

An Efficient Algorithm for Inverting a Block-Symmetric Matrix*

By S. Charmonman

1. Introduction. A matrix of block-symmetric form is typified by that in the theory of directional coupler [3] and in overlapping polymer chains [4],

$$(1) \quad R = \begin{bmatrix} A & B \\ B & A \end{bmatrix},$$

where A and B are real square matrices of order n . In general, when A and B are nonsymmetric, the block-symmetric matrix R is also nonsymmetric. One of the best methods of inverting a fully populated and nonsymmetric matrix in general is Gaussian elimination with partial pivoting. If the block-symmetry of R is disregarded the number of arithmetic operations required in inverting the $2n \times 2n$ matrix R by Gaussian elimination would be of order $8n^3$ multiplications [1, p. 178]. If compact elimination is used computer storage required would be of order $(2n)^2$.

In order to use the block-symmetry of R in its inversion Schur's identity [2] can be employed. Let the inverse of R be

$$(2) \quad R^{-1} = \begin{bmatrix} E & F \\ G & H \end{bmatrix}.$$

Schur's identity gives

Algorithm 1A. To obtain R^{-1} , compute in succession:

- (1) $\Delta = A - B(A^{-1}B)$,
- (2) $H = \Delta^{-1}$,
- (3) $G = -\Delta^{-1}BA^{-1}$,
- (4) $F = -(A^{-1}B)\Delta^{-1}$, and
- (5) $E = A^{-1} + (A^{-1}B)G$.

If $A^{-1}B$ computed in Step (1) is saved to be used again in Steps (4) and (5), the total operations required to obtain R^{-1} are two inversions, six multiplications and two additions of matrices of order n . The corresponding number of scalar multiplications is thus of order $8n^3$ which is about the same as in using Gaussian elimination on R without regarding the block-symmetry.

It can be easily verified that $E = H$ and $F = G$. Therefore an alternative to Algorithm 1A could be

Algorithm 1B. To obtain R^{-1} , compute in succession:

- (1) $E = H = (A - BA^{-1}B)^{-1}$ and
- (2) $F = G = (B - AB^{-1}A)^{-1}$.

Although Algorithm 1B appears to be more compact than Algorithm 1A the number of multiplication required is about the same, that is, of order $8n^3$ for four inversions, four multiplications and two additions of matrices of order n .

Received February 17, 1967.

* This work was supported by the National Research Council of Canada under Grant No. A-3135.

2. An Efficient Algorithm. A more efficient algorithm can be devised by use of THEOREM 1. *The inverse of*

$$R = \begin{pmatrix} A & B \\ B & A \end{pmatrix} \text{ is } \begin{pmatrix} E & F \\ F & E \end{pmatrix}$$

where $E = .5(P^{-1} + Q^{-1})$, $F = .5(P^{-1} - Q^{-1})$, $P = A + B$ and $Q = A - 2B$.

A proof can be easily constructed by equating terms of the product RR^{-1} to those of I. The algorithm obtained is

Algorithm 2. To obtain R^{-1} , compute in succession:

- (1) $P = A + B$, $Q = P - 2B$,
- (2) P^{-1} , Q^{-1} ,
- (3) $E = .5(P^{-1} + Q^{-1})$ and $F = E - Q^{-1}$.

Algorithm 2 is more efficient than Algorithms 1A and 1B in the sense that there are savings in both computer storage and the number of arithmetic operations required. Computer storage required is of order $2n^2$, say two $n \times n$ arrays whose contents are shown in Table 1.

TABLE 1
Contents of Arrays in Inversion Algorithm 2

Step \ Array	0	1	2	3
1	A	$P = A + B$	P^{-1}	$E = .5(P^{-1} + Q^{-1})$
2	B	$Q = P - 2B$	Q^{-1}	$F = E - Q^{-1}$

In each step, computation is carried out first in the first array and then in the second array. The total number of multiplications required is only of order $2n^3$, that is, in carrying out two inversions, two additions of matrices of order n , and two multiplications of a scalar to a matrix of order n . Therefore the saving in arithmetic operations in using Algorithm 2 instead of Algorithms 1A or 1B is about 75%.

3. Solution of a System of Equations. If only one system of equations is to be solved, it is better to find the solution without actually computing the inverse than by multiplication of the inverse to the right-hand side. In this case the number of scalar multiplications required in applying Gaussian elimination with partial pivoting directly to $Rx = b$ is of order $8n^3/3$ [1, p. 176]. A more efficient approach similar to that in Algorithm 2 is based on

THEOREM 2. *The solution of*

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

is $x_1 = .5(y_1 + y_2)$ and $x_2 = .5(y_1 - y_2)$ where y_1 and y_2 are solutions of $(A + B)y_1 = b_1 + b_2$ and $(A - B)y_2 = b_1 - b_2$ respectively.

A proof is again obvious. The resulting algorithm is shown in Table 2. The computer storage is two $n(n + 1)$ arrays. Broken vertical bars are used to indicate partitioned matrices.

TABLE 2
Solution of One System Equations

<i>Step</i> <i>Array</i>	0	1	2	3
1	$A \mid b_1$	$P = A + B \mid c_1 = b_1 + b_2$	Solve $Py_1 = c_1$	$x_1 = .5(y_1 + y_2)$
2	$B \mid b_2$	$Q = P - 2B \mid c_2 = c_1 - 2b_2$	Solve $Qy_2 = c_2$	$x_2 = x_1 - y_2$

Department of Computing Science
 The University of Alberta
 Edmonton, Alberta, Canada

1. L. FOX, *An Introduction to Numerical Linear Algebra*, Monographs on Numerical Analysis, Clarendon Press, Oxford, 1964. MR 29 #1733.
2. MARVIN MARCUS, "Basic theorems in matrix theory," *Nat. Bur. Standards Appl. Math. Ser.*, v. 57, 1960, p. 17. MR 22 #709.
3. C. G. MONTGOMERY, ET AL, *Principles of Microwave Circuits*, Boston Technical Publishers, Inc., Lexington, Mass., 1964, pp. 437-452.
4. E. W. MONTROLL, "Markoff chains and excluded volume effect in polymer chains," *J. Chem. Phys.*, v. 18, 1950, pp. 734-743. MR 12, 114.

A Note on the Effect of Conditionally Stable Correctors

By Fred T. Krogh

We say a corrector of the form

$$(1) \quad y_{n+1} = \sum_{i=0}^{k-1} A_i y_{n-i} + h \sum_{i=-1}^{k-1} a_i y'_{n-i}$$

is *conditionally stable* if the polynomial

$$(2) \quad \rho(z) = z^k - \sum_{i=0}^{k-1} A_i z^{k-1-i}$$

has all of its roots in the unit disk, roots of unit magnitude are simple, and there is at least one root of unit magnitude besides the root $z = 1$ (which must be a root since it is assumed that Eq. (1) is satisfied if y is a constant). In [1], Stetter obtains the remarkable result that a predictor-corrector algorithm using Simpson's rule (a conditionally stable corrector) will be relatively stable* for sufficiently small h provided the predictor is chosen judiciously and the corrector is only applied once. However, his result applies only to the integration of a single differential equation. It is the purpose of this note to point out that no result of this type can

Received October 19, 1966. Revised April 24, 1967.

* From the bewildering array of words in the current literature which describe stability, those used here seem to the author to be most descriptive. Different words are used in [1]. Several definitions of "relatively stable" are given below. In practice they are essentially equivalent.