C and respective triples (θ, η, ϵ) (average over 100 matrices for each dimension n).

n	Time	θ	η	ϵ
32	180.290+24.462	0.812+0.038	0.149+0.060	0.496+0.040
64	204.150+29.426	0.812+0.041	0.159+0.058	0.484+0.079
128	231.570+41.459	0.817+0.053	0.158+0.055	0.483+0.084

TABLE 9.2. Optimal safe triples (average over 100 matrices for each n).

n	θ	η	ϵ
32	0.400+0.000	0.100+0.000	0.500+0.000
64	0.400+0.000	0.099+0.009	0.500+0.000
128	0.400+0.000	0.100+0.000	0.500+0.000

TABLE 9.3. Average time C for the optimal safe triple.

n	Time	Tests	Divergences	θ	η	ϵ
32	361.200+53.685	100	0	0.400	0.100	0.500
64	408.120+66.563	100	0	0.400	0.100	0.500
128	447.880+59.986	100	0	0.400	0.100	0.500

TABLE 9.4. Average time C for the optimal and reasonably safe triples.

n	Time	Tests	Divergences	θ	η	ϵ
32	227.758+35.354	100	1	0.610	0.162	0.471
64	262.280+40.307	100	0	0.600	0.161	0.500
128	289.143+40.500	100	2	0.603	0.153	0.490

When users run our algorithm for various input matrices of various sizes, they perhaps may still occasionally encounter divergence. In this rare case, the simplest recipe is to reapply the algorithm with, say, $\eta = 0.001$, $\epsilon = 0.1$ and with θ halved recursively until convergence. (In the similar experiments reported in [P01a, Section 6.11], more than two recursive steps were very rarely required and the changes in ϵ and η never seriously affected the overall time cost.)

One may achieve convergence at the expense of a smaller increase of the time cost by applying more refined policies of the recursive decrease of θ and η . Conversely, the user may begin with the values θ and η lying between those in Tables 9.1 and 9.3 and recursively decrease them in case of divergence. We expect that the convergence and the time cost depend most on θ and least on ϵ . As a rule of thumb in the variations of θ and η , one may choose adding (or subtracting) 0.08 to (or from) θ and 0.01 to (or from) η . As some guidance for the users, we include Table 9.4, which shows the average values of the overall time cost + standard deviations for three triples (θ, η, ϵ) , (one for each n), which we selected as the average optimal triples excluding from the same 270 triples those for which divergence occurred in more than 5% of the cases. We call the resulting triples reasonably safe.

It is interesting that in our experiments the overall time cost $\sum_h \text{dr}^2(\tilde{X}_h)$ (scaled by deleting the factor v in (8.5)) little changed with the increase of the matrix size. As expected, it grew proportionally to $\log \kappa(M)$, when we classified our data according to both dimension and condition numbers of the input matrices M .

10. EXTENSION TO APPROXIMATING
THE MOORE–PENROSE GENERALIZED INVERSE

Let $M = U\Sigma V^*$ be the SVD of an $n \times n$ matrix M , $U^*U = V^*V = I$, $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0)$, $r = \text{rank}(M)$, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$. For a fixed nonnegative ϵ , let $\sigma_{r(\epsilon)}$ be the smallest singular value σ_i which exceeds ϵ and write $\Sigma_\epsilon = \text{diag}(\sigma_1, \dots, \sigma_{r(\epsilon)}, 0, \dots, 0)$, $\Sigma_\epsilon^+ = \text{diag}(1/\sigma_1, \dots, 1/\sigma_{r(\epsilon)}, 0, \dots, 0)$, $\sigma^+ = \sigma_0^+$, $M_\epsilon^+ = V\Sigma_\epsilon^+U^*$, $M^+ = M_0^+ = V\Sigma^+U^*$. M^+ is the Moore–Penrose generalized inverse of M , and M_ϵ^+ is the ϵ -generalized inverse, useful, e.g., in the study of noisy perturbations of signals. The condition number of M_ϵ and M_ϵ^+ is smaller than that of M and M^+ for larger ϵ .

It is well known and easily verified that the RC processes (2.1) converge to the Moore–Penrose generalized inverse M^+ where the input matrix M is singular. In this case $r = \text{rank}(M) < n$, the smallest singular value σ_r of M plays the role of σ_n in the extension of the RC processes of Section 2, and σ_- denotes an approximation to σ_r from below, defining $\kappa_+ = \sigma_+/\sigma_-$. This extension is well known [B-I66], [SS74], [PS91]. Furthermore, the appropriately scaled RC processes [PS91, Section 7] converge to the ϵ -generalized inverse matrix M_ϵ^+ for a fixed positive ϵ .

The extension of these nontrivial RC algorithms to HRC processes is straightforward. We should just keep ϵ positive to avoid divergence. Apply any of the RC processes in [PS91, Section 7], for ϵ replaced by $\epsilon_h = t_h + \epsilon$ or by $\epsilon_h = (t_h^2 + \epsilon^2)^{1/2}$ as the basic RC algorithm at the h -th homotopic steps of our HRC processes in Sections 3 and 6, respectively. This yields the approximation of M_ϵ^+ for any Hermitian nonnegative definite or Hermitian indefinite matrix M , respectively.

The analysis in Sections 3–6 is immediately extended. By combining it with the results in [PS91, Section 7], we obtain that the RC process at the h -th homotopic step converges to $M_{\epsilon_h}^+$. Finally, as soon as t_h vanishes, we have $\epsilon_h = \epsilon > 0$, and the HRC process outputs M_ϵ .

Furthermore, for larger ϵ the RC processes are better conditioned. Indeed, the smallest unsuppressed singular value of M_h is now bounded from below by $t_h + \epsilon$ in Sections 3–5 and by $\epsilon_h = (t_h^2 + \epsilon^2)^{1/2}$ in Section 6. This improves the bounds $t_h + \lambda_n^-$ and $(t_h^2 + (\lambda^-)^2)^{1/2}$ for $\epsilon > \lambda_n^-$ and $\epsilon > \lambda^-$, respectively, and thus improves the condition of M_h .

If ϵ is positive but smaller than any singular value of M , then $M_\epsilon^+ = M^+$, and the RC processes in [PS91, Section 7] as well as our extended HRC processes in Sections 3–6 converge to the Moore–Penrose generalized inverse M^+ .

For any matrix M , we may reduce the computation of M^+ and M_ϵ^+ for any $\epsilon \geq 0$ to the Hermitian case by extending (6.1) and (6.2) as follows:

$$M_\epsilon^+ = M^*(MM^*)_{\epsilon^2}^+ = (M^*M)_{\epsilon^2}^+ M^*,$$

$$N_\epsilon^+ = \begin{pmatrix} 0 & (M_\epsilon^+)^* \\ M_\epsilon^+ & 0 \end{pmatrix}, \quad N = \begin{pmatrix} 0 & M \\ M^* & 0 \end{pmatrix}.$$

Note that ϵ is not squared in the latter (indefinite) case.

For the extension of the RC and HRC methods to the computation of the numerical generalized inverse M_ϵ^+ (and in particular $M^+ = M_0^+$) for a structured matrix M , see [P01a, Section 6.10]. Here an additional problem is the recovery of M_ϵ^+ from its compressed image (displacement) because the displacement does not completely define the matrix M_ϵ^+ , even for $\epsilon = 0$, unless M is nonsingular. For Toeplitz and Hankel matrices M of full rank and for $\epsilon = 0$, the problem can be avoided based on the results in [HH93], [HH94]. Theorem 6.10.1 and Corollary 6.10.2 in [P01a] cover the classes of structured matrices for which $\text{rank}(M_\epsilon^+M - I) = n - r_\epsilon$ is small, $r_\epsilon = \text{rank}(M_\epsilon^+)$.

11. CONCLUSION

We initialized Newton's iteration and some other residual correction (RC) processes for matrix inversion by means of homotopic (continuation) methods. According to our present estimates, this approach (proposed in [P92]) competes with the nonhomotopic RC processes for Hermitian positive definite matrices and accelerates these processes by twice in the complex indefinite Hermitian case. Unlike the nonhomotopic processes, the approach requires no pre-estimation of the smallest singular value of the input matrix. If the input matrices are well conditioned and structured (e.g., Toeplitz, Hankel, Vandermonde, Cauchy, Pick matrices, or matrices with structures of similar types), the homotopic RC approach supports the fastest known iterative solution algorithms running in nearly linear time and using linear memory space. Our numerical experiments with general and structured matrices have confirmed our theoretical analysis and furnished us with the appropriate values of parameters which define the homotopic RC processes.

An additional advantage is the convergence of the RC processes to the numerical generalized inverses of M ; it holds for general matrices M and can be extended to homotopic RC processes for general matrices as well as for a large class of structured matrices M .

Our theoretical estimates support the power of the homotopic processes only in the case of well-conditioned input matrices M . According to these estimates, the number of homotopic steps grows in proportion to $\log \kappa(M)$, $\kappa = \sigma_1/\sigma_r$, $r = \text{rank}(M)$, so that for larger condition numbers $\kappa(M)$ the algorithms are superseded by numerically stable direct algorithms running in $O(n^3)$ time for general matrices and in $O(n^2)$ time for structured matrices. The phenomenon of autocorrection in compression, observed experimentally, leaves some chances for practical extension of the power of the homotopic processes to structured matrices with larger condition numbers. This is a subject of our future study.

ADDED IN PROOF

In [PMRTY] the authors introduced the techniques of additive preconditioning, which improved the conditioning of general and structured matrices. This provides critical and far-reaching support for the RC and HRC processes.

REFERENCES

- [B85] J. R. Bunch, Stability of Methods for Solving Toeplitz Systems of Equations, *SIAM Journal on Scientific and Statistical Computing*, **6**, **2**, 349–364, 1985. MR0779410 (87a:65073)
- [B-I66] A. Ben-Israel, A Note on Iterative Method for Generalized Inversion of Matrices, *Math. Comp.*, **20**, 439–440, 1966.

- [BM01] D. A. Bini, B. Meini, Approximate displacement rank and applications, *Structured Matrices in Mathematics, Computer Science, and Engineering II* (V. Olshevsky Editor), *Contemporary Mathematics*, **281**, 215–232, American Mathematical Society, Rhode Island, 2001. MR1855993 (2002g:15030)
- [CKL-A87] J. Chun, T. Kailath, H. Lev-Ari, Fast Parallel Algorithm for QR-factorization of Structured Matrices, *SIAM Journal on Scientific and Statistical Computing*, **8**, **6**, 899–913, 1987. MR0911062 (89e:65035)
- [EPY98] I. A. Emiris, V. Y. Pan, Y. Yu, Modular Arithmetic for Linear Algebra Computations in the Real Field, *J. of Symbolic Computation*, **26**, 71–87, 1998. MR1633585 (2000f:15001)
- [FF63] D. K. Faddeev, V. N. Faddeeva, *Computational Methods of Linear Algebra*, W. H. Freeman, San Francisco, 1963. MR0158519 (28:1742)
- [GL96] G. H. Golub, C. F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1989 (2nd edition), 1996 (3rd edition). MR1002570 (90d:65055); MR1417720 (97g:65006)
- [HH93] G. Heinig, F. Hellinger, On the Bezoutian Structure of the Moore–Penrose Inverses of Hankel Matrices, *SIAM J. on Matrix Analysis and Applications*, **14**, **3**, 629–645, 1993. MR1227768 (94f:15006)
- [HH94] G. Heinig, F. Hellinger, Moore–Penrose Generalized Inverse of Square Toeplitz Matrices, *SIAM J. on Matrix Analysis and Applications*, **15**, **2**, 418–450, 1994. MR1266596 (95c:15008)
- [IK66] E. Issacson, H. B. Keller, *Analysis of Numerical Methods*, Wiley, New York, 1966. MR0201039 (34:924)
- [KKM79] T. Kailath, S. Y. Kung, M. Morf, Displacement Ranks of Matrices and Linear Equations, *J. Math. Anal. Appl.*, **68**, **2**, 395–407, 1979. MR0533501 (80k:65029)
- [KS99] T. Kailath, A. H. Sayed (Editors), *Fast Reliable Algorithms for Matrices with Structure*, SIAM Publications, Philadelphia, 1999. MR1715813 (2000h:65003)
- [P90] V. Y. Pan, On Computations with Dense Structured Matrices, *Math. Comp.*, **55**, **191**, 179–190, 1990. Proceeding version: *Proc. Intern. Symp. on Symbolic and Algebraic Comp. (ISSAC'89)*, 34–42, ACM Press, New York, 1989. MR1023051 (90m:65085)
- [P92] V. Y. Pan, Parallel Solution of Toeplitz-like Linear Systems, *J. of Complexity*, **8**, 1–21, 1992. MR1153611 (92k:65202)
- [P92b] V. Y. Pan, *Can We Utilize the Cancellation of the Most Significant Digits?* *Tech. Report TR-92-061*, The International Computer Science Institute, Berkeley, California, 1992.
- [P93] V. Y. Pan, Decreasing the Displacement Rank of a Matrix, *SIAM Journal on Matrix Analysis and Applications*, **14**, **1**, 118–121, 1993. MR1199549 (93k:15039)
- [P01a] V. Y. Pan, *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser/Springer, Boston/New York, 2001. MR1843842 (2002i:65001)
- [P01b] 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.
- [Pa] V. Y. Pan, A Homotopic/Factorization Process for Toeplitz-like Matrices with Newton's/Conjugate Gradient Stages, Technical Report TR 2004014, *Ph.D. Program in Computer Science, Graduate Center, City University of New York*, New York, 2004.
- [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, A.H. Sayed, Editors), 189–210, SIAM Publications, Philadelphia, 1999. MR1715820
- [PKRC02] V. Y. Pan, M. Kunin, R. Rosholt, H. Cebecioglu, Residual Correction Algorithms for General and Structured Matrices, Technical Report TR 2002020, *Ph.D. Program in Computer Science, Graduate Center, City University of New York*, New York, 2002.
- [PMRTY] V. Y. Pan, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan, Additive Preconditioning in Matrix Computations, Technical Report TR 2005009, *Ph.D. Program in Computer Science, Graduate Center, City University of New York*, New York, 2005.

- [PR01] V. Y. Pan, Y. Rami, Newton's Iteration for the Inversion of Structured Matrices, *Advances in the Theory of Computational Mathematics, Vol. 4: Structured Matrices: Recent Developments in Theory and Computation*, (edited by D. A. Bini, E. Tyrtyshnikov and P. Yalamov), 79–90, Nova Science Publishers, Huntington, New York, 2001.
- [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. MR1878944 (2002m:65036)
- [PS91] V. Y. Pan, R. Schreiber, An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications, *SIAM J. on Scientific and Statistical Computing*, **12, 5**, 1109–1131, 1991. MR1114976 (92d:65056)
- [PVBWC04] V. Y. Pan, M. Van Barel, X. Wang, G. Codevico, Iterative Inversion of Structured Matrices, *Theoretical Computer Science*, **315, 2-3**, 581-592 (Special Issue on Algebraic and Numerical Computing), 2004. MR2073066
- [PW03] V. Y. Pan and X. Wang, Inversion of Displacement Operators. *SIAM Journal on Matrix Analysis and Applications*, **24, 3**, 660–677, 2003. MR1972673 (2004a:47012)
- [PZHD97] V. Y. Pan, A. Zheng, X. Huang, O. Dias, Newton's Iteration for Inversion of Cauchy-like and Other Structured Matrices, *J. of Complexity*, **13**, 108–124, 1997. MR1449764 (98h:65014)
- [Par80] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980. MR(81j:65063)
- [S33] G. Schultz, Iterative Berechnung der Reciproken Matrix, *Z. Angew. Meth. Mech.*, **13**, 57–59, 1933.
- [SS74] T. Söderström, W. Stewart, On the Numerical Properties of an Iterative Method for Computing the Moore–Penrose Generalized Inverse, *SIAM J. Numer. Anal.*, **11**, 61–74, 1974. MR0341843 (49:6589)

MATHEMATICS AND COMPUTER SCIENCE DEPARTMENT, LEHMAN COLLEGE, CUNY, BRONX, NEW YORK 10468; PH. D. PROGRAM IN MATHEMATICS, GRADUATE CENTER, CUNY, NEW YORK, NEW YORK 10016

E-mail address: `victor.pan@lehman.cuny.edu`

PH.D. PROGRAM IN COMPUTER SCIENCE, GRADUATE CENTER, CUNY, NEW YORK, NEW YORK 10016

MATHEMATICS AND COMPUTER SCIENCE DEPARTMENT, LEHMAN COLLEGE, CUNY, BRONX, NEW YORK 10468

UNIVERSITY OF KOCAELI, DEPARTMENT OF MATHEMATICS, 41300 IZMIT, KOCAELI, TURKEY