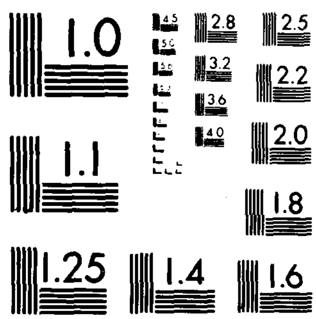
AD-A086 349 CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF COMPUTER --ETC F/G 12/1
ON THE OPTIMAL SOLUTION OF LARGE LINEAR SYSTEMS,(U)
JAN 80 J F TRAUB, H WOZNIAKOWSKI N00014-76-C-0370

UNCLASSIFIED NL

1 OF 1
AD
A086349

END
DATE
FILMED
8-80
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ON THE OPTIMAL SOLUTION OF LARGE LINEAR SYSTEMS

J. F. Traub
Department of Computer Science
Columbia University
New York, N.Y. 10027

H. Wozniakowski
Department of Computer Science
Columbia University
New York, N.Y. 10027
(Visiting from the University of Warsaw)

January 1980

DEPARTMENT of COMPUTER SCIENCE

Columbia University

12

ON THE OPTIMAL SOLUTION OF LARGE LINEAR SYSTEMS

J. F. Traub
Department of Computer Science
Columbia University
New York, N.Y. 10027

H. Wozniakowski
Department of Computer Science
Columbia University
New York, N.Y. 10027
(Visiting from the University of Warsaw)

Jan[redacted] [redacted]80

N00014-76-C-0370
NSF-MCS78-23676

This research was supported in part by the National Science Foundation under Grant MCS-7823676 and the Office of Naval Research under Contract N00014-76-C-0370, NR 044-422.

403081

ABSTRACT

We investigate the minimal number of matrix-vector multiplications to approximately solve a linear system. The minimal number of multiplications depends on the properties of a class of problems such as symmetry, positive definiteness, and bound on condition number. For different classes of problems we obtain the minimum exactly or almost exactly and establish which algorithms are optimal, that is, attain the minimum. Furthermore, we obtain quantitative results on how the lack of certain properties increases the minimum.

TABLE OF CONTENTS

1. INTRODUCTION

Many papers deal with the iterative solution of a large linear system $Ax = b$. Typically one constructs an algorithm ϕ which generates a sequence $\{x_k\}$ converging to the solution $\alpha = A^{-1}b$; the calculation of x_k requires k matrix-vector multiplications and x_k lies in a subspace spanned by b, Ab, ..., A^kb. The algorithm ϕ is often chosen to guarantee good convergence properties of the sequence $\{x_k\}$. In some cases, ϕ is defined to minimize some measure of the error in a <u>restrictive</u> class of algorithms. For instance, let this class be defined as the class of "polynomial" algorithms, i.e., $\alpha - x_k = W_k(A)\alpha$, where W_k is a polynomial of degree at most k and $W_k(0) = 1$. Then choosing W_k as the polynomial minimizing the k-th residual $\|Ax_k - b\| = \|W_k(A)\alpha\|$, we obtain the <u>minimal residual</u> algorithm, ϕ^{mr}. If A is symmetric, positive definite and $a = 1/\|A^{-1}\|$, $b = \|A\|$ are known, then choosing W_k as the polynomial minimizing $\max\{|W_k(t)| : t \in [a,b]\}$, we obtain the <u>Chebyshev</u> algorithm, ϕ^{ch}.

It seems to us that this procedure is unnecessarily restrictive. It is not clear, a priori, why an algorithm has to construct x_k of the form $\alpha - x_k = W_k(A)\alpha$. One might hope that by not restricitng the class of algorithms it is possible to obtain better algorithms.

In this paper we do <u>not</u> impose any restriction on the class of algorithms ϕ which construct x_k using the information $N_k(A,b) = [b, Ab, \ldots, A^kb]$. Assuming that the matrix A belongs to a given class of nxn nonsingular matrices F, we measure the goodness of an algorithm ϕ by the minimal number of steps k which are necessary to find x_k such that $\|Ax_k-b\| < \varepsilon$ for a given positive $\varepsilon \in (0,1]$. (We assume that $\|b\| = 1$.) We define two types of optimality. An algorithm ϕ is said to be <u>strongly optimal</u> if it requires the minimal number of steps for <u>every</u> A from the class F. An algorithm ϕ is said to be <u>optimal</u> if it requires the minimal number of steps for a worst case A from F. (For the precise definition see Section 2.)

The main result of this paper is that the minimal residual algorithm is almost strongly optimal provided that the class F is orthogonally invariant, i.e., $A \in F$ implies $QAQ^T \in F$ for any orthogonal Q. We show that the assumption of orthogonal invariance is essential. That is, if F is not orthogonally invariant, then the optimality properties of the ϕ^{mr} algorithm disappear.

Usually the class F depends on a parameter. For instance, we consider the classes F_1, F_2 and F_3 of nxn matrices with condition number bounded by a given M, $M \geq 1$. The class F_1 consists of symmetric and positive definite matrices, the class F_2 differs from F_1 by the lack of positive definitess, and the class F_3 differs from F_2 by the lack of symmetry. Note that the minimal residual algorithm, even though it is almost strongly optimal for any value of M, does not use M for the construction of the sequence $\{x_k\}$.

We also prove that if ε is not too small, the Chebyshev algorithm is optimal but <u>not</u> strongly optimal for the class F_4

of nxn matrices of the form $A = I-B$ where B is symmetric and $\|B\| \le \rho$ with $\rho < 1$. In contrast to the previous case, the ϕ^{ch} algorithm depends essentially on ρ. We also consider the class F_5 which differs from F_4 by the lack of symmetry of matrices B. We establish the asymptotic optimality of the successive approximation algorithm for this class.

For all these five classes we find the optimal class index which is defined as the number of steps required by an optimal algorithm to find x_k with $\|Ax_k-b\| < \varepsilon$. We are able to conclude precisely how the lack of positive definiteness and/or symmetry increases the optimal class index.

For the optimal algorithms considered in this paper we can also guarantee that the construction of x_k requires a close to minimal number of arithmetic operations and storage. From these properties it follows that they are almost optimal complexity algorithms, i.e. algorithms which minimize the total cost (measured by the number of arithmetic operations) of finding a vector x such that $\|Ax-b\| < \varepsilon$.

In the first six sections of this paper we consider optimal algorithms for finding a vector x such that the residual vector $Ax-b$ has norm less than ε. In Section 7 we introduce a family of approximation criteria for choosing a vector x. We generalize the previous optimality results. Among our results we show that the conjugate gradient algorithm is almost strongly optimal, that if ε is not too small then the Chebyshev algorithm is optimal (but <u>not</u> strongly optimal) for the class F_4 with an arbitrary choice of the approximation criterion, and the successive approximation algorithm is optimal (but <u>not</u> strongly optimal) for the class F_5 for a certain choice of the approximation criterion.

We stress that with a few exceptions the results of this paper are <u>not</u> asymptotic. That is, we know the exact values of the optimal class indices to within at most unity for <u>every</u> ε from the interval $(0,1]$. This is in a sharp contrast to many results in algebraic complexity where only small ε results can be established.

The problems and proof techniques of this paper follow the information approach of the monograph by Traub and Woźniakowski [80]. There are many interesting relations between the optimality results of this paper and the general results of the monograph. For the reader's convenience we do not use the general terminology and results of Traub and Woźniakowski [80].

For simplicity we consider only the real case, although the generalization to the complex case is straightforward.

We summarize the contents of the paper. Section 2 presents the basic concepts of strongly optimal, optimal, and almost strongly optimal algorithms. The minimal residual algorithm is defined in Section 3.

In Section 4 we establish the main result that the minimal residual algorithm is almost strongly optimal provided the class F is orthogonally invariant. In Sections 5 and 6 we find the optimal class index for five orthogonally invariant classes.

Section 7 deals with generalized criteria. The generalized minimal algorithm is defined and proven to be almost strongly optimal. Section 8 deals with the complexity of finding an ε-approximation. In Section 9 we briefly compare the Gauss elimination algorithm with the minimal residual algorithm. In the final section we pose some open problems concerning the optimality properties of the information studied in this paper.

2. BASIC CONCEPTS

Let F be a subclass of the class of $n \times n$ nonsingular real matrices. Let b be a given $n \times 1$ real vector such that $\|b\| = \sqrt{(b,b)} = 1$. For a given positive ε, $\varepsilon > 1$, we seek a real vector x whose residual has norm less than ε, i.e.,

$$(2.1) \qquad \|Ax - b\| < \varepsilon, \qquad A \in F.$$

We call x an ε-approximation. Since b is normalized to unity, (2.1) measures the relative error of the residual vector. In Section 7 we discuss the problem of finding x with relative error less than ε in a variety of norms.

To find an ε-approximation we need some information about the matrix A. We define an information operator N_k as

$$(2.2) \qquad N_k(A,b) = [b, Ab, A^2b, \ldots, A^kb]$$

for $k = 1, 2, \ldots$.

<u>Remark 2.1</u>

Let $z_0 = b$, $z_i = Az_{i-1}$, for $i = 1, 2, \ldots, k-1$. Then (2.2) can be rewritten as

$$(2.3) \qquad N_k(A,b) = [z_0, Az_0, Az_1, \ldots, Az_{k-1}].$$

Thus the computation of $N_k(A,b)$ requires k matrix-vector multiplications. If A is sparse $N_k(A,b)$ can be computed in time proportional to kn rather than kn^2. Usually instead of computing $N_k(A,b)$ we compute

$$N_k'(A,b) = [\, b,\ Aw_1,\ Aw_2, \ldots,\ Aw_k \,]$$

where w_i is a linear combination of $b, Ab, \ldots, A^{i-1}b$ for $i = 1, 2, \ldots, k$. It is easy to show that all the results of this paper also hold for the information operator N_k'. ∎

Remark 2.2

Note that z_i in (2.3) is a function of previously computed vectors. Thus, N_k is an example of an adaptive information operator. See Section 9 where we discuss adaptive information operators in general. ∎

We define an algorithm $\phi = \{\phi_k\}$ as a sequence of mappings $\phi_k : N_k(F, \mathbb{R}^n) \to \mathbb{R}^n$. The algorithm ϕ generates the sequence $x_k = \phi_k(N_k(A,b))$ based on the information $N_k(A,b)$. We are interested in the smallest value of k for which x_k satisfies (2.1), i.e., $\| Ax_k - b \| < \varepsilon$. In general, there exist many different matrices $\tilde{A}$ from F which share the same information as A, i.e., $N_k(\tilde{A},b) = N_k(A,b)$. Thus $x_k = \phi_k(N_k(A,b)) = \phi_k(N_k(\tilde{A},b))$ must satisfy (2.1) for A and $\tilde{A}$. Define

(2.4) $$V(y_k) = \{\tilde{A} : \tilde{A} \in F,\ N_k(\tilde{A},b) = y_k\},\quad y_k = N_k(A,b).$$

Let

(2.5) $$k(\phi, A) = \min\{\, k : \| \tilde{A}x_k - b \| < \varepsilon,\ \forall\, \tilde{A} \in V(y_k)\}$$

be the matrix index of ϕ. (If the set of k in (2.5) is empty, we set $k(\phi,A) = +\infty$.) Let

(2.6) $$k(\phi, F) = \sup_{A \in F} k(\phi, A)$$

<u>be the class index of ϕ</u>.

Thus, the matrix index of ϕ denotes the minimal number of steps required to find an ε-approximation using the algorithm ϕ for all matrices $\tilde{A}$ from F which share the same information as A. The class index of ϕ denotes the same concept for the hardest problem.

We seek algorithms with minimal indices. Let

$$(2.7) \qquad k(A) = \min_{\phi} k(\phi, A)$$

be the <u>optimal matrix index</u> and let

$$(2.8) \qquad k(F) = \max_{A \in F} k(A) \; (= \min_{\phi} k(\phi, F))$$

be the <u>optimal class index</u>.

<u>Remark 2.3</u>

It is possible that $k(A) << k(F)$. For instance, assume that $Ab = b$. Then, of course, setting $x_1 = \phi_1(y_1) = b$ we have $\tilde{A}x_1 = b$ for $\tilde{A} \in V(y_1)$. Thus $k(A) = 1$ for every ε. As we shall see later $k(F)$ can be equal to n.

Thus even if the optimal class index is large it can happen, due to favorable properties of A and b, that the optimal matrix index is small. The algorithms with small matrix index are therefore very useful for applications. This motivates our interest in algorithms with small matrix index. ∎

We are ready to introduce two concepts of optimal algorithms.

An algorithm ϕ is called <u>strongly optimal</u> iff

(2.9) $$k(\phi, A) = k(A), \quad \forall A \in F$$

and is called <u>optimal</u> iff

(2.10) $$k(\phi, F) = k(F).$$

We can sometimes establish that the matrix or class index of an algorithm is slightly larger than the optimal index. It is convenient to introduce the concepts of <u>almost</u> strongly optimal algorithm and <u>almost</u> optimal algorithm as follows. An algorithm ϕ is <u>almost strongly optimal</u> iff

(2.11) $$k(\phi, A) \le k(A) + c, \quad \forall A \in F,$$

and is <u>almost optimal</u> iff

(2.12) $$k(\phi, F) \le k(F) + c$$

for some small integer c.

Thus an almost strongly optimal algorithm requires at most c more steps than a strongly optimal one. Usually $k(A) >> c$ and therefore an almost strongly optimal algorithm is as useful in practice as a strongly optimal one.

<u>Remark 2.4</u>

All concepts introduced in this section also depend on the size n, the information N_k, the vector b and ε. To simplify notation and terminology we do not make this explicit but the reader should keep in mind that all the results are relative to n, N_k, b and ε.

Sometimes we shall need to show the dependence on b. Then we shall write $k(\phi, A, b)$ and $k(\phi, F, b)$ instead of $k(\phi, A)$ and $k(\phi, F)$, respectively. ∎

Remark 2.5

In most of this paper we focus on the minimal number of steps $k(F)$ required to find an ε-approximation. Of course, we also want to minimize the complexity (the cost) of finding an ε-approximation. In Section 8 we derive very tight bounds on the complexity of this problem and we show that the complexity depends primarily on $k(F)$. ∎

We conclude this section by showing that

$$(2.13) \qquad k(F) \le n.$$

Indeed, assume $k = n$. Since $b, Ab, \ldots, A^n b$ are linearly dependent and A is nonsingular, there exist numbers $c_1, c_2, \ldots, c_n$ such that

$$b = c_1 Ab + \ldots + c_n A^n b = A(c_1 b + \ldots + c_n A^{n-1} b).$$

Setting $x_n = \phi_n(N_n(A,b)) = c_1 b + \ldots + c_n A^{n-1} b$ we find that $\| A x_n - b \| = 0$. This implies (2.13). As we shall see later there exist many interesting classes F for which $k(F)$ is much less than n for reasonable values of ε.

Remark 2.6

We defined $k(A)$ and $k(F)$ as the mimina of $k(\phi, A)$ and $k(\phi, F)$ respectively. From (2.13) we conclude that these mimina exist. Thus, $k(A)$ and $k(F)$ are well defined. ∎

3. MINIMAL RESIDUAL ALGORITHM

In this section we derive the minimal residual algorithm. Let

$$N_k(A,b) = [z_0, z_1, \ldots, z_k] \qquad \text{with} \quad z_i = A^i b.$$

Knowing the vectors z_i we define $c_1^*, \ldots, c_k^*$ as the coefficients which minimize the norm of the residual in the space spanned by $z_1, z_2, \ldots, z_k$. Thus

$$(3.1) \qquad \| b - c_1^* z_1 - \ldots - c_k^* z_k \| = \min_{c_i} \| b - c_1 z_1 - \ldots - c_k z_k \| .$$

The minimal residual algorithm ϕ^{mr}, briefly the mr algorithm, is defined as

$$(3.2) \qquad x_k = \phi_k^{mr}(N_k(A,b)) = c_1^* b + \ldots + c_k^* A^{k-1} b .$$

Note that $x_k = A^{-1}(c_1^* z_1 + \ldots + c_k^* z_k)$. The vector $c^* = [c_1^*, \ldots, c_k^*]^T$ satisfies the linear equations

$$(3.3) \qquad M c^* = g$$

where $M = ((z_i, z_j))$ and $g = [(z_1,b), \ldots, (z_k,b)]^T$. The matrix M is nonsingular iff $z_1, z_2, \ldots, z_k$ are linearly independent. If $z_1, z_2, \ldots, z_k$ are linearly dependent then b belongs to the space $\{z_1, \ldots, z_{k-1}\}$ and $Ax_k = b$.

Let P be a polynomial of degree at most k such that $P(0) = 0$. Let Π_k be the class of such polynomials. Then (3.1) can be rewritten as

$$(3.4) \qquad \| (I - P_k^*(A)) b \| = \inf_{P \in \Pi_k} \| (I - P(A)) b \|$$

where $P_k^*(t) = c_1^* t + \ldots + c_k^* t^k \in \Pi_k$.

If A is symmetric and postiive definite then the mr algorithm is a variant of the conjugate gradient iteration. (See for instance Stiefel [58].) In this case it is known that the polynomials

$$W_k^*(t) = 1 - P_k^*(t), \quad W_k^*(0) = 1,$$

are orthogonal,

$$(3.5) \qquad (W_k^*, W_i^*) = \sum_{j=1}^{m} |c_j|^2 \lambda_j W_k^*(\lambda_j) W_i^*(\lambda_j) = 0$$

for $k \neq i$ where

$$b = \sum_{j=1}^{m} c_j \xi_j$$

with ξ_j being an eigenvector of A associated with the eigenvalue λ_j, $A\xi_j = \lambda_j \xi_j$, $\|\xi_j\| = 1$, $0 < \lambda_1 < \lambda_2 < \ldots \lambda_m$ where $m \leq n$ and $c_j \neq 0$ for $j = 1, 2, \ldots, m$.

Equation (3.5) implies that

$$(3.6) \qquad (Ar_k, r_i) = 0$$

where $r_j = Ax_j - b$ is the residual vector.

There are many efficient ways to compute x_k for symmetric positive definite matrices. For instance x_k can be found as follows. Let $x_0 = 0$. For $i = 0, 1, \ldots, k-1$ define

(3.7) $$x_{i+1} = x_i + \frac{1}{q_i}\{f_{i-1}(x_i - x_{i-1}) - r_i\}$$

where

$$q_i = \frac{(Ar_i, Ar_i)}{(r_i, Ar_i)} - f_{i-1}$$

(3.8)

$$f_{-1} = 0, \quad f_{i-1} = \frac{(r_i, Ar_i)}{(r_{i-1}, Ar_{i-1})} q_{i-1} .$$

The residual vectors r_i satisfies a similar recurrence relation as x_i, i.e., $r_{i+1} = r_i + \frac{1}{q_i}\{f_{i-1}(r_i - r_{i-1}) - Ar_i\}$. Note that x_{i+1} defined by (3.7) is a linear combination of $b, Ab, \ldots, A^i b$ and its computation requires only the knowledge of the vectors $b, Ab, \ldots, A^{i+1} b$. See Remark 2.1. Roundoff-error analysis of a class of conjugate gradient algorithms including some information on the mr algorithm can be found in Woźniakowski [80].

Remark 3.1

We assume that the initial approximation $x_0 = 0$. This assumption is not restrictive. Indeed, let x_0 be a nonzero approximation and let $c = \|b - Ax_0\| \neq 0$. Then we apply the mr algorithm (3.7) and (3.8) to the system $Ax = b'$ with $b' = (b - Ax_0)/c$. If we find x' such that $\|Ax' - b'\| < \varepsilon/c$ then $x = cx' + x_0$ is an ε-approximation to the original system since $\|Ax - b\| < \varepsilon$. ∎

If A is symmetric then the matrix M in (3.3) is Toeplitz and a recent algorithm due to Brent [78], and Yun and Gustavson [79] can be employed to find the vector c^* in time proportional to $k \log^2 k$.

We end this section by a remark on the matrix index of the mr algorithm. Let $\tilde{A} \in V(y_k)$. (See (2.4)). Then $\tilde{A}^i b = A^i b$ for $i = 1, 2, \ldots, k$. Due to (3.2) we have

$$\tilde{A} x_k - b = A x_k - b, \qquad \forall \tilde{A} \in V(y_k) \tag{3.9}$$

and (2.5) can now be simplified to

$$k(\phi^{mr}, A) = \min\{k : \| A x_k - b \| < \varepsilon\}. \tag{3.10}$$

It is obvious that $x_n = A^{-1} b$ which implies that

$$k(\phi^{mr}, A) \le n. \tag{3.11}$$

If A is symmetric and positive definite then (3.5) implies $x_m = A^{-1}b$ and $x_i \neq A^{-1}b$ for $i < m$. Thus $k(\phi^{mr}, A) \le m$ and for sufficiently small ε, $k(\phi^{mr}, A) = m$.

4. OPTIMALITY OF THE MR ALGORITHM

In this section we study optimality properties of the mr algorithm defined by (3.2). As we shall see the mr algorithm is an almost strongly optimal algorithm provided the class F is "orthogonally invariant". This concept is defined as follows. Let ω be a real $n \times 1$ vector such that $\|\omega\| = 1$. Then

$$(4.1) \qquad Q = Q(\omega) = I - 2\,\omega\,\omega^T$$

is symmetric and orthogonal, $Q^2 = I$. We say F is <u>orthogonally invariant</u> iff

$$(4.2) \qquad A \in F \quad \Rightarrow \quad QAQ \in F$$

for every Q of the form (4.1).

<u>Remark 4.1</u>

Let us recall that every orthogonal matrix Q can be decomposed into a product $Q = Q_1 Q_2 \ldots Q_n$ where Q_i is of the form (4.1). Thus, F is orthogonally invariant iff

$$A \in F \quad \Rightarrow \quad QAQ^T \in F$$

for every orthogonal Q. ∎

For example, the class of symmetric matrices, the class of symmetric positive definite matrices, and the class of matrices with condition number bounded by a given constant are orthogaonally invariant.

We first investigate how the optimal matrix index depends of b. (So we use the notation $k(A,b)$ instead of $k(A)$.)

Lemma 4.1

If F is orthogonally invariant then

(4.3) $$k(A,b) = k(QAQ\,,\,Qb)\,.$$

for every Q of the form (4.1). ∎

Proof

Let $y_k = N_k(A,b)$ and $y_k' = N_k(QAQ\,,\,Qb)$. Then $y_k' = Qy_k$. Let $\phi = \{\phi_k\}$ be an algorithm. Define the algorithm $\phi' = \{\phi_k'\}$ as

$$\phi_k'(y_k') = Q\,\phi_k(Qy_k').$$

Let $\tilde{A}' \in V(y_k')$. Then $\tilde{A} = Q\tilde{A}'Q \in V(y_k)$

and

$$\|\tilde{A}'\,\phi_k'(y_k') - Qb\| = \|Q\tilde{A}'Q\,\phi_k(y_k) - Q^2 b\| =$$
$$= \|\tilde{A}\,\phi_k(y_k) - b\|$$

This implies that $k(\phi', QAQ, Qb) \le k(\phi, A, b)$. Since ϕ is an arbitrary algorithm this yields $k(QAQ, Qb) \le k(A,b)$. To prove the reverse inequality it is enough to interchange the role of ϕ and ϕ'. ∎

From Lemma 4.1 easily follows

Lemma 4.2

If F is orthogonally invariant then the optimal class index is independent of b, i.e.

(4.4) $$k(F, b) = k(F), \quad \forall \|b\| = 1 .$$ ∎

Proof

Let b_1 and b_2 be two vectors, $\|b_1\| = \|b_2\| = 1$. Then there exists a matrix Q of the form (4.1) such that

$$Q b_1 = b_2 .$$

(The existence of such a matrix follows from the Householder transformation). Then Lemma 4.1 guarantees that

$$k(A, b_1) = k(QAQ, Qb_1) = k(QAQ, b_2)$$

which easily yields that

$$k(F, b_1) = k(F, b_2) .$$ ∎

We now prove that the algorithm ϕ^{mr} has properties analogous to those described in Lemmas 4.1 and 4.2.

Lemma 4.3

If F is orthogonally invariant then

(i) $k(\phi^{mr}, QAQ, Qb) = k(\phi^{mr}, A, b)$

for every Q of the form (4.1),

(ii) $k(\phi^{mr}, F, b) = k(\phi^{mr}, F), \quad \forall \|b\| = 1,$

i.e., the class index of ϕ^{mr} is independent of b. ∎

Proof

(i) Observe that the coefficients c_i^* of (3.1) are independent of Q. From (3.2) we get

$$x_k' = \phi_k^{mr}(N_k(QAQ, Qb)) = Q\,\phi_k^{mr}(N_k(A,b)) = Q\,x_k.$$

Since $\| b - A x_k \| = \| Qb - QAQ x_k' \|$, we have to perform exactly the same number of steps for the problem (A,b) and the problem (QAQ, Qb) to find an ε-approximation. This proves (i).

(ii) Observe, as in Lemma 4.2, that

$$k(\phi^{mr}, A, b_1) = k(\phi^{mr}, QAQ, b_2)$$

where $b_2 = Qb_1$, $\| b_1 \| = \| b_2 \| = 1$. This yields

$$k(\phi^{mr}, F, b_1) = k(\phi^{mr}, F, b_2)$$

and proves (ii). ∎

We are ready to prove the main result of this section which exhibits a close relation between the matrix index of ϕ^{mr} and the optimal matrix index.

Theorem 4.1

If F is orthogonally invariant then the matrix index of the mr algorithm differs by at most unity from the optimal matrix index, i.e.,

$$(4.5) \qquad k(A) = k(\phi^{mr}, A) + a, \ \forall A \in F,$$

where $a = 0$ or $a = -1$.

Furthermore either $a = 0$ or $a = -1$ can hold. ∎

Proof

Let $\phi = \{\phi_k\}$ be any algorithm. Let $k = k(\phi, A) < +\infty$. This means that

(4.6) $$\|\tilde{A}\, x_k' - b\| < \varepsilon\,, \qquad \tilde{A} \in V(y_k)$$

where $x_k' = \phi(N_k(A,b))$. Decompose

$$x_k' = z_1 + z_2$$

where z_1 is a linear combination of $b, Ab, \ldots, A^k b$ and z_2 is orthogonal to $b, Ab, \ldots, A^k b$. Define

$$\omega = \begin{cases} \dfrac{z_2}{\|z_2\|} & \text{if } z_2 \neq 0\,, \\ 0 & \text{otherwise.} \end{cases}$$

Clearly $(\omega, A^i b) = 0$ for $i = 0, 1, \ldots, k$. Let

$$\tilde{A} = Q A Q \quad \text{with} \quad Q = I - 2\omega\, \omega^T\,.$$

Then $\tilde{A} \in F$ and

$$\tilde{A}^i b = Q A^i Q b = Q A^i b = A^i b\,, \quad i = 1, 2, \ldots, k\,.$$

Thus, $\tilde{A} \in V(y_k)$ and

$$\|\tilde{A} x_k' - b\| = \|A x_k' - 2(\omega, x_k')\, A\omega - b\|$$

Note that $(\omega, x_k')\, A\omega = Az_2$ which yields

$$\|\tilde{A} x_k' - b\| = \|Az_1 - b - Az_2\|.$$

Observe that

$$\| Az_1 - b \| \le \frac{1}{2} (\|Az_1 - b - Az_2\| + \| Az_1 - b + Az_2 \|) =$$

$$= \frac{1}{2} (\| \tilde{A} x'_k - b \| + \| A x'_k - b \|) < \varepsilon$$

due to (4.6).

Recall that $x_{k+1} = \phi^{mr}_{k+1} (N_{k+1}(A,b))$ lies in the same subspace as z_1 and

$$\| Ax_{k+1} - b \| \le \| Az_1 - b \| < \varepsilon .$$

From (3.10) we conclude that

$$k(\phi^{mr}, A) \le k+1 = k(\phi , A) + 1.$$

Since ϕ is an arbitrary algorithm we have

$$k(\phi^{mr},A) \le k(A) + 1 .$$

On the other hand it is obvious that $k(A) \le k(\phi^{mr} , A)$. This proves (4.5).

We now show that either value of a can occur. Let $n = 2$ and

$$F = \{A : A = A^* > 0 \text{ and } \operatorname{cond}(A) \le M\}$$

be the class of 2×2 symmetric positive definite matrices with condition number $\operatorname{cond}(A) = \|A\| \|A^{-1}\|$ bounded by a given number M, $M > 1$. Note that F is orthogonally invariant. Let $b = [1,0]^T$ and $\varepsilon \le \sqrt{1 / (1 + c^2)}$ where

$$c = 2 \sqrt{M} / (M - 1) .$$

(i) Assume first that $Ab = [c , 1]^T$. Then

$$x_1 = \phi_1^{mr}(b , Ab) = \frac{c}{1 + c^2} b ,$$

and

$$\| A x_1 - b \| = 1/\sqrt{1+c^2} \ge \varepsilon$$

Thus, $k(\phi^{mr}, A) > 1$. From (2.13) we conclude

$$k(\phi^{mr}, A) = 2.$$

We show that $k(A) = 1$. Indeed, knowing b and Ab we conclude that A is of the form

$$A = \begin{bmatrix} c & 1 \\ 1 & x \end{bmatrix}$$

where x is a real number choosen in such a way that A is positive definite and $\operatorname{cond}(A) \le M$. The eigenvalues of A are

$$\lambda_{1,2} = (c + x \pm \sqrt{(c-x)^2 + 4})/2.$$

Thus $x > 1/c$ guarantees positive definiteness of A. The condition number of A is

$$\operatorname{cond}(A) = f(x) = (c + x + \sqrt{(c-x)^2 + 4})/(c + x - \sqrt{(c-x)^2 + 4})$$

Note that

$$f(x) \ge \min_x f(x) = f(c + 2/c) = M .$$

Since $\operatorname{cond}(A)$ is at most M we conclude that

$$x = c + 2/c .$$

This means that $V(y_k)$ consists of one element and the algorithm

$$\phi_1(b, Ab) = A^{-1} b$$

is well defined and has zero error. This proves that

$$k(A) = 1 = k(\phi^{mr}, A) - 1 .$$

Hence, (4.5) holds with $a = -1$.

(ii) Assume now that $Ab = b$. Then, of course,

$$x_1 = \phi_1^{mr}(b, Ab) = b \quad \text{and} \quad k(A) = k(\phi^{mr}, A) = 1$$

Hence (4.5) holds with $a = 0$. This completes the proof of the theorem. ∎

From Theorem 4.1 it easily follows that ϕ^{mr} is almost strongly optimal and $k(\phi^{mr}, F)$ differs at most by unity from the optimal class index $k(F)$.

<u>Corollary 4.1</u>

If F is orthogonally invariant then

(i) the minimal residual algorithm is almost strongly optimal (with $c = 1$ in (2.11)).

(ii) $k(F) = k(\phi^{mr}, F) + a$ where $a = 0$ or $a = -1$.

In Section 6 we show that either value of a in (ii) of Corollary 4.1 is possible. Since $k(F)$ is usually large, Corollary 4.1 states that $k(F)$ is essentially equal to $k(\phi^{mr}, F)$. Thus, it is enough to know $k(\phi^{mr}, F)$ to conclude the value of $k(F)$. In Sections 5 and 6 we find $k(\phi^{mr}, F)$ for different orthogonally invariant classes F.

We end this section by a remark that if F is <u>not</u> orthogonally invariant then none of the optimality properties of the algorithm ϕ^{mr} hold. More precisely we present an example of F for which the mr algorithm can be arbitrarily far from optimal. We also show that $k(F)$ depends on b.

Example 4.2

Let Φ be the class of $n \times n$ symmetric tridiagonal matrices whose diagonal elements are equal to unity. Thus $A \in \Phi$ implies

$$A = \begin{pmatrix} 1 & a_1 & & & \\ a_1 & 1 & \ddots & & \\ & \ddots & \ddots & & \\ & & & 1 & a_{n-1} \\ & & & a_{n-1} & 1 \end{pmatrix} , \quad a_i \in \mathbb{R} .$$

Let

$$F = \{A : A \in \Phi; \ \ \mathrm{cond}(A) \le M\}$$

for a given M, $M > 1$. The class F is not orthogonally invariant since the matrix QAQ with Q of the form (4.1) is not necessarily tridiagonal. We consider two cases for two different vectors b.

(i) Assume first that

$$b_1 = [1, 1, \dots, 1]^T$$

Then knowing $z = Ab_1 = [z_1, \dots, z_n]^T$ we get

$$1 + a_1 = z_1$$

$$a_{i-1} + 1 + a_i = z_i , \quad i = 2, \dots, n-1.$$

From this we find the coefficients a_i,

$$a_1 = z_1 - 1$$

$$a_i = z_i - 1 - a_{i-1} , \quad i = 2, \dots, n-1.$$

Since we know the matrix A, the algorithm

$$x_1' = \phi_1(b_1 , Ab_1) = A^{-1} b_1$$

is well defined and $\| A x_1' - b_1 \| = 0$. Thus

(4.7) $k(F, b_1) = 1, \forall \varepsilon \in (0, 1]$.

It can be verified that for sufficiently small ε, the algorithm ϕ^{mr} has to use the information $N_n(A, b)$ which means that

$$k(\phi^{mr}, F, b_1) = n.$$

Hence we get the smallest possible value of $k(F, b_1)$ and the largest possible value of $k(\phi^{mr}, F, b_1)$.

(ii) Assume now that

$$b_2 = [1, 0, \ldots, 0]^T.$$

Then $Ab_2 = [1, a_1, 0, \ldots, 0]^T$ supplies only the information about the first row (and the first column) of the matrix A. Similarly knowing $Ab, \ldots A^i b$, we know the first i rows (and columns) of the matrix A. Since the off-diagonal elements of the jth row, $j = i+1, i+2, \ldots, n-1$, are unknown, it is easy to conclude that for sufficiently small ε we have

(4.8) $k(F, b_2) = n$.

Thus, for the same value of ε, $k(F, b)$ can be equal to unity for some b as in (4.7) and can be equal to n for a different b as in (4.8). This illustrates that if F is not orthogonally invariant, $k(F, b)$ depends on b. ∎

5. MATRICES WITH BOUNDED CONDITION NUMBER.

In this section we deal with three orthogonally invariant classes of matrices defined as

$$(5.1)\qquad F_1 = \{A : A = A^T > 0,\ \mathrm{cond}(A) \le M\},$$

$$(5.2)\qquad F_2 = \{A : A = A^T,\ \mathrm{cond}(A) \le M\},$$

$$(5.3)\qquad F_3 = \{A : \mathrm{cond}(A) \le M\}$$

where $\mathrm{cond}(A) = \|A\|\ \|A^{-1}\|$ is the condition number and M is a given number such that $M \ge 1$. That is, F_1 is the class of symmetric positive definite matrices with condition number bounded by M, F_2 differs from F_1 by the lack of positive definiteness and F_3 differs from F_2 by the lack of symmetry. The case of most interest is when M is large since such problems arise in applications and are difficult to solve.

Our main interest in this section is to find the optimal class indices for the three classes and to see how the lack of positive definiteness and the lack of symmetry increase the optimal class index. We find the optimal class index by computing the class index of the mr algorithm and using Corollary 4.1. We are ready to prove

<u>Theorem 5.1</u>

Let F_i be defined by (5.i) for $i = 1, 2, 3$. Then

$$(5.4)\qquad k(F_1)-a_1 = k(\phi^{mr}, F_1) = \min\left(n, \left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{\sqrt{M}+1}{\sqrt{M}-1} \right\rfloor + 1\right).$$

(5.5) $k(F_2)-a_2 = k(\phi^{mr}, F_2) = \min(n, 2 \lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} / \ln \frac{M+1}{M-1} \rfloor + 2)$,

(5.6) $k(F_3) = k(\phi^{mr}, F_3) = n$

where $a_i = 0$ or $a_i = -1$ for $i = 1, 2$. ∎

Proof

Let $x_k = \phi_k^{mr}(N_k(A,b))$ be the sequence generated by the mr algorithm. Assume first that $M = 1$ and $A \in F_2$. Then $A = cI$ for some nonzero constant c. Since $Ab = cb$, $x_1 = \frac{1}{c}b$ and $Ax_1 - b = 0$. Thus $k(F_1) = k(F_2) = k(\phi^{mr}, F_1) = k(\phi^{mr}, F_2) = 1$ which agrees with (5.4) and (5.5) for $a_1 = a_2 = 0$. Hence, without loss of generality we assume $M > 1$ in the proof of (5.4) and (5.5).

(i) We first prove (5.4). It is well-known that for symmetric positive definite matrices the mr algorithm converges at least as fast as the Chebyshev algorithm, i.e.,

(5.7) $\| Ax_k - b \| \leq 2\rho^k / (1 + \rho^{2k})$, $k < n$.

where $\rho = (\sqrt{M} - 1)/(\sqrt{M} + 1)$. (To show (5.7) it is enough to define

$$P(t) = 1 - T_k(f(t))/T_k(f(0))$$

where $f(t) = (2t - \|A^{-1}\|^{-1} - \|A\|)/(\|A\| - \|A^{-1}\|^{-1})$ and T_k is the Chebyshev polynomial of degree k, and next apply (3.4).)

It is also known that (5.7) is sharp, i.e. there exists a matrix A from F_1 such that we have equality in (5.7). For completeness we sketch the construction of such a matrix A. Recall that

$$(5.8) \qquad \sum_{i=1}^{k+1}{}'' \; T_j(z_i)\, T_s(z_i) = 0\,, \qquad j < s \le k$$

for

$$z_i = \cos\left(\frac{\pi(k+1-i)}{k}\right), \quad i = 1, 2, \ldots, k+1,$$

where Σ'' denotes a finite sum whose first and last terms are to be halved. Let

$$(5.9) \qquad \lambda_i = c\,[\,M+1 + (M-1)\,z_i\,]/\,2\,, \quad i = 1, 2, \ldots, k+1,$$

with

$$c = 2 \sum_{i=1}^{k+1}{}'' \,[\,M+1 + (M-1)\,z_i\,]^{-1}\,.$$

Then $\lambda_1 = c < \lambda_2 < \ldots < \lambda_{k+1} = cM$, $\lambda_{k+1}/\lambda_1 = M$. Further let

$$\lambda = [\,\frac{1}{\sqrt{2}}\,\lambda_1^{-1/2}\,,\ \lambda_2^{-1/2}\,,\ \ldots,\ \lambda_k^{-1/2}\,,\ \frac{1}{\sqrt{2}}\,\lambda_{k+1}^{-1/2}\,,\ 0, \ldots, 0$$

Note that $\|\lambda\| = 1$. Define $Q = [\xi_1, \xi_2, \ldots, \xi_n]$ as an orthogonal matrix such that $Q\lambda = -b$. Finally let

$$(5.10) \qquad A = Q\,\mathrm{diag}(\lambda_1, \ldots, \lambda_{k+1}, \ldots, \lambda_{k+1})\;Q^T\,.$$

Clearly $A = A^T > 0$ and $\mathrm{cond}(A) = M$. Thus $A \in F_1$. Note that $A\xi_i = \lambda_i\,\xi_i$ for $i = 1, 2, \ldots, k+1$ and

$$b = -(\frac{1}{\sqrt{2}}\,\lambda_1^{-1/2}\,\xi_1 + \lambda_2^{-1/2}\,\xi_2 + \ldots + \lambda_k^{-1/2}\,\xi_k + \frac{1}{\sqrt{2}}\,\lambda_{k+1}^{-1/2}\,\xi_{k+1})\,.$$

Let

$$W_j(t) = \frac{T_j(f(t))}{T_j(f(0))}\,, \quad f(t) = (2t - c(M+1))/(c(M-1))\,.$$

Then $W_j(0) = 1$. Set $m = k+1$,

$c_1 = -\frac{1}{\sqrt{2}} \lambda_1^{-1/2}$, $c_2 = -\lambda_2^{-1/2}, \ldots,$ $c_k = -\lambda_k^{-1/2}$ and $c_{k+1} = -\frac{1}{\sqrt{2}} \lambda_{k+1}^{-1/2}$, in (3.5). Then (W_j, W_s) is proportional to

$$\sum_{i=1}^{k+1}{}'' T_j(f(\lambda_i))\, T_s(f(\lambda_i)) = \sum_{i=1}^{k+1}{}'' T_j(z_i)\, T_s(z_i) = 0$$

for $j < s \le k$, due to (5.8). Hence $\{W_j\}$ is orthogonal and W_k is a unique solution of (3.4). Thus

$$\text{(5.11)} \qquad \| Ax_k - b \|^2 = \| W_k(A) b \|^2 = \sum_{i=1}^{k+1}{}'' \lambda_i^{-1}\, T_k^2(f(\lambda_i)) / T_k^2(f(0)) =$$
$$= (2\rho^k / (1 + \rho^{2k}))^2$$

since $T_k^2(f(\lambda_i)) = T_k^2(z_i) = 1$ and $\sum_{i=1}^{k+1}{}'' \lambda_1^{-1} = \| \lambda \|^2 = 1$. This proves that (5.7) is sharp.

Let k^* be the smallest integer such that

$$2\rho^{k^*} / (1 + \rho^{2k^*}) < \varepsilon \ .$$

Then

$$k^* = \left\lfloor \ln \frac{1 + \sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{\sqrt{M}+1}{\sqrt{M}-1} \right\rfloor + 1 .$$

Let $k = k(\phi^{mr}, F_1)$. Note that if $k^* \le n$ then $k = k^*$. Indeed, $k < k^*$ implies $k < n$ and we can find a matrix A from F_1 such that

$$\| Ax_k - b \| = 2\rho^k / (1 + \rho^{2k}) \ge \varepsilon \ .$$

This is a contradiction. Thus $k = k^*$. Since $x_n = \alpha$, k is at most n. This and Corollary 4.1 proves (5.4).

(ii) We now prove (5.5). Let $p = \lfloor k/2 \rfloor$. Since A is symmetric, then A^2 is positive definite. Assume that all eigenvalues of A^2 lie in $[c_1, c_2]$. Then $c_1 > 0$ and since $\mathrm{cond}(A) \le M$ we conclude $c_2/c_1 \le M^2$. Define

$$P(t) = 1 - \frac{T_p(f(t^2))}{T_p(f(0))}, \quad f(t) = (2t - c_1 - c_2)/(c_2 - c_1).$$

Note that $P(0) = 0$ and P is an even polynomial of degree $2p \le k$. From (3.4) we conclude

$$\text{(5.12)} \qquad \| Ax_k - b \| \le \| (I - P(A))b \| \le 2\rho^p/(1 + \rho^{2p})$$

where now $\rho = (M-1)/(M+1)$. Assuming that

$$\text{(5.13)} \qquad 2\lfloor k/2 \rfloor \le n - 2$$

we construct a matrix A for which we achieve equality in (5.12). Similarily to (5.9) define

$$\lambda_i = c[M^2 + 1 + (M^2 - 1) z_i]/2, \quad i = 1, 2, \ldots, p+1$$

with $c = 2 \sum_{i=1}^{p+1} \| [M^2 + 1 + (M^2 - 1) z_i]^{-1}$, and $z_i = \cos(\pi(p + 1 - i)/p)$. (If $p = 0$ we define $z_1 = -1$.) Then for $p \ge 1$ we have $\lambda_1 = c < \lambda_2 < \ldots < \lambda_{p+1} = cM^2$, $\lambda_{p+1}/\lambda_1 = M^2$. Define the $(p+1) \times 1$ vector d as

$$d = \left[\frac{1}{\sqrt{2}} \lambda_1^{-1/2}, \lambda_2^{-1/2}, \ldots, \lambda_p^{-1/2}, \frac{1}{\sqrt{2}} \lambda_{p+1}^{-1/2} \right]^T .$$

Next let λ be a $n \times 1$ vector defined as

$$\lambda = \frac{1}{\sqrt{2}} [d, d, 0, \ldots, 0]^T .$$

Note that $\| \lambda \| = 1$. Further let Q be an orthogonal matrix such that $Q\lambda = -b$. Finally let

$$(5.14) \qquad A = Q\,\mathrm{diag}(\sqrt{\lambda_1},\ldots,\sqrt{\lambda_{p+1}},-\sqrt{\lambda_1},\ldots,-\sqrt{\lambda_{p+1}},\ldots,-\sqrt{\lambda_{p+1}})Q^T .$$

Clearly, $A = A^T$. For $p \ge 1$, $\mathrm{cond}(A) = \sqrt{\lambda_{p+1}/\lambda_1} = M$ and for $p = 0$, $\mathrm{cond}(A) = 1$. Thus $A \in F_2$. Note that

$$m_j \stackrel{\mathrm{df}}{=} (A^{2j}b,b) = \sum_{i=1}^{p+1}{}'' \lambda_i^{j-1} ,$$

$$(A^{2j+1}b,b) = 0.$$

It is straightforward to verify that the solution c^* of (3.3) is given by

$$c_1^* = c_3^* = \ldots = c^*_{2\lceil k/2\rceil-1} = 0$$

and the coefficients c^*_{2i} satisfy the system

$$\begin{pmatrix} m_2, & m_3, \ldots, & m_{p+1} \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ m_{p+1}, & \cdots & m_{2p} \end{pmatrix} \begin{pmatrix} c_2 \\ c_4 \\ \cdot \\ \cdot \\ c_{2p} \end{pmatrix} = \begin{pmatrix} m_1 \\ m_2 \\ \cdot \\ \cdot \\ m_p \end{pmatrix} .$$

For $p=0$ we have $k=1$ and $x_1=0$ which proves that (5.12) is sharp in this case. For $p \ge 1$, we get the same system as for the symmetric positive definite case with k replaced by p and M replaced by M^2. From this we conclude that

$$Ax_k - b = -W_k(A^2)\,b$$

where

$$W_k(t) = \frac{T_p(f(t))}{T_p(f(0))}, \quad f(t) = (2t - c(M^2+1))/(c(M^2-1)).$$

As in (i) we have

$$\| Ax_k - b \|^2 = \sum_{i=1}^{p+1} \lambda_i^{-1} T_p^2(f(\lambda_i))/T_p^2(f(0)) =$$

$$= (2\rho^p/(1+\rho^{2p}))^2 .$$

This proves that (5.12) is sharp as long as (5.13) holds.

Let k^* be the minimal integer such that

$$2\rho^{p^*}/(1+\rho^{2p^*}) < \varepsilon \qquad \text{with} \quad p^* = \lfloor k^*/2 \rfloor .$$

Then

$$k^* = 2 \left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{M+1}{M-1} \right\rfloor + 2 .$$

Let $k = k(\phi^{mr}, F_2)$. Note that if $k^* \le n$ then $k = k^*$. Indeed, $k < k^*$ implies $2\lfloor k/2 \rfloor \le n-2$ and we can find a matrix A from F_2 such that

$$\| Ax_k - b \| = 2\rho^p/(1+\rho^{2p})$$

where $p = \lfloor k/2 \rfloor < k^*/2 = p^*$. Thus $\| Ax_k - b \| \ge \varepsilon$ which is a contradiction. Hence $k = k^*$. Since $x_n = \alpha$, k is at most n. This and Corollary 4.1 proves (5.5).

(iii) We finally prove (5.6). Let $b = [1, 0, \ldots, 0]^T$ and

$$A_1 = \begin{pmatrix} 0 & & & 1 \\ 1 & & & \\ & 1 & \ddots & \\ & & 1 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & & & -1 \\ 1 & & & \\ & 1 & \ddots & \\ & & 1 & 0 \end{pmatrix}$$

Observe that A_i is orthogonal, $\mathrm{cond}(A_i) = 1$. Thus $A_i \in F_3$. Further

$$A_1^i\, b = A_2^i\, b = [0,\ldots,0,\underset{i}{1},0,\ldots,0]^T$$

for $i < n$. Let $\phi = \{\phi_k\}$ be any algorithm and let ξ_n denote the n-th component of $x_k = \phi_k(N_k(A,b))$, $k < n$. Then

$$\max(\| A_1 x_k - b \|, \| A_2 x_k - b \|) \geq$$

$$\max(| \xi_n - 1 |, |\xi_n + 1 |) \geq 1 \geq \varepsilon.$$

This proves that $k(\phi,A) \geq n$. Since $k(F_3)$ is independent of b due to Lemma 4.2, we conclude

$$k(F_3) = k(\phi^{mr}, F_3) = n.$$

This proves (5.6) and completes the proof of the theorem. ∎

Theorem 5.1 states how the optimal class index depends on ε and M. For small value of ε and large values of M we can simplify (5.4) and (5.5) to

$$(5.15) \qquad k(F_1) = \min\left(n, \frac{\sqrt{M}}{2} \ln \frac{2}{\varepsilon}(1 + o(1))\right) + a_1$$

$$(5.16) \qquad k(F_2) = \min\left(n, M \ln \frac{2}{\varepsilon}(1 + o(1))\right) + a_2$$

<u>Remark 5.1</u>

In typical applications there is a relation between n, M and ε. For instance, if one approximates a two dimensional elliptic differential equation then the corresponding matrix is symmetric and positive definite with $M \cong n$. One usually sets $\varepsilon = n^{-1}$ which yields

$$k(F_1) \cong \frac{\sqrt{n}}{2} \ln 2n . \qquad \blacksquare$$

If n is sufficiently large, i.e., if the minimum in (5.5) is attained for the second argument, then

$$(5.17) \qquad \frac{k(F_2)}{k(F_1)} = \frac{2 \left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{M+1}{M-1} \right\rfloor + 2 + a_2}{\left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{\sqrt{M}+1}{\sqrt{M}-1} \right\rfloor + 1 + a_1} = 2\sqrt{M}(1 + o(1))$$

This shows that the lack of positive definitness increases the optimal class index roughly $2\sqrt{M}$ times. For large M, which arise frequently in practice, this is a very significant difference.

We discuss Theorem 5.1 for the class F_3. The theorem states that if fewer than n matrix-vector multiplications are permitted it is impossible to find an ε-approximation no matter what algorithm is used. Note that this result holds for arbitrary ε, and M, i.e., ε and M can even be equal to unity. It is the lack of symmetry which causes the increase of the optimal class index to its maximal value n.

Remark 5.2

Using a similar proof technique it is possible to show that (5.6) holds for much more general adaptive information operators. Namely, assume that

(5.18) $\qquad N_k(A,b) = [b, Az_1, Az_2, \ldots, Az_k]$

where $z_i = z_i(b, Az_1, Az_2, \ldots, Az_{i-1})$ is an arbitrary function of b and previously computed information. Then for any algorithm $\phi = \{\phi_k\}$ there exists a matrix A from F_3 such that

(5.19) $\qquad \| A\,\phi_k(N_k(A,b)) - b \| \geq 1, \quad \forall\, k < n.$

This means even the adaptive information (5.18) is too weak to find an ε-approximation using less than n steps. Once more (5.19) holds for arbitrary ε and M. See section 9 for a general discussion of adaptive information. ∎

We summarize this discussion in

Corollary 5.1

For small ε, large M and

$$n \geq 2\left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{M+1}{M-1} \right\rfloor + 2$$

we have

$$k(F_1) = \frac{\sqrt{M}}{2} \ln \frac{2}{\varepsilon} \quad (1 + o(1)),$$

$$\frac{k(F_2)}{k(F_1)} = 2\sqrt{M} \;(1 + o(1)),$$

$$\frac{k(F_3)}{k(F_1)} = \frac{2n}{\sqrt{M} \ln \frac{2}{\varepsilon}} \quad (1 + o(1)). \qquad ∎$$

6. MATRICES OF THE FORM $A = I - B$.

In this section we consider two additional orthogonally invariant classes of matrices. For these we find the class index of the mr algorithm and the optimal class index. We also show when the Chebyshev algorithm and the successive approximation algorithm are optimal.

Let

$$(6.1) \qquad F_4 = \{A : A = I - B, \quad B = B^T, \quad \|B\| \le \rho < 1\},$$

$$(6.2) \qquad F_5 = \{A : A = I - B, \qquad \|B\| \le \rho < 1\}.$$

Thus F_4 is the class of symmetric positive definite matrices of the form $I - B$ where the norm of B is bounded by a known constant ρ, $\rho < 1$. The class F_5 differs from F_4 by the nonsymmetry of the matrices B. Of course, $F_4 \subset F_5$. Note that for $\rho = 0$, $F_4 = F_5 = \{I\}$. To exclude this trivial case we assume that $\rho > 0$.

<u>Remark 6.1</u>

Observe that

$$(6.3) \qquad \operatorname{cond}(A) \le (1+\rho)/(1-\rho), \qquad \forall A \in F_5,$$

and (6.3) is sharp. This establishes a relation between the class F_4 and the class F_1 with $M = (1+\rho)/(1-\rho)$. Note, however, that if $A \in F_4$ then $\|A\| \le 1 + \rho$ and $\|A^{-1}\| \le (1-\rho)^{-1}$. These bounds do not hold for matrices from F_1. The class F_5

is related to the class F_3 with the same $M = (1+\rho)/(1-\rho)$. The difference between F_5 and F_3 appears if M goes to unity, i.e., ρ goes to zero. Then F_5 contains only the identity matrix whereas F_3 contains matrices of the form cQ where c is a real constant and Q is an orthogonal matrix. ∎

We first find the class index of the mr algorithm for the two classes F_4 and F_5.

Theorem 6.1

Let F_4 and F_5 be given by (6.1) and (6.2). Then

$$(6.4)\qquad k(\phi^{mr}, F_4) = \min(n, \lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} / \ln \frac{1+\sqrt{1-\rho^2}}{\rho} \rfloor + 1),$$

$$(6.5)\qquad k(\phi^{mr}, F_5) = \min(n, \lfloor \frac{\ln \varepsilon}{\ln \rho} (1-\delta) \rfloor + 1)$$

where $\delta = \delta(\varepsilon,\rho)$ satisfies

$$(6.6)\qquad 0 \le \delta \le \frac{1}{2} \frac{\ln(1-\rho^2+\rho^2\varepsilon^2)}{\ln \varepsilon} .$$ ∎

Proof

(i) We prove (6.4). Let $M = (1+\rho)/(1-\rho)$. Define the matrix A by (5.10). Let

$$B = I - \frac{1-\rho}{c} A$$

where c is given by (5.9). Then $B = B^T$ and $\|B\| = \rho$. Thus $A_1 = I - B = \frac{1-\rho}{c} A$ belongs to F_4. Then $x_k = \phi_k^{mr}(N_k(A_1,b)) = \frac{c}{1-\rho} \phi_k^{mr}(N_k(A,b))$ and (5.11) yields

$$\|A_1 x_k - b\| = \|A\phi_k^{mr}(N_k(A,b)) - b\| = 2\rho_1^k/(1+\rho_1^{2k})$$

where $\rho_1 = (\sqrt{M}-1)/(\sqrt{M}+1)$.

Thus the class index of the mr algorithm for the class F_4 is the same as for the class F_1 with $M = (1+\rho)/(1-\rho)$. Since

$$\frac{\sqrt{M}+1}{\sqrt{M}-1} = \frac{1+\sqrt{1-\rho^2}}{\rho}$$

(6.4) follows from (5.4).

(ii) To prove (6.5) observe that knowing $A^i b$ we also know $B^i b$, $B = I - A$, for $i = 0, 1, \ldots, k$. Thus the algorithm

$$x_k' = \phi_k(N_k(A,b)) = b + Bb + \ldots + B^{k-1}b$$

is well defined and

$$\| Ax_k' - b \| = \| B^k b \| \le \rho^k .$$

Since x_k' lies in the space spanned by $b, Ab, \ldots, A^{k-1}b$, then

$$\| Ax_k - b \| \le \| Ax_k' - b \| \le \rho^k \tag{6.7}$$

where $x_k = \phi_k^{mr}(N_k(A,b))$. We now find a lower bound on $\| Ax_k - b \|$ for $k < n$.

Let $b = [1, 0, \ldots, 0]^T$ and $B = \rho Q$ where

$$Q = \begin{pmatrix} 0 & & & 1 \\ 1 & 0 & & \\ & 1 & \ddots & \\ & & 1 & 0 \end{pmatrix}.$$

Then

$$B^i b = \rho^i [0, \ldots, 0, \underset{i+1}{1}, 0, \ldots, 0]^T \quad i < n.$$

Since $A = I - B$, (3.4) easily yields that

$$\| Ax_k - b \| = \min_{P_k(1)=1} \| P_k(B)\, b \|$$

where $P_k(t) = p_o + p_1 t + \ldots + p_k t^k$ is a polynomial of degree at most k and $p_o + p_1 + \ldots + p_k = 1$. Since the $B^i b$ are orthogonal, $i = 0, 1, \ldots, k$, then

$$\| P_k(B) b \|^2 = p_o^2 + p_1^2 \rho^2 + \ldots + p_k^2 \rho^{2k}.$$

By a standard technique we can show that

(6.8) $$\|Ax_k - b\| = \min_{p_0+\dots+p_k=1} \sqrt{(p_0^2+\dots+p_k^2\,\rho^{2k})} =$$

$$= \rho^k \sqrt{\frac{1-\rho^2}{1-\rho^{2(k+1)}}} .$$

Compare with (6.7).

Let k be the smallest integer such that $\| Ax_k - b \| < \varepsilon$. From (6.7) and (6.8) it easily follows that

$$k \le \min(n, \lfloor \ln \varepsilon / \ln \rho \rfloor + 1) ,$$

$$k \ge \min(n, \lfloor \frac{\ln \varepsilon}{\ln \rho} (1 - \frac{1}{2} \frac{\ln(1-\rho^2+\varepsilon^2\rho^2)}{\ln \varepsilon}) \rfloor + 1) .$$

This proves (6.5) and (6.6) and completes the proof of the theorem. ∎

To find the optimal class index for the class F_4 we use Theorem 6.1 and the properties of the Chebyshev algorithm. It is known that the Chebyshev algorithm ϕ^{ch} applied to the system $x = Bx + b$, with $A = I - B \in F_4$, constructs the sequence $\{x_k\}$ such as

(6.9) $$b - Ax_k = \frac{T_{k+1}(\frac{1}{\rho}B)}{T_{k+1}(\frac{1}{\rho})} b .$$

The vector x_k can be computed from the recurrence conditions

$$x_{-1} = 0 , \quad x_0 = b ,$$

(6.10) $$x_{i+1} = c_{i+1}(Bx_i + b - x_{i-1}) + x_{i-1},$$

$$c_0 = 2 , \quad c_{i+1} = \frac{1}{1 - \frac{\rho^2}{4} c_i}$$

for $i = 0, 1, \ldots$. Note that x_k depends on b, $Bb, \ldots, B^k b$ or equivalently on $N_k(A,b)$. Thus

$$x_k = \phi_k^{ch}(N_k(A,b)).$$

Remark 6.2

Note that the Chebyshev algorithm is not well defined for the class F_1. Indeed, if $A \in F_1$ then the norm of $B = I - A$ can be larger than unity. ∎

Let

$$(6.11) \qquad q(\varepsilon) = \left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{1+\sqrt{1-\rho^2}}{\rho} \right\rfloor .$$

We find the class index of the Chebyshev algorithm.

Lemma 6.1

$$k(\phi^{ch}, F_4) = q(\varepsilon) .$$

∎

Proof

From (6.9) we have

$$(6.12) \qquad \| b - Ax_k \| \le |T_{k+1}(1/\rho)|^{-1} = 2\rho_2^{k+1} / (1 + \rho_2^{2(k+1)})$$

where $\rho_2 = \rho/(1 + \sqrt{1-\rho^2})$. To show that (6.12) is sharp assume that $Bb = \rho b$. Then (6.9) implies

$$b - Ax_k = \{T_{k+1}(1)/ T_{k+1}(1/\rho)\} b$$

Since $T_{k+1}(1) = 1$ we obtain the desired result. Note that the smallest k for which $|T_{k+1}(1/\rho)|^{-1} < \varepsilon$ is equal to $q(\varepsilon)$.

This proves Lemma 6.1. ∎

We are now ready to derive the optimal class index.

Theorem 6.2

$$k(F_4) = \min(n, q(\varepsilon)).$$ ∎

Proof

Assume first that $q = q(\varepsilon) < n$. Then (6.4) and Corollary (4.1) yield

$$k(\phi^{mr}, F_4) = q + 1 \le k(F_4) + 1.$$

Since $k(F_4) \le k(\phi^{ch}, F_4)$, Lemma 6.1 gives $k(F_4) = q$.

If $q \ge n$, $k(\phi^{mr}, F_4) = n \le k(F_4) + 1$ which yields $k(F_4) \ge n-1$. We defer the proof that $k(F_4) = n$ until Section 7, Theorem 7.3. ∎

We obtain the optimality properties of the Chebyshev algorithm.

Theorem 6.3

(i) The Chebyshev algorithm is

optimal if $q(\varepsilon) \le n$, $k(\phi^{ch}, F_4) = k(F_4) = q(\varepsilon)$,

not optimal if $q(\varepsilon) > n$, $k(\phi^{ch}, F_4) - k(F_4) = q(\varepsilon) - n$.

(ii) The Chebyshev algorithm is not strongly optimal. More precisely there exists a matrix $A \in F_4$ such that

$$k(\phi^{ch}, A) - k(A) = k(\phi^{ch}, F_4) - 1 = q(\varepsilon) - 1.$$ ∎

Proof

Conclusion (i) follows directly from Lemma 6.1 and Theorem 6.2. To prove (ii) set $A = I - B$ where $Bb = \rho b$. Then $k(A) = 1$ and from the proof of Lemma 6.1 it follows that $k(\phi^{ch}, A) = q(\varepsilon) = k(\phi^{ch}, F_4)$. ∎

Note that the assumption $q(\varepsilon) \le n$ implies that ε is not too small relative to ρ and n. For small ε, ρ close to unity, and n so large that $q(\varepsilon) < n$, we have

$$(6.13) \qquad k(F_4) = k(\phi^{ch}, F_4) = k(\phi^{mr}, F_4) - 1 = \frac{\ln \frac{2}{\varepsilon}}{\sqrt{2(1-\rho)}}(1 + o(1)).$$

which corresponds to $k(F_1)$ with $M = (1+\rho)/(1-\rho)$.

We now proceed to the class F_5. First of all observe that we do not have the exact class index of the minimal residual algorithm since an unknown δ appears in (6.5). Note that $\delta = \delta(\varepsilon, \rho)$ goes to zero with ε for fixed ρ. However, if ρ goes to unity with fixed ε, then $\delta \in (0,1)$ and (6.5) is not useful. It is possible to improve (6.5) but we do not pursue this here.

For sufficiently small ε, $\delta = o(1)$ and (6.5) can be written as

$$(6.14) \qquad k(\phi^{mr}, F_5) = \min\left(n, \left\lfloor \frac{\ln \varepsilon}{\ln \rho}(1 + o(1)) \right\rfloor + 1\right).$$

From Corollary 4.1 we find

$$(6.15) \qquad k(F_5) = \min\left(n, \left\lfloor \frac{\ln \varepsilon}{\ln \rho}(1 + o(1)) \right\rfloor + 1\right) + a_3$$

where $a_3 = 0$ or $a_3 = -1$. If, additionally, ρ is close to unity and n is so large that the minimum in (6.5) is attained for the second argument, we have

$$(6.16) \qquad k(F_5) = \frac{\ln \frac{1}{\varepsilon}}{1 - \rho} (1 + o(1)).$$

Note that for the corresponding class F_3 we always have $k(F_3) = n$. Comparing (6.16) with (6.13) we see that

$$(6.17) \qquad \frac{k(F_5)}{k(F_4)} = \sqrt{\frac{2}{1-\rho}} (1 + o(1)).$$

This shows how the lack of symmetry increase the optimal class index.

We now show that the *successive approximation* algorithm ϕ^{sa} is asymptotically optimal for the class F_5. This algorithm constructs the sequence $\{x_k\}$ as

$$(6.18) \qquad \begin{aligned} x_0 &= b \\ x_{i+1} &= Bx_i + b, \quad i = 0, 1, \ldots . \end{aligned}$$

Thus x_k depends on $b, Bb, \ldots, B^k b$ and $x_k = \phi_k^{sa}(N_k(A,b))$ with $A = I - B$. Obviously

$$\| b - Ax_k \| = \| B^{k+1} b \| \le \rho^{k+1} .$$

Note that this estimate is sharp since for $Bb = \rho b$ we get equality. This proves that the class index of ϕ^{sa} is the smallest k for which $\rho^{k+1} < \varepsilon$. Thus

$$(6.19) \qquad k(\phi^{sa}, F_5) = \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

Comparing with (6.16) we have

$$\frac{k(\phi^{sa},F_5)}{k(F_5)} = 1 + o(1).$$

for small ε and large n. This shows that the successive approximation algorithm is asymptotically optimal. As was the case for the Chebyshev algorithm, the algorithm ϕ^{sa} is <u>not</u> strongly optimal since for $A = I - B$, where $Bb = \rho b$, we have

$$k(\phi^{sa},A) - k(A) = k(\phi^{sa},F_5) - 1 = \lceil \ln \varepsilon / \ln \rho \rceil - 1 .$$

We summarize these properties in

<u>Theorem 6.4</u>

The successive approximation algorithm ϕ^{sa} is asymptotically optimal for small ε and large n,

$$k(F_5) \cong k(\phi^{sa},F_5) = \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

The algorithm ϕ^{sa} is <u>not</u> strongly optimal. ∎

The importance of our optimality result concerning F_5 is that in numerical practice the linear system $Mx = g$ is often transformed into $x = Bx + b$. Examples of such transformation are the Richardson, Jacobi, Gauss-Seidel and SOR algorithms. Our result states that asymptotically the transformed system with a nonsymmetric matrix B is best solved by the successive approximation algorithm.

7. GENERALIZED CRITERIA

In this section we introduce a family of approximation criteria depending on a parameter p. The criterion used in Sections 2-6 corresponds to $p = 1$. The values of p of greatest practical importance are $p = 0, 1/2, 1$.

A lower bound on the optimal matrix index is obtained for any orthogonally invariant class and for any value of p. For some values of p, we define a "generalized minimal residual algorithm" for which this lower bound is almost achieved. We next find the optimal class indices for the class F_4 with arbitrary p and for the class F_5 with $p = 0$. We establish the optimality of the Chebyshev algorithm for F_4 with any p and the optimality of the successive approximation algorithm for F_5 with $p = 0$.

In (2.1) we defined an ε-approximation as a vector whose residual has norm less than ε. Here we assume that the ε-approximation x satisfies the inequality

$$(7.1) \qquad \frac{\| A^p(x - \alpha) \|}{\| A^p \alpha \|} < \varepsilon$$

where $\alpha = A^{-1}b$ and p is a nonnegative real. Note that for $p = 1$, (7.1) coincides with (2.1). For $p = 0$, (7.1) means that the relative error of x is less than ε.

If p is not an integer we assume that A is symmetric and positive definite to guarantee the existence of A^p.

We generalize the concept of the matrix index of ϕ to

$$(7.2) \qquad k(\phi,A) = \min\{k : \|\tilde{A}^p(x_k - \tilde{A}^{-1}b)\| / \|\tilde{A}^{p-1}b\| < \varepsilon, \ \forall \tilde{A} \in V(y_k)\}$$

where $\phi = \{\phi_k\}$, $x_k = \phi_k(N_k A, b))$ and $V(y_k)$ is given by (2.4). (If the set of k is empty, we set $k(\phi,A) = +\infty$.) Then all concepts introduced in Section 2 may be generalized in an obvious way using the new definition of the matrix index of ϕ.

For given A and m define the coefficients $c_0^*, c_1^*, \ldots, c_m^*$ and the error $e(A,m)$ as

$$(7.3) \qquad e(A,m) = \| A^p(\alpha - c_0^* b - \ldots - c_m^* A^m b) \| = \min_{c_i} \| A^p(\alpha - c_0 b - \ldots - c_m A^m b)$$

Let

$$m(A) = \min\{m : e(\tilde{A},m)/\|\tilde{A}^p \tilde{\alpha}\| < \varepsilon, \ \forall \tilde{A} \in V(y_m)\}$$

where $\tilde{\alpha} = \tilde{A}^{-1}b$. We prove

<u>Theorem 7.1</u>

If F is orthogonally invariant then

$$(7.4) \qquad k(A) \geq m(A), \quad \forall A \in F. \qquad \blacksquare$$

<u>Proof</u>

As in the proof of theorem 4.1 let $\phi = \{\phi_k\}$ be any algorithm such that $k = k(\phi,A) < +\infty$. This means

$$(7.5) \qquad \|\tilde{A}^p(x_k' - \tilde{\alpha})\| / \|\tilde{A}^p \tilde{\alpha}\| < \varepsilon, \ \forall \tilde{A} \in V(y_k),$$

where $x_k' = \phi_k(N_k(A,b))$. Decompose

$$x_k' = z_1 + z_2$$

where $z_1 \in \mathrm{lin}(b, Ab, \ldots, A^k b)$ and z_2 is orthogonal to $b, Ab, \ldots, A^k b$. Define $\tilde{A}_1 = Q\,\tilde{A}\,Q$ with $Q = I - 2\omega\omega^T$ and $\omega = z_2 / \|z_2\|$ for a nonzero z_2 and $\omega = 0$ for $z_2 = 0$. Then $\tilde{A}_1 \in F$ and $\tilde{A}_1^i b = A^i b$, $i = 1, 2, \ldots, k$. Thus $\tilde{A}_1 \in V(y_k)$. Observe that

$$(7.6) \qquad \tilde{\alpha}_1 = Q\tilde{\alpha} \quad \text{and} \quad \|\tilde{A}_1^p\,\tilde{\alpha}_1\| = \|\tilde{A}^p\,\tilde{\alpha}\|\,.$$

Furthermore,

$$(7.7) \qquad \|\tilde{A}_1^p(x_k' - \tilde{\alpha}_1)\| = \|\tilde{A}^p\,Q(z_1 - \tilde{\alpha} + z_2 + 2(\omega,\tilde{\alpha})\omega)\| =$$

$$= \|\tilde{A}^p(z_1 - \tilde{\alpha}) + \tilde{A}^p z_2 + 2((\omega,\tilde{\alpha}) - (\omega, z_1 - \tilde{\alpha} + z_2 + 2(\omega,\tilde{\alpha})\omega))\tilde{A}^p\omega\|$$

$$= \|\tilde{A}^p(z_1 - \tilde{\alpha}) + \tilde{A}^p z_2 - 2(\omega, z_2)\tilde{A}^p\omega\| = \|\tilde{A}^p(z_1 - \tilde{\alpha}) - \tilde{A}^p z_2\|\,.$$

From (7.5), (7.6) and (7.7) we get

$$\frac{e(\tilde{A},k)}{\|\tilde{A}^p\,\tilde{\alpha}\|} \le \frac{\|\tilde{A}^p(z_1 - \tilde{\alpha})\|}{\|\tilde{A}^p\,\tilde{\alpha}\|} \le \frac{1}{2}\left(\frac{\|\tilde{A}^p(z_1 - \tilde{\alpha}) - \tilde{A}^p z_2\|}{\|\tilde{A}^p\,\tilde{\alpha}\|} + \frac{\|\tilde{A}^p(z_1 - \tilde{\alpha}) + \tilde{A}^p z_2\|}{\|\tilde{A}^p\,\tilde{\alpha}\|}\right)$$

$$\le \frac{1}{2}\left(\frac{\|\tilde{A}_1^p(x_k' - \tilde{\alpha}_1)\|}{\|\tilde{A}_1^p\,\tilde{\alpha}_1\|} + \frac{\|\tilde{A}^p(x_k' - \tilde{\alpha})\|}{\|\tilde{A}^p\,\tilde{\alpha}\|}\right) < \varepsilon\,.$$

Thus $k \ge m(A)$. Since ϕ is an arbitrary algorithm we conclude $k(A) \ge m(A)$. Hence (7.4) is proven. ∎

Theorem 7.1 provides a lower bound on the optimal matrix index. The next part of this section is devoted to finding algorithms whose class indices are close to this lower bound.

As we shall see, this can only be done for certain values of p.

We check when the coefficients c_i^* defined by (7.3) can be computed in terms of the information $N_k(A,b)$. From (7.3) it follows that $c^* = [c_0^*, c_1^*, \ldots, c_m^*]^T$ satisfies the linear equations

$$(7.8) \qquad Hc^* = h$$

where $H = ((A^{i+p}b, A^{j+p}b))$, $i, j = 0, 1, \ldots, m$, and $h = [(A^p b, A^{p-1}b), \ldots, (A^{m+p}b, A^{p-1}b)]^T$.

We consider two cases.

(i) $A = A^T$. Then if $2p$ is integer, $2p \geq 1$ and $m = k - \lceil p \rceil$, the vector c^* depends only on $N_k(A,b)$.

(ii) $A \neq A^T$. Then if p is integer, $p \geq 1$ and $m = k - p$, the vector c^* depends only on $N_k(A,b)$.

If either (i) or (ii) holds then the algorithm $\phi^{mr} = \{\phi_k^{mr}\}$,

$$(7.9) \qquad x_k = \phi_k^{mr}(N_k(A,b)) = c_o^* b + \ldots + c_{k-\lceil p \rceil}^* A^{k-\lceil p \rceil} b$$

is well defined and is called the <u>generalized minimal residual</u> algorithm.

Note that for $p = 1$, (7.9) coincides with (3.2). Assuming that $A = A^T > 0$ we can set $p = 1/2$ and the algorithm ϕ^{mr} is known as the <u>classical conjugate gradient</u> algorithm. See for instance Stiefel [58]. In this case one of the possible ways to compute x_k is as follows.

Let $x_0 = 0$. For $i = 0, 1, \ldots, k-1$ define

(7.10) $$x_{i+1} = x_i + \frac{1}{q_i}\{f_{i-1}(x_i - x_{i-1}) - r_i\}, \quad r_i = Ax_i - b,$$

where

$$q_i = \frac{(Ar_i, r_i)}{(r_i, r_i)} - f_{i-1}$$

(7.11) $$f_{-1} = 0, \quad f_{i-1} = \frac{(r_i, r_i)}{(r_{i-1}, r_{i-1})} q_{i-1}.$$

Compare with (3.7) and (3.8).

For $p = 0$ and $A = A^T$ the first component (b,α) of the vector h is in general unknown. If, however, one considers the consistent system $Mx = g$ and if one agrees to multiply this system by M^T then $A = M^TM$, $b = M^Tg$, and $(b,\alpha) = (g,g)$ is computable. Then the generalized minimal residual algorithm is well defined and is known as the <u>minimum</u> <u>error</u> algorithm. In this case we can compute x_k as follows. Let $x_0 = 0$. For $i = 0, 1, \ldots, k-1$ define

(7.12) $$x_{i+1} = x_i + \frac{1}{q_i}\{f_{i-1}(x_i - x_{i-1}) - r_i\}, \quad r_i = Ax_i - b,$$

where

$$q_i = \frac{(r_i, r_i)}{||Mx_i - g||^2} - f_{i-1},$$

(7.13) $$f_{-1} = 0, \quad f_{i-1} = \frac{||Mx_i - g||^2}{||Mx_{i-1} - g||^2} q_{i-1}$$

We are ready to show that the generalized minimal residual algorithm is almost strongly optimal.

Theorem 7.2

Let F be orthogonally invariant. Suppose that the following two conditions hold:

(i) If $A \in F$ implies $A = A^T$, $\forall A \in F$, then $2p$ is an integer, otherwise p is an integer.

(ii) If (b, α) is known and $A \in F$ implies $A = A^T, \forall A \in F$, then $p \geq 0$, otherwise $p > 0$.

Then the generalized minimal algorithm is almost strongly optimal,

$$(7.14) \qquad k(A) + a = k(\phi^{mr}, A) = m(A) + \lceil p \rceil, \quad \forall A \in F,$$

where a is an integer from $[0, \lceil p \rceil]$. ∎

Proof

Note first that (i) and (ii) guarantee that the algorithm ϕ^{mr} is well defined. From (7.3) and (7.9) we have

$$\| A^p(\alpha - x_k) \| = e(A, k - \lceil p \rceil).$$

Thus

$$k(\phi^{mr}, A) = m(A) + \lceil p \rceil.$$

Obviously $k(\phi^{mr}, A) \geq k(A)$ which due to (7.4) yields

$$0 \leq a = k(\phi^{mr}, A) - k(A) \leq \lceil p \rceil.$$

This proves (7.14). ∎

Observe that for $p = 1$, the conditions (i) and (ii) are always satisfied and Theorem 7.2 coincides with Theorem 4.1.

For $p=1/2$, Theorem 7.2 states that the classical conjugate gradient algorithm is almost strongly optimal and the matrix index of the classical conjugate gradient differs by at most unity from the optimal matrix index.

If p can be set equal to zero, then (7.14) states that

$$k(A) = k(\phi^{mr},A) = m(A)$$

Thus, the minimum error algorithm is strongly optimal.

We now end this section by finding the optimal class index $k(F)$ for the class $F=F_4$ for arbitrary p, and for the class $F=F_5$ with $p=0$. We also indicate which algorithms are optimal but not strongly optimal. Recall that

$$q(\varepsilon) = \left\lfloor \ln \frac{\varepsilon}{1-\sqrt{1-\varepsilon^2}} \Big/ \ln \frac{1+\sqrt{1-\rho^2}}{\rho} \right\rfloor.$$

Theorem 7.3

Let $F=F_4$ be defined by (6.1). Then for arbitrary p,

$$k(F_4) = \min(n,\ q(\varepsilon)).$$

Let the ϕ^{ch} be Chebyshev algorithm defined by (6.10). Then

$$k(\phi^{ch},\ F_4) = q(\varepsilon).$$

If $q(\varepsilon) \le n$ then the Chebyshev algorithm is optimal but not strongly optimal.

If $q(\varepsilon) > n$ then the Chebyshev algorithm is not optimal. ∎

Proof

Based on the proofs of Theorems 5.1 and 6.1 we can show that for arbitrary p,

$$m(F_4) \overset{\underline{df}}{=} \max_{A \in F_4} m(A) = \min(n,\ q(\varepsilon)).$$

From (6.9) it is obvious that

$$\|A^p(\alpha - \phi^{ch}(N_k(A,b)))\| \le 2\|A^p\alpha\|\ \rho_2^{k+1}/(1+\rho_2^{2(k+1)}),\ \forall A \in F_4\ ,$$

where $\rho_2 = \rho/(1+\sqrt{1-\rho^2})$. Thus

$$k(\phi^{ch},\ F_4) \le q(\varepsilon).$$

If $q(\varepsilon) \le n$ then

$$k(F_4) \le k(\phi^{ch},\ F_4) \le q(\varepsilon) = m(F_4) \le k(F_4)$$

due to Theorem 7.1. This proves the optimality of the Chebyshev algorithm for $q(\varepsilon) \le n$.

We show that $k(\phi^{ch}, F_4) = q(\varepsilon)$. As in the proof of Lemma 6.1 define $A = I - B$ with $Bb = \rho b$. Observe that

$$\alpha - \phi^{ch}(N_k(A,b)) = \{T_{k+1}(1)/T_{k+1}(1/\rho)\}\ \alpha$$

Then $q(\varepsilon) = k(\phi^{ch}, A) \le k(\phi^{ch}, F_4) \le q(\varepsilon)$ which yields the needed result.

Since $k(A) = 1$, we have $k(\phi^{ch}, A) - k(A) = q(\varepsilon) - 1$ which proves that the Chebyshev iteration is not strongly optimal. Finally, note that $k(F_4)$ is at most n which completes the proof of Theorem 7.3. ∎

<u>Remark 7.1</u>

Observe that the proof of Theorem 7.3 for $p = 1$ completes the proof of Theorem 6.2. ∎

For the class $F = F_5$ with $p = 0$ we prove

Theorem 7.4

Let $F = F_5$ be defined by (6.2) and let $p = 0$. Then

$$k(F_5) = \min(n, \lfloor \ln \varepsilon / \ln \rho \rfloor).$$

Let ϕ^{sa} be the successive approximation algorithm defined by (6.18). Then

$$k(\phi^{sa}, F_5) = \lfloor \ln \varepsilon / \ln \rho \rfloor.$$

If $\lfloor \ln \varepsilon / \ln \rho \rfloor \le n$ then the successive approximation algorithm is optimal but not strongly optimal.

If $\lfloor \ln \varepsilon / \ln \rho \rfloor > n$ then the successive approximation algorithm is not optimal. ∎

Proof

Let $x_k = \phi_k^{sa}(N_k(A,b))$ with $A = I - B$. Since

$$\|\alpha - x_k\| = \|B^{k+1}\alpha\| \le \rho^{k+1}\|\alpha\|,$$

we have

$$k(F_5) \le k(\phi^{sa}, F_5) \le \lfloor \ln \varepsilon / \ln \rho \rfloor. \tag{7.15}$$

We show that (7.15) is sharp for $\lfloor \ln \varepsilon / \ln \rho \rfloor \le n$. Let $k < n$ and $A = I - B$, $\tilde{A} = I - \tilde{B}$, where

$$B = \rho \begin{bmatrix} Q & \\ & I \end{bmatrix}, \quad \tilde{B} = \rho \begin{bmatrix} \tilde{Q} & \\ & I \end{bmatrix}$$

where Q and $\tilde{Q}$ are $(k+1) \times (k+1)$ matrices and I is the (n-k-1) x (n-k-1) identity matrix. Let

$$Q = \begin{bmatrix} 0 & & & & 0 \\ 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & 0 \end{bmatrix} \qquad \tilde{Q} = \begin{bmatrix} 0 & & & & -1 \\ 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & 0 \end{bmatrix}$$

Observe that $\|B\| = \|\tilde{B}\| = \rho$ which implies that $A, \tilde{A} \in F_5$. Furthermore $B^i b = \tilde{B}^i b$, $i = 1, 2, \ldots, k$, for $b = [1, 0, \ldots, 0]^T$. Thus $A^i b = \tilde{A}^i b$, $i = 1, 2, \ldots, k$. Let $\alpha = A^{-1} b$ and $\tilde{\alpha} = \tilde{A}^{-1} b$. Then it is easy to show that

$$\tilde{\alpha} = \frac{1 - \rho^{k+1}}{1 + \rho^{k+1}} \alpha . \tag{7.16}$$

Let ϕ be an arbitrary algorithm, $x_k = \phi_k(N_k(A, b))$ and let $k = k(\phi, A) < n$. Then

$$\max \left(\frac{\|x_k - \alpha\|}{\|\alpha\|}, \frac{\|x_k - \tilde{\alpha}\|}{\|\tilde{\alpha}\|} \right) < \varepsilon . \tag{7.17}$$

From (7.16) and (7.17) we have

$$\frac{2\rho^{k+1}}{1 + \rho^{k+1}} \|\alpha\| = \|\tilde{\alpha} - \alpha\| \le \|x_k - \tilde{\alpha}\| + \|x_k - \alpha\| < \varepsilon(\|\tilde{\alpha}\| + \|\alpha\|)$$

$$= \frac{2\varepsilon}{1 + \rho^{k+1}} \|\alpha\| .$$

Thus $\rho^{k+1} < \varepsilon$, which implies

$$k = k(\phi, A) \ge \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

Since ϕ is arbitrary,

$$k(F_5) \ge k(A) \ge \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

From (7.15) we get

$$k(F_5) = k(\phi^{sa}, F_5) = \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

as long as $\lfloor \ln \varepsilon/\ln \rho \rfloor \le n$.

This proves that the algorithm ϕ^{sa} is optimal.

Define $A = I - B$ where $Bb = \rho b$. Then $k(\phi^{sa}, A) = \lfloor \ln \varepsilon/\ln \rho \rfloor$.
This and (7.15) proves that

$$k(\phi^{sa}, F_5) = \lfloor \ln \varepsilon / \ln \rho \rfloor .$$

Since $k(A) = 1$, we have

$$k(\phi^{sa}, A) - k(A) = k(\phi^{sa}, F_5) - 1 = \lfloor \ln \varepsilon / \ln \rho \rfloor - 1$$

which proves that the algorithm ϕ^{sa} is not strongly optimal.

Note that $k(F_5)$ is at most n which completes the proof of Theorem 7.4. ∎

For $p = 1$ we established only the asymptotic behavior of $k(F_5)$. For $p = 0$ we have the exact value of $k(F_5)$.

8. COMPLEXITY

We have given lower and upper bounds on the optimal matrix and class indices for computing an ε-approximation. We show how these results can be employed to bound complexity (minimal cost) of finding an ε-approximation.

We first outline our model of computation. For simplicity, let the cost of each arithmetic operation be unity. We assume that the cost of one matrix-vector multiplication Ax, for an arbitrary vector x, is cn. Note that $c = c(A)$ depends on the structure of A and for sparse matrices c is usually proportional to unity rather than to n. In this paper we discuss algorithms depending on the information $N_k(A,b) = [b, Ab, \ldots, A^k b]$. As noted above this does not necessarily means that we actually compute $A^i b$, $i = 1, 2, \ldots, k$. Rather it means that we compute Az_i, $i = 1, 2, \ldots, k$, where z_i is a linear combination of $b, Ab, \ldots, A^{i-1}b$. For any choice of z_i, we perform k matrix-vector multiplications and we therefore assume that the cost of $N_k(A,b)$ is kcn.

Let $\phi = \{\phi_k\}$ be an algorithm. To find $x_k = \phi_k(N_k(A,b))$, given $y_k = N_k(A,b)$, we compute $\phi_k(y_k)$. Let $d(\phi,k)$ denote the combinatory complexity of ϕ, i.e. the cost of combining the information y_k to produce x_k. Note that y_k represents $(k+1)n$ scalar data. We postulate that the algorithm ϕ

uses every scalar piece of data at least once and therefore

(8.1) $\quad d(\phi,k) \geq kn, \ \forall\phi.$

If the combinatory complexity of ϕ is <u>linear</u> in the total number of scalar data of $N_k(A,b)$, i.e., $d(\phi,k) \leq c_1 kn$ for some "small" constant c_1 independent of A, then $d(\phi,k)$ is close to minimal.

Let $\text{comp}(\phi,A)$ denote the cost of finding an ε-approximation, i.e. the cost of computing $x_k = \phi_k(N_k(A,b))$ such that $\| \tilde{A}x_k - b \| < \varepsilon$ (or $\| \tilde{A}^p(x_k - \tilde{\alpha}) \| / \| \tilde{A}^{p-1}b \| < \varepsilon$ if criterion (7.1) is used) for every matrix $\tilde{A}$ which has the same information as A, $\tilde{A} \in V(y_k)$. By definition we have to perform $k(\phi,A)$ matrix-vector multiplications to find an ε-approximation. Thus

(8.2) $\quad \text{comp}(\phi,A) = c(A)n\, k(\phi,A) + d(\phi,k(\phi,A))$

<u>Remark 8.1</u>

The quantities defined in this section also depend on ε, b, N_k, and F. We remind the reader that for simplicity we do not exhibit this dependence in our notation or terminology. ∎

Due to (8.1) we have

(8.3) $\quad \text{comp}(\phi,A) \geq (c(A) + 1)n\, k(\phi,A).$

We seek algorithms with minimal complexity. Define

(8.4) $\quad \text{comp}(A) = \min\limits_{\phi} \text{comp}(\phi,A).$

Since $k(\phi,A) \geq k(A)$, (8.3) yields

(8.5) $$\mathrm{comp}(A) \geq (c(A) + 1)n\, k(A).$$

Thus equations (8.4) and (8.5) motivate the following definition.

An algorithm ϕ is <u>an optimal complexity algorithm for A</u> iff

(8.6) $$\mathrm{comp}(\phi,A) = \mathrm{comp}(A)$$

and ϕ is an <u>almost optimal complexity algorithm for A</u> iff there exist two small integers c_1 and c_2 such that

(8.7) $$\mathrm{comp}(\phi,A) \leq (c(A) + c_1)n(k(A) + c_2).$$

Due to (8.5), $c_1 \geq 1$ and $c_2 \geq 0$. In many cases, $k(A)$ is much larger than c_2 and $c(A)$ is much larger than c_1. This yields

(8.8) $$\mathrm{comp}(A) \cong \mathrm{comp}(\phi,A) \cong c(A)n\, k(A).$$

We are ready to prove

<u>Theorem 8.1</u>

An almost strongly optimal algorithm with linear combinatory complexity is an almost optimal complexity algorithm for every A from F. ∎

<u>Proof</u>

If ϕ is an almost strongly optimal algorithm then

$k(\phi,A) \le k(A) + c_2$, $\forall\, A \in F$, where c_2 is a small integer due to (2.11). The algorithm ϕ has also linear combinatory complexity, $d(\phi,k) \le c_1 kn$. Thus

$$\mathrm{comp}(\phi,A) \le (c(A) + c_1)n\ (k(A) + c_2)$$

which agrees with (8.7). ∎

We proved that the minimal residual algorithm ϕ^{mr} is almost strongly optimal for orthogonally invariant classes. See Corollary 4.1. We now consider the combinatory complexity of ϕ^{mr}. For the classes $F = F_1$ or $F = F_4$, the algorithm ϕ^{mr} is defined by (3.7) and (3.8). From this it is obvious that its combinatory complexity is linear with $c_1 \le 14 + 2/n$. For the classes $F = F_2$ or $F = F_4$, the combinatory complexity of ϕ^{mr} is also linear due to, as noted in Section 3, a fast algorithm for the solution of any linear system with a Toeplitz matrix.

Observe that the Chebyshev algorithm ϕ^{mr} defined by (6.10) for the class $F = F_4$ also has linear combinatory complexity with $c_1 \le 4 + 5/n$. The algorithm ϕ^{ch} is <u>not</u> an almost optimal complexity algorithm for <u>every</u> A. Theorem 6.3 states that ϕ^{ch} is optimal whenever $q(\varepsilon) \le n$ and then $k(\phi^{ch},F_4) = k(F_4)$. Thus, if A is such that $k(A)$ is close to $k(F_4)$, then the Chebyshev algorithm is an almost optimal complexity algorithm for A.

Similarily, for different criteria as defined by (7.1) we conclude that for the class $F = F_1$ or $F = F_4$ with $p = 1/2$, the classical conjugate gradient algorithm is an almost optimal complexity algorithm. The Chebyshev algorithm is an almost optimal complexity algorithm for matrices A such that $k(A)$ is close to $k(F)$ for $F = F_4$ with arbitrary p, whenever $q(\varepsilon) \le n$.

Finally, for the class $F = F_5$ with $p = 0$, observe that the successive approximation algorithm ϕ^{sa} defined by (6.18) has combinatory complexity equal to kn which due to (8.1) is minimal. Theorem 7.4 states that ϕ^{sa} is optimal whenever $\lceil \ln \varepsilon / \ln \rho \rceil \le n$. Thus, for matrices such that $k(A) = k(F_5)$ the algorithm ϕ^{sa} is an optimal complexity algorithm.

We summarize these results in

Theorem 8.2

(i) The minimal residual algorithm is an almost optimal complexity algorithm for every matrix A from the classes F_1, F_2 and F_4 with $p = 1$.

(ii) The Chebyshev algorithm is an almost optimal algorithm for matrices A from the class F_4 for arbitrary p whenever $q(\varepsilon) \le n$ and $k(A)$ is close to $k(F_4)$.

(iii) The classical conjugate gradient algorithm is an almost optimal complexity algorithm for every matrix A from the classes F_1 and F_4 with $p = 1/2$.

(iv) The successive approximation algorithm is an optimal complexity algorithm for matrices A from the class

F_5 with $p = 0$ whenever $\lfloor \ln \varepsilon / \ln \rho \rfloor \le n$ and $k(A) = k(F_5)$. ∎

9. COMPARISON WITH DIRECT ALGORITHMS

The results of this paper enable us to compare direct algorithms for the solution of linear equations with optimal (or nearly optimal) algorithms using the information $N_k(A,b)$. This comparison can be done for different classes of matrices and with mathematical rigor. Here, however, we confine ourselves to the comparison of the Gauss elimination algorithm and the minimal residual algorithm for dense and sparse matrices from the class F_1, i.e. for the class of symmetric and positive definite matrices with condition number bounded by M.

We first consider the cost of the arithmetic operations and then briefly discuss storage requirements. Assume A is known. We discuss the case of dense matrices. Then the Gauss elimination algorithm requires $n^3/3 + O(n^2)$ arithmetic operations to find the exact solution of $Ax = b$. (We neglect the effect of roundoff errors.) For simplicity we assume that the cost of the Gauss elimination algorithm is $n^3/3$, $\mathrm{comp}(G) = n^3/3$, taking the cost of each arithmetic operations as unity.

Even if the matrix A is known, it can be more efficient to use "partial" information $N_k(A,b)$ and apply the minimal residual algorithm. Of course, in this case instead of the exact solution we want to find an ε-approximation (in the sense of (2.1)). Section 8 yields that the cost of the mr algorithm is

$$\mathrm{comp}(\phi^{mr}) \doteq (c(A) + c_1)n(\left\lfloor \ln \frac{1 + \sqrt{1 - \varepsilon^2}}{\varepsilon} / \ln \frac{\sqrt{M} + 1}{\sqrt{M} - 1} \right\rfloor + 1).$$

Since A is dense, $c(A)$ is proportional to n. Then $c(A) + c_1 \leq 2n + O(1)$. For simplicity we omit the lower order term and conclude that if

$$\text{(9.1)} \qquad \left\lfloor \ln \frac{1 + \sqrt{1 - {}^2}}{\varepsilon} / \ln \frac{\sqrt{M} + 1}{\sqrt{M} - 1} \right\rfloor < \frac{n}{6} - 1,$$

then $\mathrm{comp}(\phi^{mr}) < \mathrm{comp}(G)$. Equation (9.1) exhibits the relation between ε, M and n which guarantees that the mr algorithm is more efficient than the Gauss elimination algorithm. Note that for $n \geq 7$, (9.1) holds provided that either ε is not too small or that M is not too large. Thus if we want to find an ε-approximation with moderate ε or if the condition number of the system $Ax = b$ is moderate then the minimal residual algorithm is superior to the Gauss elimination algorithm.

We now discuss the case of sparse matrices. The cost of the Gauss elimination algorithm (in fact the cost of any direct algorithm) for sparse matrices depends critically on the structure of A. For some favorable cases, the cost is proportional to n, for some "bad" cases, it can be still proportional to n^3. To include all cases assume that for the sparse case the cost of the Gauss elimination algorithm is

$$\mathrm{comp}(G) = c_2\, n^{\beta}$$

for some $\beta \in [1,3]$ and a positive c_2.

For the mr algorithm, set $c_3 = c(A) + c_1$. Since A is sparse, c_3 is of order unity. If

$$(9.2)\qquad \left\lfloor \ln \frac{1+\sqrt{1-\varepsilon^2}}{\varepsilon} \Big/ \ln \frac{\sqrt{M}+1}{\sqrt{M}-1} \right\rfloor < \frac{c_2}{c_3}\, n^{\beta} - 1$$

then $\mathrm{comp}(\phi^{mr}) < \mathrm{comp}(G)$. If $c_2 n^{\beta}/c_3 - 1 > 0$, then (9.2) holds provided that either ε is not too small or that M not too large. In these cases, the mr algorithm is superior to the Gauss elimination algorithm.

For example, set $c_2 = 1$, $\beta = 2$, and $c_3 = 20$. Approximating the logarithms, (9.2) can be simplified to

$$(9.3)\qquad \frac{\sqrt{M}}{2} \ln \frac{2}{\varepsilon} < \frac{n^2}{20} - 1 .$$

Then the mr algorithm is superior to the Gauss elimination algorithm provided (9.3) holds.

We now compare storage requirements. As above we distinguish between dense and sparse matrices. For dense matrices, the Gauss elimination algorithm requires storage proportional to n^2. The mr algorithm uses storage proportional to n plus storage required to compute Ax. Therefore, if Ax can be computed with storage less than n^2, the mr algorithm is superior. For example, if A can be generated, then storage is proportional to n.

For sparse matrices, the storage required by the Gauss elimination algorithm depends critically on the structure of A and may vary from n to n^2. On the other hand, the storage of the mr algorithm is always proportional to n.

10. OPEN PROBLEMS

In this paper we studied optimal algorithms for the solution of $Ax = b$ using the information operator $N_k(A,b) = [b, Ab, \ldots, A^k b]$. We have focused on this information operator because it is widely used in practice and because it is susceptible to a very thorough analysis. It would of course be desirable to generalize results of this paper to more general information operators. Until this is accomplished we won't know if $N_k(A,b)$ is "optimal" information.

For instance, let

$$N_k(A,b) = [b, Az_1, Az_2, \ldots, Az_k] \tag{10.1}$$

where $z_i = z_i(b, Az_1, \ldots, Az_{i-1})$ for $i = 1, 2, \ldots, k$. That is, we still compute the matrix-vector multiplications but now the vector z_i is an arbitrary function of the previously computed information. For information (10.1) we can generalize the definition of the optimal matrix and class indices in an obvious way. We ask what is the optimal choice of the z_i, i.e., for which z_i are the optimal indices minimized. We propose

<u>Conjecture 10.1</u>

If F is orthogonally invariant then the optimal matrix and class indices are minimized for the vectors $z_i = A^{i-1}b$, $i = 1, 2, \ldots, k$. That is, the information $N_k(A,b) = [b, Ab, \ldots, A^k b]$ is optimal in the class of information operators of the form (10.1). ∎

We now consider more general information operators than (10.1). Let

(10.2) $\qquad N_s(A,b) = [b,\, L_1(A;b),\, L_2(A;b,u_1),\ldots,\, L_s(A;b,u_1,\ldots,u_{s-1})$,

where $u_i = L_i(A;b,u_1,\ldots,u_{i-1})$, $i = 1, 2,\ldots, s-1$, and L_i is a functional which depends linearly on the first argument. The L_i can depend nonlinearly on b and on the previously computed information $u_1, u_2,\ldots, u_{i-1}$. Note that (10.2) is the general form of adaptive linear information and (10.1) as well as (2.2) are special examples of (10.2). We ask what is the optimal adaptive linear information, i.e. what functionals L_i minimize the optimal matrix and class indices. In would also be interesting to know the minimal value of s for which we can find the exact solution of a linear system. From Rabin [72] we can conclude that $s \le (n+1)(n+2)/2 - 1$ with no restriction on the class F.

We also want to pose a complexity problem. We showed that for the information $N_k(A,b) = [b, Ab,\ldots, A^k b]$ there exist algorithms which are optimal (or almost optimal) and which have linear combinatory complexity. These two properties guarantee finding an ε-approximation with minimal (or almost minimal) complexity.

Let $N_s(A,b)$ be an optimal adaptive linear information of the form (10.2). Does there exist an almost optimal algorithm using $N_s(A,b)$ with linear combinatory complexity? Or conversely, is it true that if an information operator is better that $N_k(A,b) = [b, Ab, \ldots, A^k b]$, then the combinatory complexity of an almost optimal algorithm cannot be linear?

We can establish one result for $N_s(A,b)$. The functionals L_i in (10.2) must depend on b. Otherwise the information $N_s(A,b)$ does not supply enough knowledge to find an ε-approximation. To show this assume that

$$(10.3) \qquad N_s(A,b) = [b, L_1(A), L_2(A;u_1), \ldots, L_s(A;u_1, \ldots, u_s),$$

where $u_i = L_i(A;u_1,\ldots,u_{i-1})$ is independent of b. As in (2.8), let $k(F)$ be the minimal value of s such that there exists an algorithm which uses $N_s(A,b)$ and finds an ε-approximation in the sense of (7.1).

For simplicity we establish the desired result only for the class F_4. Without loss of generality we assume that $\varepsilon \le \rho$. (Otherwise the algorithm $\phi_s(N_s(A,b)) = b$ yields an ε-approximation.)

<u>Theorem 10.1</u>

Let $\varepsilon \le \rho$, $F = F_4$ and p be arbitrary. There exists a vector b such that

$$k(F_4) \ge \frac{n(n+1)}{2} . \qquad \blacksquare$$

<u>Proof</u>

Let $A = I + B$ where

$$(10.4) \qquad L_i(B, u_1, \ldots u_{i-1}) = 0, \quad i = 1, 2, \ldots, s,$$

and $u_i = L_i(I, u_1, \ldots, u_{i-1})$. Note that (10.4) corresponds to s homogenous linear equations in coefficients of B. Since B is an nxn symmetric matrix, we have $n(n+1)/2$ unknowns. If $s < n(n+1)/2$ then there exists a nonzero matrix B satisfying (10.4). We can normalize B such that $\|B\| = \rho$. Define a vector b such that $Bb = cb$ with $c = \pm\rho$. Let $\tilde{A} = I - B$. Then $\tilde{A} \in F_4$ and $N_s(\tilde{A},b) = N_s(A,b)$. Let $\phi = \{\phi_k\}$ be an algorithm and $x_k = \phi_k(N_k(A,b))$. Let

$$a = \max\left(\frac{\|\tilde{A}^p(x_k - \tilde{\alpha})\|}{\|\tilde{A}^p\,\tilde{\alpha}\|}\ ,\ \frac{\|A^p(x_k - \alpha)\|}{\|A^p\,\alpha\|}\right)$$

where $\alpha = A^{-1}b$ and $\tilde{\alpha} = A^{-1}b$. Then $\alpha = \frac{1}{1+c}\,b$, $\|A^p\alpha\| = (1+c)^{p-1}$, $\tilde{\alpha} = \frac{1}{1-c}\,b$ and $\|\tilde{A}\tilde{\alpha}\| = (1-c)^{p-1}$. Let

$$x_k = c_1 b + x$$

where $c_1 = (x_k, b)$ and x is orthogonal to b. Then $((I \pm B)^p x, b) = 0$ and

$$\|(I \pm B)^p(x_k - \frac{1}{1\pm c}\,b)\|^2 \geq |\,c_1(1 \pm c)^p - (1 \pm c)^{p-1}|.$$

Thus

$$a \geq \max(|\,c_1(1+\rho) - 1|,\ |\,c_1(1-\rho) - 1|) \geq \rho \geq \varepsilon\,.$$

Since ϕ is arbitrary, this proves that it is impossible to find an ε-approximation for $s < n(n+1)/2$. This completes the proof. ∎

Note that for the class F_4, we can recover the matrix $A = (a_{kj})$ knowing a suitable chosen $N_s(A,b)$ with $s = n(n+1)/2$.

REFERENCES

Brent [78] — Brent, R.P., private communication.

Rabin [72] — Rabin, M.O., "Solving Linear Equations by Means of Scalar Products", in Complexity of Computer Computations, edited by R.E. Miller and J.W. Thatche Plenum Press, New York, 1972, 11-20.

Stiefel [58] — Stiefel, E., "Kernel Polynomials in Linear Algebra and Their Numerical Applications", NBS. Appl. Math., 43 1958, 1-22.

Traub and Woźniakowski [80] — Traub, J.F. and Woźniakowski, H., A General Theory of Optimal Algorithms, Academic Press, New York, 1980.

Woźniakowski [80] — Woźniakowski, M., "Roundoff-Error Analysis of a New Class of Conjugate-Gradient Algorithms", Linear Algebra and Its Applications, 1980.

Yun and Gustavson [79] — Yun, D.Y. and Gustavson, F.G., "Fast Computation of the Rational Hermite Interpolant and Solving Toeplitz Systems of Equations via the Extended Euclidean Algorithm", Symbolic and Algebraic Computation, Lecture Notes in Computer Science No. 72, edited by E.W. Ng, Springer Verlag, 1979, 58-64.

DATE
FILMED
-8