

NAG Library Routine Document

C02ANF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

C02ANF determines the roots of a quartic equation with complex coefficients.

2 Specification

```
SUBROUTINE C02ANF(E, A, B, C, D, ZEROR, ZEROI, ERREST, IFAIL)
INTEGER                                IFAIL
double precision                    ZEROR(4), ZEROI(4), ERREST(4)
complex*16                          E, A, B, C, D
```

3 Description

C02ANF attempts to find the roots of the quartic equation

$$ez^4 + az^3 + bz^2 + cz + d = 0,$$

where e , a , b , c and d are complex coefficients with $e \neq 0$. The roots are located by finding the eigenvalues of the associated 4 by 4 (upper Hessenberg) companion matrix H given by

$$H = \begin{pmatrix} 0 & 0 & 0 & -d/e \\ 1 & 0 & 0 & -c/e \\ 0 & 1 & 0 & -b/e \\ 0 & 0 & 1 & -a/e \end{pmatrix}.$$

The eigenvalues are obtained by a call to F08PSF (ZHSEQR). Further details can be found in Section 8.

To obtain the roots of a cubic equation, C02AMF can be used.

4 References

Golub G H and Van Loan C F (1996) *Matrix Computations* (3rd Edition) Johns Hopkins University Press, Baltimore

5 Parameters

- | | | |
|----|--|--------------|
| 1: | E – complex*16
<i>On entry:</i> e , the coefficient of z^4 .
<i>Constraint:</i> $E \neq (0.0, 0.0)$. | <i>Input</i> |
| 2: | A – complex*16
<i>On entry:</i> a , the coefficient of z^3 . | <i>Input</i> |
| 3: | B – complex*16
<i>On entry:</i> b , the coefficient of z^2 . | <i>Input</i> |
| 4: | C – complex*16
<i>On entry:</i> c , the coefficient of z . | <i>Input</i> |

- 5: D – *complex*16* *Input*
On entry: d , the constant coefficient.
- 6: ZEROR(4) – *double precision* array *Output*
 7: ZEROI(4) – *double precision* array *Output*
On exit: ZEROR(i) and ZEROI(i) contain the real and imaginary parts, respectively, of the i th root.
- 8: ERREST(4) – *double precision* array *Output*
On exit: ERREST(i) contains an approximate error estimate for the i th root.
- 9: IFAIL – INTEGER *Input/Output*
On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.
On exit: IFAIL = 0 unless the routine detects an error (see Section 6).
 For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

If on entry IFAIL = 0 or -1 , explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, $E = (0.0, 0.0)$.

IFAIL = 2

The companion matrix H cannot be formed without overflow.

IFAIL = 3

The iterative procedure used to determine the eigenvalues has failed to converge.

7 Accuracy

If IFAIL = 0 on exit, then the i th computed root should have approximately $|\log_{10}(\text{ERREST}(i))|$ correct significant digits.

8 Further Comments

The method used by the routine consists of the following steps, which are performed by routines from LAPACK in Chapter F08.

- (a) Form matrix H .
- (b) Apply a diagonal similarity transformation to H (to give H').
- (c) Calculate the eigenvalues and Schur factorization of H' .
- (d) Calculate the left and right eigenvectors of H' .
- (e) Estimate reciprocal condition numbers for all the eigenvalues of H' .
- (f) Calculate approximate error estimates for all the eigenvalues of H' (using the 1-norm).

9 Example

This example finds the roots of the quartic equation

$$z^4 + 16iz^2 - (8 - 8i)z - 65 = 0.$$

9.1 Program Text

```
*      CO2ANF Example Program Text
*      Mark 20 Release. NAG Copyright 2001.
*      .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*      .. Local Scalars ..
      COMPLEX *16      A, B, C, D, E
      INTEGER          I, IFAIL
*      .. Local Arrays ..
      DOUBLE PRECISION ERREST(4), ZEROI(4), ZEROR(4)
*      .. External Subroutines ..
      EXTERNAL         CO2ANF
*      .. Executable Statements ..
      WRITE (NOUT,*) 'CO2ANF Example Program Results'
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) E, A, B, C, D
      IFAIL = 1

*
      CALL CO2ANF(E,A,B,C,D,ZEROR,ZEROI,ERREST,IFAIL)
*
      IF (IFAIL.EQ.0) THEN
        WRITE (NOUT,*)
        WRITE (NOUT,*) ' Roots of quartic equation  ',
+          '          Error estimates'
        WRITE (NOUT,*) '          ',
+          '          (machine-dependent)'
        WRITE (NOUT,*)
        DO 20 I = 1, 4
          WRITE (NOUT,99999) ' z = ', ZEROR(I), ZEROI(I), '*i',
+          ERREST(I)
20      CONTINUE
      ELSE
        WRITE (NOUT,*)
        WRITE (NOUT,99998) ' ** CO2ANF returned with IFAIL = ', IFAIL
      END IF

*
99999 FORMAT (1X,A,1P,E12.4,SP,E12.4,A,8X,SS,E9.1)
99998 FORMAT (1X,A,I5)
      END
```

9.2 Program Data

```
CO2ANF Example Program Data
( 1.0,  0.0)
( 0.0,  0.0)
( 0.0, 16.0)
( -8.0,  8.0)
(-65.0,  0.0) : Values of E, A, B, C and D
```

9.3 Program Results

CO2ANF Example Program Results

Roots of quartic equation	Error estimates (machine-dependent)
z = 3.0000E+00 -2.0000E+00*i	3.0E-15
z = 1.0000E+00 -2.0000E+00*i	2.9E-15
z = -2.0000E+00 +1.0000E+00*i	2.9E-15
z = -2.0000E+00 +3.0000E+00*i	3.0E-15