C06RCF – NAG Fortran Library Routine Document

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

C06RCF computes the discrete quarter-wave Fourier sine transforms of m sequences of real data values.

2 Specification

```
SUBROUTINE C06RCF(DIRECT, M, N, X, WORK, IFAIL)
CHARACTER*1       DIRECT
INTEGER           M, N, IFAIL
real              X(M*(N+2)), WORK(M*N+2*N+15)
```

3 Description

Given m sequences of n real data values x_j^p, for $j = 1,2,\ldots,n$ and $p = 1,2,\ldots,m$, this routine simultaneously calculates the quarter-wave Fourier sine transforms of all the sequences defined by

$$\hat{x}_k^p = \frac{1}{\sqrt{n}}\left(\sum_{j=1}^{n-1} x_j^p \times \sin\left(j(2k-1)\frac{\pi}{2n}\right) + \frac{1}{2}(-1)^{k-1}x_n^p\right), \quad \text{if DIRECT = 'F'},$$

or its inverse

$$x_k^p = \frac{2}{\sqrt{n}}\sum_{j=1}^{n} \hat{x}_j^p \times \sin\left((2j-1)k\frac{\pi}{2n}\right), \quad \text{if DIRECT = 'B'},$$

for $k = 1,2,\ldots,n$ and $p = 1,2,\ldots,m$.

(Note the scale factor $\frac{1}{\sqrt{n}}$ in this definition.)

A call of the routine with DIRECT = 'F' followed by a call with DIRECT = 'B' will restore the original data.

The transform calculated by this routine can be used to solve Poisson's equation when the solution is specified at the left boundary, and the derivative of the solution is specified at the right boundary (Swarztrauber [2]).

The routine uses a variant of the fast Fourier transform (FFT) algorithm (Brigham [1]) known as the Stockham self-sorting algorithm, described in Temperton [4], together with pre- and post-processing stages described in Swarztrauber [3]. Special coding is provided for the factors 2, 3, 4 and 5.

4 References

[1] Brigham E O (1973) *The Fast Fourier Transform* Prentice–Hall

[2] Swarztrauber P N (1977) The methods of cyclic reduction, Fourier analysis and the FACR algorithm for the discrete solution of Poisson's equation on a rectangle *SIAM Rev.* **19 (3)** 490–501

[3] Swarztrauber P N (1982) Vectorizing the FFT's *Parallel Computation* (ed G Rodrique) Academic Press 51–83

[4] Temperton C (1983) Fast mixed-radix real Fourier transforms *J. Comput. Phys.* **52** 340–350

5 Parameters

1: DIRECT — CHARACTER*1 *Input*

On entry: if the **F**orward transform as defined in Section 3 is to be computed, then DIRECT must be set equal to 'F'. If the **B**ackward transform is to be computed then DIRECT must be set equal to 'B'.

Constraint: DIRECT = 'F' or 'B'.

2: M — INTEGER *Input*

On entry: the number of sequences to be transformed, m.

Constraint: M $\geq$ 1.

3: N — INTEGER *Input*

On entry: the number of real values in each sequence, n.

Constraint: N $\geq$ 1.

4: X(M∗(N+2)) — ***real*** array *Input/Output*

On entry: the data must be stored in X as if in a two-dimensional array of dimension (1:M,1:N+2); each of the m sequences is stored in a **row** of the array. In other words, if the data values of the pth sequence to be transformed are denoted by x_j^p, for $j = 1, 2, \ldots, n$ and $p = 1, 2, \ldots, m$, then the first mn elements of the array X must contain the values

$$x_1^1, x_1^2, \ldots, x_1^m,\ x_2^1, x_2^2, \ldots, x_2^m, \ldots,\ x_n^1, x_n^2, \ldots, x_n^m.$$

The $(n+1)$th and $(n+2)$th elements of each row x_{n+1}^p, x_{n+2}^p, for $p = 1, 2, \ldots, m$, are required as workspace. These $2m$ elements may contain arbitrary values as they are set to zero by the routine.

On exit: the m quarter-wave sine transforms stored as if in a two-dimensional array of dimension (1:M,1:N+2). Each of the m transforms is stored in a **row** of the array, overwriting the corresponding original sequence. If the n components of the pth quarter-wave sine transform are denoted by $\hat{x}_k^p$, for $k = 1, 2, \ldots, n$ and $p = 1, 2, \ldots, m$, then the $m(n+2)$ elements of the array X contain the values

$$\hat{x}_1^1, \hat{x}_1^2, \ldots, \hat{x}_1^m,\ \hat{x}_2^1, \hat{x}_2^2, \ldots, \hat{x}_2^m, \ldots,\ \hat{x}_n^1, \hat{x}_n^2, \ldots, \hat{x}_n^m, 0, 0, \ldots, 0 \text{ (}2m\text{ times).}$$

5: WORK(M∗N+2∗N+15) — ***real*** array *Workspace*

The workspace requirements as documented for this routine may be an overestimate in some implementations. For full details of the workspace required by this routine please refer to the Users' Note for your implementation.

On exit: WORK(1) contains the minimum workspace required for the current values of M and N with this implementation.

6: IFAIL — INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. For users not familiar with this parameter (described in Chapter P01) the recommended value is 0.

On exit: IFAIL = 0 unless the routine detects an error (see Section 6).

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 detected by the routine:

IFAIL = 1

On entry, M < 1.

IFAIL = 2

On entry, N < 1.

IFAIL = 3

On entry, DIRECT is not equal to one of 'F' or 'B'.

IFAIL = 4

An unexpected error has occurred in an internal call. Check all subroutine calls and array dimensions. Seek expert help.

7 Accuracy

Some indication of accuracy can be obtained by performing a subsequent inverse transform and comparing the results with the original sequence (in exact arithmetic they would be identical).

8 Further Comments

The time taken by the routine is approximately proportional to $nm \times \log n$, but also depends on the factors of n. The routine is fastest if the only prime factors of n are 2, 3 and 5, and is particularly slow if n is a large prime, or has large prime factors.

9 Example

This program reads in sequences of real data values and prints their quarter-wave sine transforms as computed by C06RCF with DIRECT = 'F'. It then calls the routine again with DIRECT = 'B' and prints the results which may be compared with the original data.

9.1 Program Text

```
*     C06RCF Example Program Text.
*     Mark 19 Release. NAG Copyright 1999.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MMAX, NMAX
      PARAMETER        (MMAX=5,NMAX=20)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, M, N
*     .. Local Arrays ..
      real             WORK(MMAX*NMAX+2*NMAX+15), X((NMAX+2)*MMAX)
*     .. External Subroutines ..
      EXTERNAL         C06RCF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'C06RCF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
   20 CONTINUE
      READ (NIN,*,END=120) M, N
      IF (M.LE.MMAX .AND. N.LE.NMAX) THEN
         DO 40 J = 1, M
            READ (NIN,*) (X(I*M+J),I=0,N-1)
   40    CONTINUE
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Original data values'
         WRITE (NOUT,*)
```

```
         DO 60 J = 1, M
            WRITE (NOUT,99999) (X(I*M+J),I=0,N-1)
   60    CONTINUE
         IFAIL = 0
*
*        -- Compute transform
         CALL C06RCF('Forward',M,N,X,WORK,IFAIL)
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Discrete quarter-wave Fourier sine transforms'
         WRITE (NOUT,*)
         DO 80 J = 1, M
            WRITE (NOUT,99999) (X(I*M+J),I=0,N-1)
   80    CONTINUE
*
*        -- Compute inverse transform
         CALL C06RCF('Backward',M,N,X,WORK,IFAIL)
*
         WRITE (NOUT,*)
         WRITE (NOUT,*) 'Original data as restored by inverse transform'
         WRITE (NOUT,*)
         DO 100 J = 1, M
            WRITE (NOUT,99999) (X(I*M+J),I=0,N-1)
  100    CONTINUE
         GO TO 20
      ELSE
         WRITE (NOUT,*) 'Invalid value of M or N'
      END IF
  120 CONTINUE
      STOP
*
99999 FORMAT (6X,7F10.4)
      END
```

9.2 Program Data

```
C06RCF Example Program Data
3  6 : Number of sequences, M, and number of values in each sequence, N
 0.3854  0.6772  0.1138  0.6751  0.6362  0.1424  : X, sequence 1
 0.5417  0.2983  0.1181  0.7255  0.8638  0.8723  : X, sequence 2
 0.9172  0.0644  0.6037  0.6430  0.0428  0.4815  : X, sequence 3
```

9.3 Program Results

```
 C06RCF Example Program Results

 Original data values

           0.3854    0.6772    0.1138    0.6751    0.6362    0.1424
           0.5417    0.2983    0.1181    0.7255    0.8638    0.8723
           0.9172    0.0644    0.6037    0.6430    0.0428    0.4815

 Discrete quarter-wave Fourier sine transforms

           0.7304    0.2078    0.1150    0.2577   -0.2869   -0.0815
           0.9274   -0.1152    0.2532    0.2883   -0.0026   -0.0635
           0.6268    0.3547    0.0760    0.3078    0.4987   -0.0507
```

```
Original data as restored by inverse transform

         0.3854    0.6772    0.1138    0.6751    0.6362    0.1424
         0.5417    0.2983    0.1181    0.7255    0.8638    0.8723
         0.9172    0.0644    0.6037    0.6430    0.0428    0.4815
```