A new look at CMRH and its relation to GMRES

Hassane Sadok and Daniel B. Szyld

Report 10-04-25
April 2010
Revised July 2011

Department of Mathematics
Temple University
Philadelphia, PA 19122

This report is available in the World Wide Web at
http://www.math.temple.edu/~szyld

A NEW LOOK AT CMRH AND ITS RELATION TO GMRES*

HASSANE SADOK† AND DANIEL B. SZYLD‡

Abstract. CMRH is a Krylov subspace method which uses the Hessenberg process to produce a basis of a Krylov method, and minimizes a quasiresidual. This method produces convergence curves which are very close to those of GMRES, but using fewer operations and storage. In this paper we present new analysis which explains why CMRH has this good convergence behavior. Numerical examples illustrate the new bounds.

1. Introduction. CMRH (Changing Minimal Residual method based on the Hessenberg process), introduced in [9], is a Krylov subspace method for the solution of $n \times n$ linear systems of the form

$$Ax = b. \tag{1.1}$$

This iterative method uses the Hessenberg process to compute at the kth step, a basis $\{\ell_1, \ell_2, \ldots, \ell_k\}$ of the Krylov subspace

$$\mathcal{K}_k = \mathcal{K}_k(A, r_0) = \text{span}\{r_0, Ar_0, A^2r_0, \ldots, A^{k-1}r_0\}, \tag{1.2}$$

where $r_0 = b - Ax_0$, is the initial residual and x_0 is the initial vector. Furthermore, this basis is such that the matrix $L_k = [\ell_1, \ell_2, \ldots, \ell_k]$ is unit lower triangular (which some times is called trapezoidal); for more details on the Hessenberg process, see, e.g., [12, §§6.28–6.30]. The CMRH approximation to the solution of (1.1) at the kth step is $x_k^c = x_0 + z_k^c$, where $z_k^c \in \mathcal{K}_k$ is the solution of the following constrained minimization problem

$$\min_{u \in \mathbb{R}^{k+1},\ z \in \mathcal{K}_k(A, r_0)} \|u\|, \quad \text{subject to} \quad A\,z = r_0 + L_{k+1}u, \tag{1.3}$$

and the norm $\|\cdot\|$ here and throughout the paper is the Euclidean norm. For more details, see Section 3 and [9].

The CMRH method shares many of the computational properties of the well-known GMRES method; see Section 2 and, e.g., [7, 8, 11] for details on GMRES. Some of the properties shared by these two methods include the fact that the matrix A is only needed as an operator for a matrix-vector product, that only one matrix-vector product is needed at each iteration, and that it is easy to update the kth approximation by solving a (small) $(k+1) \times k$ least squares problem. One advantage of CMRH is that, at the $k+1$st iteration, the matrix-vector product with A is $A\ell_k$, and since the first $k-1$ entries of ℓ_k are zeros, this product can be implemented without the need to access the first k columns of A, with possible savings in the cost of this matrix-vector product, especially when the matrix is not so sparse. This fact also allows for an implementation in which the first k columns of the matrix A are used to store the basis vectors of L_k (in the case that A is full) [5]. We note that overwriting the columns of A during an Hessenberg process is common when

*This revision dated 16 July 2011.

†Laboratoire de Mathématiques Pures et Appliquées, Université du Littoral, Centre Universitaire de la Mi-Voix, B.P. 699, 62228 Calais Cedex, France (sadok@lmpa.univ-littoral.fr).

‡Department of Mathematics, Temple University (038-16), 1805 N. Broad Street, Philadelphia, Pennsylvania 19122-6094, USA (szyld@temple.edu). Research supported in part by the U.S. Department of Energy under grant DE-FG02-05ER25672.

computing eigenvalues. The use of iterative methods, such as GMRES (or CMRH), for the solution of dense problems can be very beneficial; see, e.g., [1] and some of the experiments in Section 5.

In [5] it was shown that CMRH is indeed a valid alternative to GMRES for certain situations. These include the cases where the matrix A is explicitly stored and are not very sparse, e.g., those arising from the discretization of boundary element methods or integral equations. Furthermore, it has been observed that in practice, CMRH behaves very much like GMRES, in the sense that if GMRES exhibits superlinear converges, so does CMRH for the same problem, and if GMRES stagnates, CMRH does so as well. Furthermore, the convergence curves for the the two methods are always very near each other (see Section 5 and [9]), and we note that CMRH can be implemented with fewer operations per step than GMRES. We have found no case in which CMRH suffer from instability. On the contrary, as shown in Section 5, there are cases where the error of the approximate solution of GMRES starts to grow, while the computed residual keeps decreasing. For the same example, no such behavior is observed for CMRH: the behavior of the error reflects that of the computed residual.

In this paper we present some new analysis and perspectives of CMRH which helps us understand why this method works so well. In particular, we explain why the convergence behavior of CMRH is similar to that of GMRES. This is mostly done in Section 4, where we present explicitly an upper triangular matrix R_k which gives the change of basis between the Arnoldi basis and the Hessenberg basis of the Krylov subspace. The same matrix is used to show that, except for a rank one matrix, the two resulting upper Hessenberg matrices are spectrally equivalent. In fact, at termination, the two matrices have the same eigenvalues. Furthermore, the ratio between the kth CMRH residual and the kth GMRES residual is bounded by the condition number of R_k. In Section 5 we present some numerical experiments illustrating our theoretical observations.

Throughout the paper, unless otherwise noted, $\|x\|$ refers to the Euclidean norm, and $\langle x, y\rangle$ to the Euclidean inner product. By I_k we denote the $k \times k$ identity matrix, with columns denoted $e_1^{(k)}, \ldots, e_k^{(k)}$. When the length of these vectors is clear from the context, we drop the superscript.

2. The GMRES method. We review the GMRES method by considering first a non-traditional approach to the Arnoldi basis. For the usual description of Arnoldi and of GMRES see further below and also, e.g., [7, 8, 11]. Let

$$K_k = [r_0, Ar_0, A^2 r_0, \ldots, A^{k-1} r_0] \tag{2.1}$$

be the $n \times k$ *Krylov matrix*. It follows that $K_{k+1} = [r_0, AK_k]$. Consider the QR factorization of

$$K_k = V_k \tilde{R}_k, \tag{2.2}$$

where $V_k \in \mathbb{R}^{n\times k}$ is such that $V_k^T V_k = I_k$, and $\tilde{R}_k$ is upper triangular. In other words, the columns of V_k form an orthonormal basis of $\mathcal{K}_k$.

We can then write

$$K_{k+1} \begin{bmatrix} 0 \\ \\ I_k \end{bmatrix} = V_{k+1} \tilde{R}_{k+1} \begin{bmatrix} 0 \\ \\ I_k \end{bmatrix} = A\, K_k = A\, V_k\, \tilde{R}_k. \tag{2.3}$$

Since $\tilde{R}_k^{-1}$ is also upper triangular, from (2.3) we can define the following $(k+1)\times k$ upper Hessenberg matrix with the following relations

$$H_{k+1,k} = \tilde{R}_{k+1} \begin{bmatrix} 0_{1\times k} \\ \\ I_k \end{bmatrix} \tilde{R}_k^{-1} = V_{k+1}^H A V_k = \begin{bmatrix} H_k \\ \\ v_{k+1}^H A V_k \end{bmatrix} = \begin{bmatrix} H_k \\ \\ h_{k+1,k} \left(e_k^{(k)}\right)^T \end{bmatrix},$$

and $H_k \equiv V_k^T A V_k$ is a square upper Hessenberg matrix of order k representing a restriction and projection of the operator A to $\mathcal{K}_k$. Moreover we have the following *Arnoldi relation* as

$$A\,V_k = V_{k+1} H_{k+1,k} = V_k\,H_k + h_{k+1,k}\,v_{k+1}\,e_k^T. \tag{2.4}$$

The Arnoldi process to determine V_k begins with $v_1 = r_0/\|r_0\|$, and at step $j+1$, Av_j is orthogonalized with respect to $v_1, v_2, \ldots, v_j$, thus obtaining a vector in the direction of the new vector v_{j+1}. The orthogonal coefficients $h_{ji} = \langle Av_j, v_i\rangle$, $i = 1, \ldots, j$, are precisely the entries of the upper Hessenberg matrix $H_{k+1,k}$. The entry $h_{j+1,j}$ is such that v_{j+1} thus has unit norm.

GMRES is a minimum residual method, and the approximation to the solution of (1.1), $x_k \in x_0 + \mathcal{K}_k$, is such that the norm of the residual $r_k = b - Ax_k$ is minimum over all such vectors. In other words, we are looking for $x_k = x_0 + z_k$, $z_k \in \mathcal{K}_k$, or equivalently $x_k = x_0 + V_k y_k$, $y_k \in \mathbb{R}^k$ which solve the following minimization problems

$$\min_{y\in\mathbb{R}^k} \|b - A\,(x_0 + V_k y)\| = \min_{y\in\mathbb{R}^k} \|\beta e_1^{(k+1)} - H_{k+1,k}\,y\| = \tag{2.5}$$

$$\min_{y\in\mathbb{R}^k} \|r_0 - AV_k y\| = \min_{z\in\mathcal{K}_k} \|r_0 - Az\| = \min_{z\in\mathcal{K}_k} \|Az - r_0\|, \tag{2.6}$$

where $\beta = \|r_0\|$.

In practice, GMRES implementations are based on solving the small least squares problem (2.5). We formulate now some other theoretical ways of interpreting the GMRES approximating problem. If we write now $v = A\,z - r_0$, we can see that the minimization problem (2.6) is equivalent to the following constrained minimization problem

$$\min_{v\in\mathbb{R}^n,\ z\in K_k(r_0,A)} \|v\|, \quad \text{subject to} \quad A\,z = r_0 + v. \tag{2.7}$$

Since $v = Az - r_0 \in \mathcal{K}_{k+1}$, we can write it as $v = V_{k+1}w$, with $w \in \mathbb{R}^{k+1}$. Therefore this problem is equivalent to

$$\min_{w\in\mathbb{R}^{k+1},\ z\in K_k(r_0,A)} \|w\|, \quad \text{subject to} \quad A\,z = r_0 + V_{k+1}w. \tag{2.8}$$

3. The CMRH method. Our description of CMRH in this section parallels that of GMRES in the previous section. We begin by considering the LU factorization of the $n\times k$ Krylov matrix

$$K_k = L_k U_k, \tag{3.1}$$

with $L_k \in \mathbb{R}^{n\times k}$ lower unit triangular, and U_k upper triangular. As we did in (2.3), we can now write

$$K_{k+1} \begin{bmatrix} 0 \\ \\ I_k \end{bmatrix} = L_{k+1}U_{k+1} \begin{bmatrix} 0 \\ \\ I_k \end{bmatrix} = A\,K_k = A\,L_k\,U_k,$$

and similarly since U_k^{-1} is upper triangular, we define the following $(k+1)\times k$ upper Hessenberg matrix

$$H_{k+1,k}^{(h)} = U_{k+1}\begin{bmatrix} 0_{1\times k} \\ \\ I_k \end{bmatrix} U_k^{-1} = \begin{bmatrix} H_k^{(h)} \\ \\ h_{k+1,k}^{(h)}\left(e_k^{(k)}\right)^T \end{bmatrix},$$

with $H_k^{(h)}$ being square upper Hessenberg matrix of order k. We also have the following *Hessenberg relation*

$$\begin{aligned} A\,L_k &= L_{k+1}U_{k+1}\begin{bmatrix} 0 \\ \\ I_k \end{bmatrix} U_k^{-1} \\ &= L_{k+1}H_{k+1,k}^{(h)} = L_k\,H_k^{(h)} + h_{k+1,k}^{(h)}\,\ell_{k+1}\,e_k^T. \end{aligned} \tag{3.2}$$

Thus, the columns of L_k form a different (non-orthogonal) basis of $\mathcal{K}_k$; see Section 4 for some identities with the upper Hessenberg matrix $H_k^{(h)}$.

The Hessenberg process consists of building iteratively the basis $\ell_1,\ldots,\ell_k$ of $\mathcal{K}_k$, so that L_k is unit lower triangular. It begins appropriately by computing $\ell_1 = r_0/\alpha$, with $\alpha = (r_0)_1$, i.e., the first entry of r_0. At each step $j+1$, it proceeds by computing $w = A\ell_j$, and subtracting from it multiples of $\ell_1,\ldots,\ell_j$, to annihilate the first j components of w. The resulting vector is then normalized so that the $j+1$ entry is one. The coefficients of these operations are precisely the entries of $H_{k+1,k}^{(h)}$. In this process, some needed coefficients may be zero (or small), in which case, pivoting is needed. This is not problematic, since a corresponding nonzero is always available after a permutation. It is not hard to see that the Hessenberg process takes many fewer operations (about nk^2 fewer in the full case) and less storage than the Arnoldi process. For more details, see, e.g., [5], [9], or [12, §§6.28–6.30].

We can now define the CMRH method as the counterpart to GMRES, being interpreted as computing the solution of the constrained minimization problem (2.8), but now using the basis of $\mathcal{K}_{k+1}$ provided from the Hessenberg process. Thus, formally, the CMRH approximation x_k^c is such that $x_k^c = x_0 + z_k^c$, $z_k^c \in \mathcal{K}_k$ which solves the following constrained problem

$$\min_{u\in\mathbb{R}^{k+1},\ z\in\mathcal{K}_k(A,r_0)} \|u\|, \quad \text{subject to} \quad A\,z = r_0 + L_{k+1}u. \tag{1.3}$$

Since the columns of L_{k+1} are not orthogonal, we cannot write directly a counterpart to (2.7), but if we denote by $L_{k+1}^\dagger$ the pseudoinverse of L_{k+1}, we can write a formulation equivalent to (1.3) as follows

$$\min_{v\in\mathbb{R}^n,\ z\in\mathcal{K}_k(r_0,A)} \|L_{k+1}^\dagger v\|, \quad \text{subject to} \quad A\,z = r_0 + v;$$

in fact, any left inverse of L_{k+1} will work here, but we use the pseudoinverse for simplicity; though see further comments in Section 4. Thus, we can say equivalently that the CMRH approximation is $x_k = x_0 + z_k^c$, where z_k^c solves

$$\min_{z\in\mathcal{K}_k(r_0,A)} \|L_{k+1}^\dagger(Az - r_0)\|,$$

i.e., it minimizes a quasiresidual. This is similar to the well-known QMR method [2]. Alternatively we can view this minimization, as a minimization of the residual using a seminorm; see [5], [9].

In terms of implementation, we write $z \in \mathcal{K}_k$ as $z = L_k y$, $y \in \mathbb{R}^k$, so that the CMRH approximation is $x^c = x_0 + L_k y_k^c$, where y_k^c solves

$$\min_{y\in\mathbb{R}^k} \|L_{k+1}^\dagger (r_0 - AL_k y)\| = \min_{y\in\mathbb{R}^k} \|L_{k+1}^\dagger (\alpha L_{k+1} e_1^{(k+1)} - L_{k+1} H_{k+1,k}^{(h)} y)\| \tag{3.3}$$

$$= \min_{y\in\mathbb{R}^k} \|\alpha e_1^{(k+1)} - H_{k+1,k}^{(h)} y\|, \tag{3.4}$$

and this is how CMRH is implemented, i.e., solving the small least squares problem (3.4), just as in GMRES, where (2.5) was solved; see [5], [9].

We point out that as is the case with QMR, CMRH is a *non-optimal method* since the basis chosen for $\mathcal{K}_k$ is not orthogonal. Therefore the analysis and bounds in [10] apply in particular to CMRH.

4. Relation between CMRH and GMRES. We begin the section with two new results relating the upper Hessenberg matrices produced by the Arnoldi and Hessenberg methods.

Let us define the $k \times k$ upper triangular matrix $R_k = \tilde{R}_k U_k^{-1}$. We show below that this upper triangular matrix is precisely the change of basis matrix that takes the basis of the Arnoldi basis of the Krylov subspace into the Hessenberg basis. Indeed, from (2.2) and (3.1) it follows directly that $V_k \tilde{R}_k = L_k U_k$, from where we can write

$$L_k = V_k \tilde{R}_k U_k^{-1} = V_k R_k, \tag{4.1}$$

i.e., the QR factorization of L_k. We use these matrices to rewrite the Arnoldi relation (2.4) as

$$AL_k R_k^{-1} = L_{k+1} R_{k+1}^{-1} H_{k+1,k}.$$

Comparing with (3.2) we obtain the following relation between the upper Hessenberg matrices obtained with the Arnoldi process and the Hessenberg process.

PROPOSITION 4.1. *$H_{k+1,k}^{(h)} = R_{k+1}^{-1} H_{k+1,k} R_k$, or equivalently*
$H_{k+1,k} = R_{k+1} H_{k+1,k}^{(h)} R_k^{-1}$.

COROLLARY 4.2. *Let $\rho_k \in \mathbb{R}^k$ be the vector containing the first k components of the $k+1$st column of R_{k+1}. Then, the following relations hold.*

$$R_k H_k^{(h)} R_k^{-1} + \frac{h_{k+1,k}^{(h)}}{r_{k,k}} \rho_k (e_k^{(k)})^T = H_k, \tag{4.2}$$

and

$$r_{k+1,k+1} h_{k+1,k}^{(h)} = h_{k+1,k} r_{k,k}. \tag{4.3}$$

Proof. Observe that both $\tilde{R}_k$ and U_k are principal submatrices of $\tilde{R}_{k+1}$ and U_{k+1}, respectively. In particular, U_k^{-1} is a principal submatrix of U_{k+1}^{-1}. Consequently, since these are all upper triangular matrices, $R_k = \tilde{R}_k U_k^{-1}$ is a principal submatrix of R_{k+1}.

Thus, we have that

$$R_{k+1}H_{k+1,k}^{(h)} = \left[\begin{array}{c|c} R_k & \rho_k \\ \hline 0 & r_{k+1,k+1} \end{array}\right] \left[\begin{array}{c} H_k^{(h)} \\ \hline h_{k+1,k}^{(h)} \end{array}\right]$$

and

$$H_{k+1,k}R_k = \left[\begin{array}{c} H_k \\ \hline h_{k+1,k} \end{array}\right] \left[\begin{array}{c} R_k \\ \\ \end{array}\right]$$

Since it follows from Proposition 4.1 that $R_{k+1}H_{k+1,k}^{(h)} = H_{k+1,k}R_k$ one obtains that $R_kH_k^{(h)} + h_{k+1,k}^{(h)}\rho_k e_k^T = H_kR_k$ and also that (4.3) holds. The results follows. ∎

REMARK 4.3. Note that from (4.3) it follows that

$$\frac{h_{k+1,k}^{(h)}}{r_{k,k}} = \frac{h_{k+1,k}}{r_{k+1,k+1}},$$

and also that if the Arnoldi method terminates ($h_{k+1,k} = 0$), so does the Hessenberg method ($h_{k+1,k}^{(h)} = 0$). The relation (4.2) shows that at termination both methods produce upper Hessenberg matrices with the same eigenvalues.

We also note that we can write the explicit form of pseudoinverse of L_k. Indeed, from (4.1) it follows that $L_k^\dagger = R_k^{-1}V_k^T$.

We discuss now some bounds on the residual norms. We mention in passing that the bounds in [10] are not very tight for CMRH, and therefore cannot be used to fully understand the behavior of the method (see the experiments in Section 5). The following result is in the spirit of [10], and indicates that as long as the condition number of R_k does not grow too fast, the CMRH residual norm will be close to the GMRES residual norm.

THEOREM 4.4. *Let r_k^G and r_k^c be the GMRES and CMRH residuals at the kth iteration beginning with the same initial residual r_0, respectively. Then*

$$\|r_k^G\| \le \|r_k^c\| \le \kappa(R_{k+1})\|r_k^G\|, \tag{4.4}$$

where $\kappa(R_{k+1}) = \|R_{k+1}\|\|R_{k+1}^{-1}\|$ is the condition number of R_{k+1}.

Proof. The first inequality in (4.4) is standard and follows from (2.5). Since both r_k^G and r_k^c are in $\mathcal{K}_{k+1}$, we can write $r_k^G = L_{k+1}u_k^G = V_{k+1}w_k^G$, $r_k^c = L_{k+1}u_k^c = V_{k+1}w_k^c$ with $u_k^G,\ u_k^c,\ w_k^G,\ w_k^c \in \mathbb{R}^{k+1}$. Thus $R_{k+1}u_k^G = w_k^G$. Because of the characterization (1.3), we have that

$$\|u_k^c\| \le \|u_k^G\| \le \|R_{k+1}^{-1}\|\|w_k^G\| = \|R_{k+1}^{-1}\|\|r_k^G\|.$$

On the other hand, from (4.1),

$$\|r_k^c\| = \|L_{k+1}u_k^c\| \le \|L_{k+1}\|\|u_k^c\|.$$

Since $\|L_{k+1}\| = \|R_{k+1}\|$, the proof is complete. ∎

We note that from (4.1) it follows that $\kappa(R_{k+1}) = \kappa(L_{k+1})$, and thus, we recover Theorem 4 of [9], which in turn is similar to a result on QMR [2]. What is of interest here is that unlike the cases cited, we have an upper triangular matrix, which is the change of basis between the two basis of the Krylov subspace.

We end this section with further relations between CMRH and GMRES iterates and residuals. We also provide the exact formulae for w_k^G and w_k^c defined in the proof of the preceding theorem.

PROPOSITION 4.5. *The following relations hold.*

(i) $R_k = V_k^H L_k$, $R_k^{-1} = L_k^\dagger V_k$.

(ii) $r_k^G = \|r_0\| V_{k+1}(I - H_{k+1,k} H_{k+1,k}^\dagger) e_1$

(iii) $r_k^c = \|r_0\| V_{k+1}(I - H_{k+1,k}\big(R_{k+1}^{-1} H_{k+1,k}\big)^\dagger R_{k+1}^{-1}) e_1$.

(iv) $\dfrac{\|x_k^G - x_k^c\|}{\|r_0\|} = \|(H_{k+1,k}^\dagger - \big(R_{k+1}^{-1} H_{k+1,k}\big)^\dagger R_{k+1}^{-1}) e_1\|$.

Proof. The relation (i) follows directly from (4.1) and the fact that $L_k^\dagger$ is the pseudoinverse of L_k. Relations (ii) and (iii) follow from the expression of the minimizing element of (2.5) and (3.3), respectively. We now use the Arnoldi relation $AV_k = V_{k+1} H_{k+1,k}$ in (ii) and (iii), and obtain

$$r_k^G - r_k^c = \|r_0\| A V_k \left(H_{k+1,k}^\dagger - \big(R_{k+1}^{-1} H_{k+1,k}\big)^\dagger R_{k+1}^{-1} \right) e_1,$$

from where (iv) follows directly. □

5. Numerical Experiments. In this section we provide experimental results using the CMRH method, and comparing its performance with the GMRES. We illustrate the theory developed in this paper with two classes of matrices. The first involve well conditioned matrices (examples 5.1, 5.2 and 5.3); while the matrices in the next two examples are badly conditioned. In in the first four experiments we have $n = 1000$, $x_0 = 0$. For the first three examples, the right hand side is chosen such that the exact solution x^* is a known random vector. The first five numerical experiments were carried out using Matlab. Fortran was used for part of the experiments of the last example.

We begin with examples from the Matlab gallery test matrices [6].

	LU	GMRES	CMRH
Iter.		13	13
$\|residual\|$	$4.42\ 10^{-14}$	$4.07\ 10^{-14}$	$5.11\ 10^{-14}$
$\|error\|$	$3.03\ 10^{-14}$	$3.98\ 10^{-14}$	$3.35\ 10^{-14}$

TABLE 5.1
Results obtained for the matrix Ris.

EXAMPLE 5.1. In this example we consider the symmetric Hankel matrix (A=gallery('ris',n)) with elements

$$a(i,j) = \frac{0.5}{n - i - j + 1.5}\ .$$

The converge of both methods is very fast: both GMRES and CMRH converge in 13 iterations, as can be seen in Table 5.1. In this table, and in those of the other experiments, we also report the norms of the residual and of the error of the computed

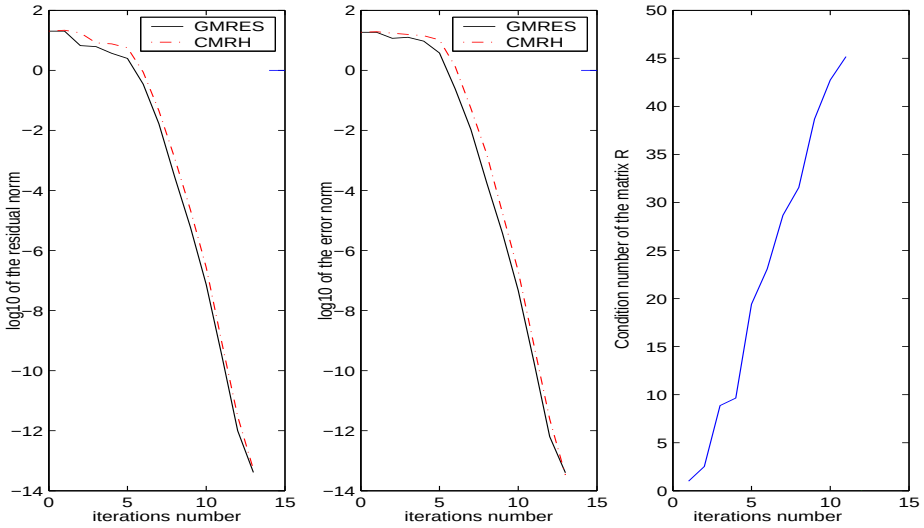

FIG. 5.1. *Example 5.1: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

approximation. For comparison, we also report these two quantities when the system is solved with a direct method (we use the default direct method in Matlab) and label it LU in the tables. For this experiments, for all three methods reported, the norms of the residual and of the error are of the same order of magnitude. In Figure 5.1, we present the usual convergence curves for CMRH and GMRES (residual norms), as well as the norms of the corresponding errors. As it can be appreciated, the curves for CMRH and GMRES are very close to each other. We also present in the same figure the values of $\kappa(R_k)$ for each iteration k. One can see that these values are very small.

EXAMPLE 5.2. For this experiment we use the Riemann matrix (A=gallery('riemann',n)) with elements

$$a(i,j) = i \text{ if } i+1 \text{ divides } j+1, \text{ and } a(i,j) = -1 \text{ otherwise.}$$

As it can be seen in Table 5.2 and Figure 5.2, the curves for GMRES and CMRH are very close to each other, and $\kappa(R_k)$ grows only to be of order 100.

	LU	GMRES	CMRH
Iter.		241	242
$\|residual\|$	$4.62\ 10^{-11}$	$1.74\ 10^{-11}$	$5.01\ 10^{-11}$
$\|error\|$	$5.77\ 10^{-13}$	$1.82\ 10^{-12}$	$2.70\ 10^{-12}$

TABLE 5.2
Results obtained for the matrix Riemann.

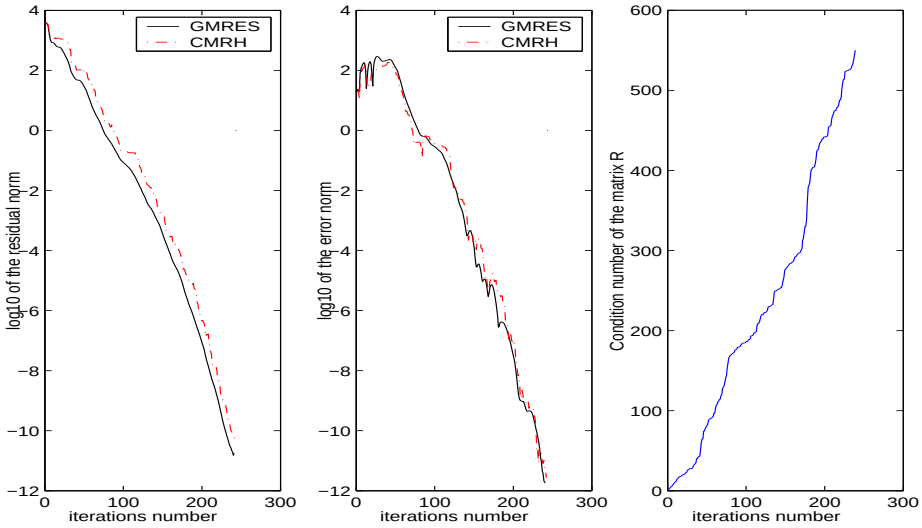

FIG. 5.2. *Example 5.2: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

EXAMPLE 5.3. In this example, we consider a random matrix $A = rand(1000)$ of dimension 1000. Such a matrix has condition number O(1000) on average and smoothly distributed singular values. As it can be observed in Figure 5.3 both GMRES and CMRH stagnate, i.e., they make negligible progress until the iteration number reaches $n = 1000$. This is the usual case for this kind of matrices. In Table 5.3 we also observe that the achieved accuracy is of the same order of magnitude for all methods. Note that the condition number of R_k reaches a value of only the order of 10^3.

	LU	GMRES	CMRH
Iter.		1000	1000
$\|residual\|$	$1.13\ 10^{-11}$	$1.52\ 10^{-11}$	$3.05\ 10^{-11}$
$\|error\|$	$1.09\ 10^{-10}$	$1.03\ 10^{-10}$	$1.89\ 10^{-10}$

TABLE 5.3
Results obtained for a random matrix of size $n = 1000$.

For the next two examples we use the Matlab Regularization Tools from [3]. As is well known, these matrices are not well conditioned.

EXAMPLE 5.4. Consider the solution of the Fredholm integral equation of the first kind

$$\int_{-6}^{6} \kappa(s,t)x(t)dt = y(s), \qquad -6 \le s \le 6. \tag{5.1}$$

Its solution, kernel, and right-hand side are given by

$$x(t) = \begin{cases} 1 + \cos(\frac{\pi}{3}t), & \text{if } |t| < 3, \\ 0, & \text{otherwise,} \end{cases}$$
$$\kappa(s,t) = x(s-t),$$
$$y(s) = (6 - |s|)(1 + \frac{1}{2}\cos(\frac{\pi}{3}s)) + \frac{9}{2\pi}\sin(\frac{\pi}{3}|s|).$$

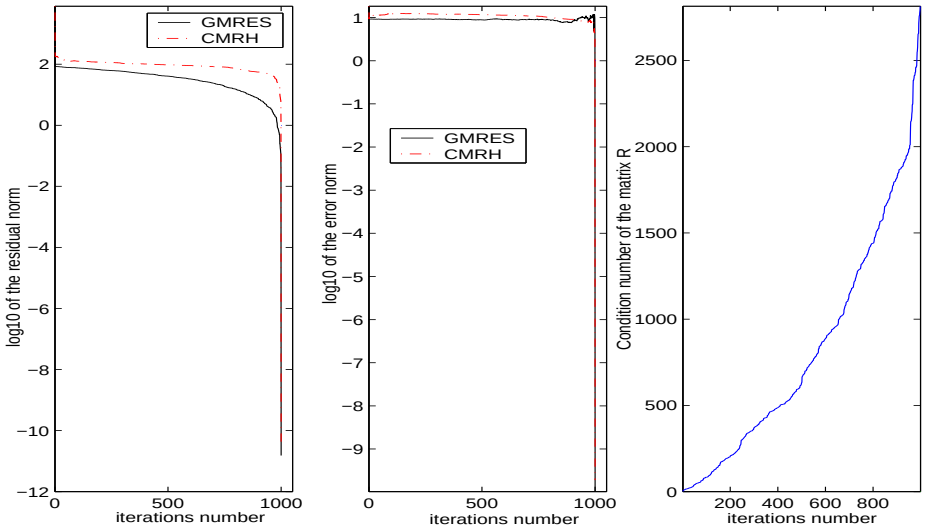

FIG. 5.3. *Example 5.3: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

We use the code phillips from [3] to discretize (5.1) by a Galerkin method with orthonormal box functions as test and trial functions to obtain the symmetric matrix A of order $n = 1000$.

	LU	GMRES	CMRH
Iter.		157	133
$\|residual\|$	$1.47\ 10^{-14}$	$4.59\ 10^{-12}$	$3.8\ 10^{-11}$
$\|error\|$	$4.38\ 10^{-5}$	$2.29\ 10^{-5}$	$2.34\ 10^{-5}$

TABLE 5.4
Results obtained for the matrix Phillips.

In Table 5.4, we can observe the effect of the high condition number of A: the norm of the error of the obtained approximations is several orders of magnitude larger than that of the corresponding residuals. At the same time, we note in Figure 5.4 that $\kappa(R_k)$ is only of the order of 100, and this implies that the convergence curves of CMRH and GMRES are close to each other. This fact is also observed in Figure 5.4.

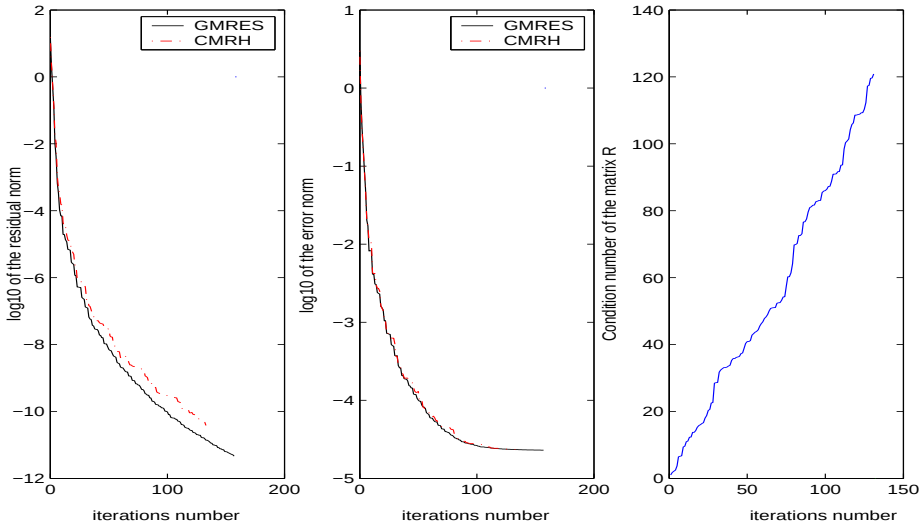

FIG. 5.4. *Example 5.4: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

EXAMPLE 5.5. We consider the integral equation

$$\int_0^{\pi/2} \kappa(s,t)x(t)dt = g(s), \qquad 0 \le s \le \pi, \tag{5.2}$$

where

$$\kappa(s,t) = exp(s\,cos(t)), \quad \text{and} \quad g(s) = 2sin(s)/s.$$

The solution is given by $f(t) = sin(t)$. We used the Matlab code baart from [3] to discretize (5.2) by a Galerkin method with 1000 orthonormal box functions. This yields the matrix A of order $n = 1000$. We use the right hand side b so that the solution is the known solution.

	LU	GMRES	CMRH
Iter.		7	7
$\|residual\|$	$8.84\ 10^{-15}$	$1.90\ 10^{-13}$	$2.58\ 10^{-13}$
$\|error\|$	$9.45\ 10^{+2}$	$2.87\ 10^{-5}$	$2.89\ 10^{-5}$

TABLE 5.5
Results obtained for the matrix Baart.

As in Example 5.4, we can see in Table 5.5 that the norms of the errors are much larger than those of the residual. Note however that for this example, the direct method gives an error which is seven orders of magnitude higher than the iterative methods. Note also, that the iterative methods converge very fast and that their convergence curves stay very close to each other, with $\kappa(R_k)$ only reaching a value of the order of 10; see Figure 5.5.

EXAMPLE 5.6. For our last numerical experiment, we consider the matrix A, which is defined by:

$$A_{i,j} = \begin{cases} -log\,|z_i - z_j|\,, & \text{for } \ i \neq j, \\ -log\,|r_i|\,, \end{cases}$$

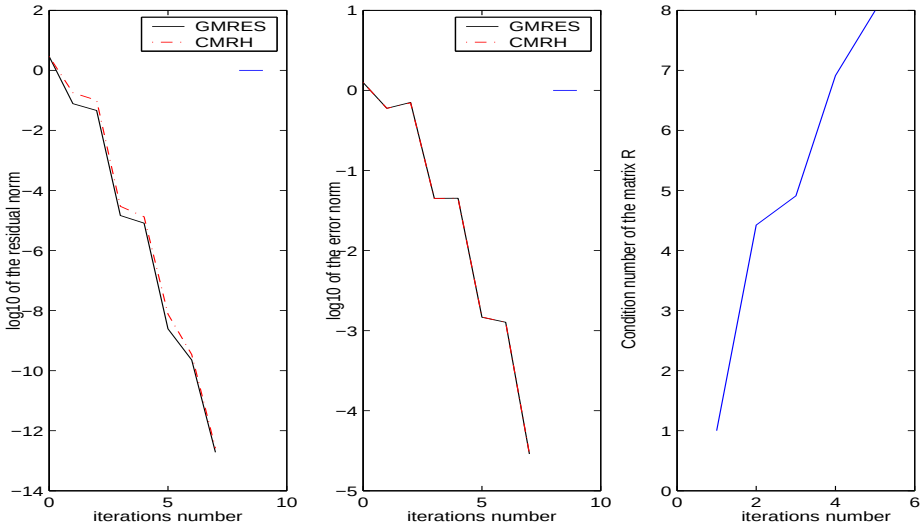

FIG. 5.5. *Example 5.5: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

where z_i are n randomly distributed points in a unit square centered at the origin in the complex plane and where each r_i is a number in $(0, d_i]$, d_i being the distance between the point z_i and its nearest neighbor. We observe that computing an off-diagonal entry A_{ij} corresponds, up to a factor of $-\frac{1}{2\pi}$, to evaluating the free space Green's function for the Laplacian in two dimensions with argument $z_i - z_j$. For more details, see [4]. We choose the right hand side, so that the exact solution x^* is equal to $x_i^* = 1$, for $i = 1, \ldots, n$. We run three experiments for this example. In the first one we used Matlab.

In this first experiment, we set $n = 1000$, and compare the behavior of CMRH and GMRES. As shown in Figure 5.6 and Table 5.6, the curves of both GMRES and CMRH are very close to each other, even $\kappa(R_k)$ grows. GMRES and CMRH converge in 690 and 694 respectively. We can also remark that despite the fact that the compueted Arnoldi vectors loose orthogonality ($\|I - V_k^T V_k\|$ increases) as shown in Figure 5.7, both methods produce an accurate solution.

	LU	GMRES	CMRH
Iter.		690	694
$\|residual\|$	$3.21\ 10^{-12}$	$1.22\ 10^{-11}$	$8.38\ 10^{-12}$
$\|error\|$	$7.14\ 10^{-11}$	$7.39\ 10^{-11}$	$3.73\ 10^{-11}$

TABLE 5.6
Results obtained for the matrix Helsing of size $n = 1000$.

In the second test, we set $n = 20000$, and compare the behavior of CMRH and GMRES, as shown in Figure 5.8 and Table 5.7. As it can be observed, the curves of both GMRES and CMRH are very close to each other for most of the computation. At a certain point (after 8000 iterations), the computed CMRH residual stagnates around 10^{-8} with an error stagnating at about the same magnitude. On the other hand, the computed GMRES residual continues to decrease, but the corresponding error grows by at least an order of magnitude. A direct method works well for this

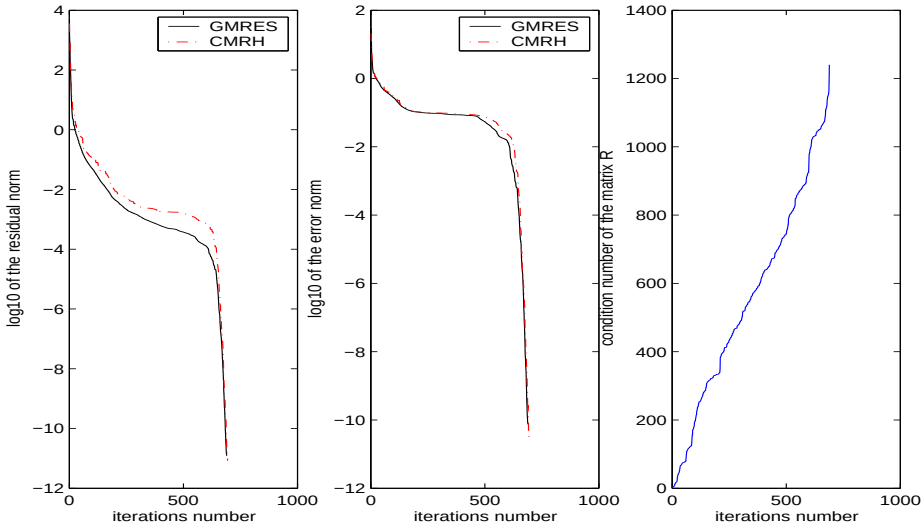

FIG. 5.6. *Example 5.6, $n = 1000$: norms of residuals (left) and errors (center); and $\kappa(R_k)$ (right).*

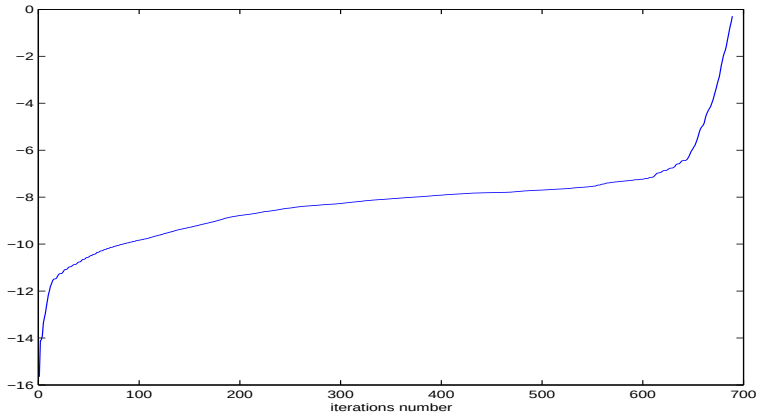

FIG. 5.7. *Example 5.6 $n = 1000$, $\log_{10} \|I - V_k^T V_k\|$.*

problem.

	LU	GMRES	CMRH	GMRES	CMRH
Iter.		8715	8715	9000	9000
$\|residual\|$	$2.67\ 10^{-9}$	$5.16\ 10^{-9}$	$5.43\ 10^{-9}$	$6.37\ 10^{-10}$	$3.12\ 10^{-9}$
$\|error\|$	$1.13\ 10^{-9}$	$1.52\ 10^{-8}$	$2.34\ 10^{-8}$	$2.46\ 10^{-7}$	$2.34\ 10^{-8}$

TABLE 5.7
Results obtained for the matrix Helsing of size $n = 20000$.

In the last run, we consider the same problem, but of a larger size, namely $n = 88000$. For this problem, the full GMRES can not be used, since the memory needed would exceed our available 64GB. The CMRH method works in this case, since the basis of the Krylov subspace overwrites the columns of the matrix, and

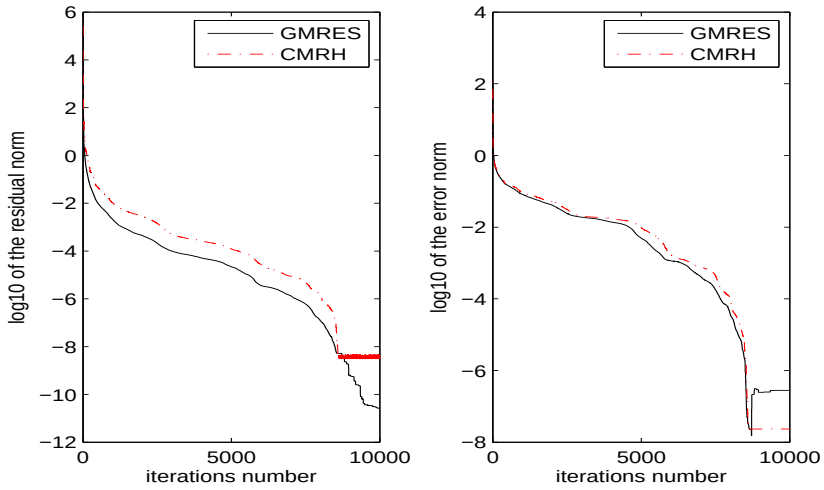

FIG. 5.8. *Example 5.6, $n = 20000$: norms of residuals (left) and errors (right).*

very little additional storage is needed. CMRH converges in 30801 iterations, and the norm of the error at this iteration was 1.14×10^{-5}. The error in the direct method was 5.77×10^{-6}. The curve of the quasiresidual obtained from (3.4) is given in in Figure 5.9.

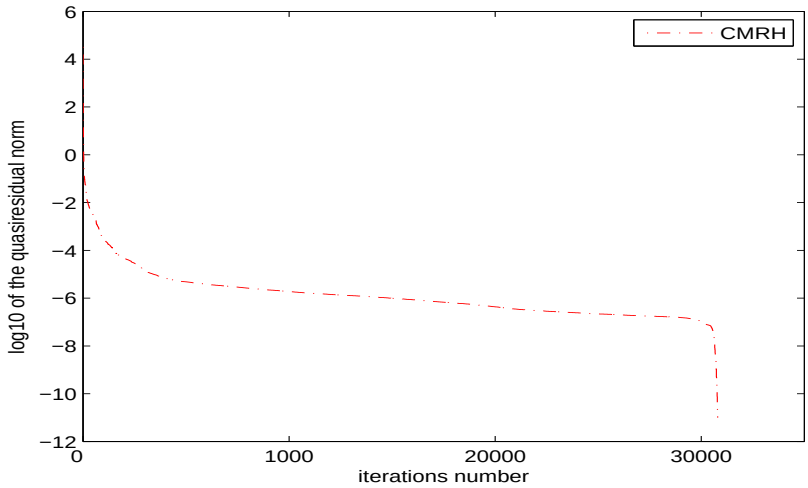

FIG. 5.9. *Results obtained for the matrix Helsing of size $n = 88000$: norms of the quasiresiduals.*

6. Conclusions. We have presented some new analysis of the Hessenberg and CMRH methods, and formulae relating its iterates and residuals with those of Arnoldi and GMRES. These results indicate why CMRH works so well in comparison to GMRES (but with lower cost and potentially with lower storage requirement, especially for matrices which are not so sparse). Numerical experiments illustrate these obser-

vations.

REFERENCES

[1] Roland W. Freund, Gene H. Golub, and Nöel M. Nachtigal. Iterative solution of linear systems. *Acta Numerica*, pages 57–100, 1992.

[2] Roland W. Freund and Nöel M. Nachtigal. QMR: A quasi-minimal residual method for non-Hermitian linear systems. *Numerische Mathematik*, 60:315–339, 1991.

[3] Per Christian Hansen, Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems. *Numerical Algorithms*, 6:1–35, 1994. Software is available in Netlib at `http://www.netlib.org`.

[4] Johan Helsing, Approximate inverse preconditioners for some large dense random electrostatic interaction matrices, *BIT Numerical Mathematics*, 46:307–323, 2006.

[5] Mohammed Heyouni and Hassane Sadok. A new implementation of the CMRH method for solving dense linear systems. *Journal of Computational and Applied Mathematics*, 213:387–399, 2008.

[6] Nicholas J. Higham. The Matrix Computation Toolbox. Available online at `http://www.ma.man.ac.uk/ higham/mctoolbox`.

[7] Yousef Saad and Martin H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869, 1986.

[8] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. The PWS Publishing Company, Boston, 1996. Second edition, SIAM, Philadelphia, 2003.

[9] Hassane Sadok. CMRH: A new method for solving nonsymmetric linear systems based on the Hessenberg reduction algorithm. *Numerical Algorithms*, 20:303–321, 1999.

[10] Valeria Simoncini and Daniel B. Szyld. The effect of non-optimal bases on the convergence of Krylov subspace methods. *Numerische Mathematik*, 100:711–733, 2005.

[11] Valeria Simoncini and Daniel B. Szyld. Recent computational developments in Krylov subspace methods for linear systems. *Numerical Linear Algebra with Applications*, 14:1–59, 2007.

[12] James H. Wilkinson. *The Algebraic Eigenvalue Problem*. Clarendon Press, Oxford, 1965.