NASA TN D-1112

NASA TN D-1112

6P

N63-14249
Code 1

TECHNICAL NOTE

D-1112

A RECURSION RELATION ASSOCIATED WITH A CERTAIN SPECIAL TYPE DETERMINANT

By E. Baylis Shanks

George C. Marshall Space Flight Center
Huntsville, Alabama

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

WASHINGTON April 1963

copy 1

Code -1

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

TECHNICAL NOTE D-1112

A RECURSION RELATION ASSOCIATED WITH A CERTAIN SPECIAL TYPE DETERMINANT

By E. Baylis Shanks

SUMMARY

This paper is concerned with the development of formulas of a recursive nature from a certain special type determinant. These formulas lead directly to computational routines that are effective in the evaluation of such determinants. Such computational routines are also applicable to a wide variety of recursion formulas, obtained by specialization of the parameters involved.

From the expansion of this type determinant, a recursion formula is obtained and later simplified to the form

$$R_o = 1, \quad a_{n+1,n} = -1, \quad b_{j+1,j} = -a_{j+1,j}$$

$$D_n = b_{21}\ b_{32} \ldots b_{n,n-1} R_n$$

$$R_i = b_{i+1,i}^{-1} \sum_{j=1}^{i} a_{i-j+1,i}\ R_{i-j}\ .$$

This set of formulas is very efficient for computational purposes as discussed in the paper.

I. INTRODUCTION

This paper grew out of an investigation of an eigenvalue problem concerned with the Schrodinger wave equation using the Yukawa potential. In that investigation, a recursion formula for the coefficients of the power series solution of the differential equation was used as a recursion formula and to deduce a determinant that determined the coefficients. It was discovered that routines for the effective evaluation of such types of determinants could be deduced as a generalization of that work and that these routines were more effective in precision and in time than the more general routines now available. These routines may be used either for evaluations of these special type determinants or of quite general recursion relations.

II. MATHEMATICAL TREATMENT

The type determinant treated in this paper is the following

$$D_n = \begin{vmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ O & a_{32} & a_{33} & \cdots & a_{3n} \\ & & \cdot & \cdot & \cdot \\ O & O & O & & a_{nn} \end{vmatrix}$$

in which $a_{ij} = O$ for $i > j+1$. By a straightforward expansion by minors, the following recursion formula is obtained.

$$D_i = a_{ii}D_{i-1} + \sum_{j=2}^{i} (-1)^{j+1} a_{i,i-1}a_{i-1,i-2} \cdot \cdot \cdot a_{i-j+2,i-j+1}a_{i-j+1,i}D_{i-j}, \qquad (1)$$

where $D_0 = 1$. This formula will generate D_n in n steps by taking in turn the values 1, 2, ..., n for i in equation 1.

Formula 1 can be greatly simplified by setting $(-1)^{i-j+1}D_j = a_{21} \ldots a_{j+1,j}R_j$ for j=1, 2, ..., n. This results in the formula

$$-a_{i+1,i}R_i = \sum_{j=1}^{i} a_{i-j+1,i}R_{i-j}, \qquad (2)$$

where $R_0 = 1$ and for the last step in the process $a_{n+1,n} = -1$. Then it easily follows that

$$D_n = (-1)^n a_{21} a_{32} \ldots a_{n,n-1}R_n \ . \qquad (3)$$

By changing the sign of $a_{j+1,j}$ with the substitution $b_{j+1,j} = -a_{j+1,j}$, we obtain the following set of formulas that are complete

$$R_0 = 1, \; a_{n+1,n} = -1, \; b_{j+1,j} = -a_{j+1,j}$$

$$D_n = b_{21} b_{32} \ldots b_{n,n-1} R_n \tag{4}$$

$$R_i = b_{i+1,i}^{-1} \sum_{j=1}^{i} a_{i-j+1,i} R_{i-j} .$$

It has been assumed in obtaining equations 2 and 4 that $a_{i+1,i} \neq 0$ for each i as is evidenced by the last equation in 4.

If $b_{i+1,i} = 0$, then set

$$D_i = b_{21} b_{32} \ldots b_{i,i-1} \sum_{j=1}^{i} a_{i-j+1,i} R_{i-j}$$

as is done when $i = n$. Proceed to calculate the remaining principal minor D'_{i+1} as is done for D_n. Then $D_n = D_i D'_{i+1}$. This process may be repeated as many times as zeroes occur immediately below the main diagonal.

If zeroes occur (or are suspected to occur), another procedure would be to base the calculation on equation 1, which is correct whether zeroes occur or not. In general, equation 1 leads to more computational time and effort than equations 4.

It is evident from equation 1 and equations 4 that these formulas cover a wide variety of recursion relations and also are applicable in the solution of "almost diagonal" determinants. It is also true that these cases occur quite frequently in practice in physics, engineering, and other applied mathematics fields. These recursion relations might be termed "two dimensional." For example, if we set $a_{ij} = a_{i+1,j+1} = a_i$ for $i > j$ and $b_{i+1,i} = b_i$, we obtain the "one dimensional" recursion formula

$$b_i R_i = C_{i-1} R_{i-1} + \sum_{j=1}^{i-1} a_{i-j-1} R_{i-j-1}, \qquad a_{jj} = C_j , \tag{5}$$

which was associated with the Schrodinger - Yukawa differential equation mentioned above.

III. NUMERICAL CRITIQUE

In approximating by evaluation the real roots of polynomials P_i, given by the determinant associated with relation 5 in the case just mentioned, the IBM 7090 required 19 minutes for 645 evaluations of P_{40} by the available routine. Since this technique was practically prohibitive, other procedures were sought. A procedure was devised based on relation 5 (actually programmed with the cooperation of Mr. Audie Anderson of the General Electric Company staff). This procedure enabled the 7090 to make 645 evaluations (the same number) in 4.2 minutes, not for just the case $i = 40$, but for all cases $i = 1, 2, \ldots, 75$ (645 evaluations for each such case). This is convincing evidence of the importance of such procedures.

It is of interest to compare the procedure outlined by equations 4 with the well-known process of replacing the $a_{i+1,i}$ by zeroes or diagonalizing the determinant. To make clear the comparison, it is informative to write out a few steps for each method as is done below.

Process Based on Equation 4

$$R_1 = b_{21}^{-1} \quad a_{11}$$

$$R_2 = b_{32}^{-1} \quad \left(a_{22} \frac{a_{11}}{b_{21}} + a_{12}\right)$$

$$R_3 = b_{43}^{-1} \quad \left(a_{33} \frac{a_{22} \frac{a_{11}}{b_{21}} + a_{12}}{b_{32}} + a_{23} \frac{a_{11}}{b_{21}} + a_{13}\right)$$

$$\vdots$$

Diagonal Process

$$d_1 = a_{11}$$

$$d_2 = a_{22} + \frac{b_{21}}{a_{11}} \ a_{12}$$

$$d_3 = a_{33} + \left(\frac{b_{32}}{a_{22} + \frac{b_{21}}{a_{11}}} \right) \left(a_{23} + \frac{b_{21}}{a_{11}} \ a_{13}\right)$$

$$\vdots$$

A close analysis of the two reveals several interesting facts. The two procedures are the reciprocals of one another in a certain evident sense. The procedure in the case of formula 4 calls for i multiplications and i-1 additions of known quantities in order to obtain R_i. Here we count division as a multiplication by the reciprocal number as will be done in both cases. The procedure in the case of the diagonal process calls for $n-i+2$ multiplications and $n-i+1$ additions of known quantities in order to obtain d_i. At the end of the procedure, in each case, there are $n-1$ multiplications of known quantities in order to obtain D_n. In the case of formula 4, obtaining R_n calls for $n-1$ multiplications instead of the expected n because $b_{n+1,n} = 1$. On the other hand, the diagonal process only proceeds from $i=2$ to $i=n$. Taking all of these facts into account it is seen that precisely the same number of operations are involved in the two procedures, namely $\frac{1}{2}(n-1)(n+4)$ multiplications and $\frac{1}{2}n(n-1)$ additions. On this basis, we have "dead heat."

However, there is a difference between the two that might work against the diagonal process with respect to round-off error propagation. In the diagonal procedure, the divisors are rounded-off numbers; whereas, in the procedure of formula 4, the dividends are rounded-off numbers while the divisors are given numbers. Such a difference could be decisive in favor of the procedure of formula 4. Tests are planned to determine this difference on the IBM 7090 with determinants that are not too difficult to evaluate precisely.

IV. CONCLUDING REMARKS

From the simplicity of formulas 1 and 4, it is evident that programs can be written that are relatively simple because of the recursive properties of those formulas. These programs are being written and will be made a part of the library of the MSFC Computation Division.

Another point of interest and importance is illustrated by the difference in formulas 1 and 4. The two are mathematically equivalent as demonstrated above, but the difference in the number of multiplications is significant and, in general, formula 4 would have a decided advantage in machine time. This illustrates the great importance of theoretical analyses of the machine routines that are to be used in practice.