NAG Fortran Library Routine Document

G13CFF

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

For a bivariate time series, G13CFF calculates the gain and phase together with lower and upper bounds from the univariate and bivariate spectra.

2 Specification

```
      SUBROUTINE G13CFF(XG, YG, XYRG, XYIG, NG, STATS, GN, GNLW, GNUP, PH,
     1                  PHLW, PHUP, IFAIL)
      INTEGER           NG, IFAIL
      real              XG(NG), YG(NG), XYRG(NG), XYIG(NG), STATS(4), GN(NG),
     1                  GNLW(NG), GNUP(NG), PH(NG), PHLW(NG), PHUP(NG)
```

3 Description

Estimates of the gain $G(\omega)$ and phase $\phi(\omega)$ of the dependency of series y on series x at frequency ω are given by

$$\hat{G}(\omega) = \frac{A(\omega)}{f_{xx}(\omega)}$$

$$\hat{\phi}(\omega) = \cos^{-1}\left(\frac{cf(\omega)}{A(\omega)}\right), \qquad \text{if } qf(\omega) \geq 0$$

$$\hat{\phi}(\omega) = 2\pi - \cos^{-1}\left(\frac{cf(\omega)}{A(\omega)}\right), \quad \text{if } qf(\omega) < 0.$$

The quantities used in these definitions are obtained as in Section 3 of the document for G13CEF.

Confidence limits are returned for both gain and phase, but should again be taken as very approximate when the coherency $W(\omega)$, as calculated by G13CEF, is not significant. These are based on the assumption that both $(\hat{G}(\omega)/G(\omega)) - 1$ and $\hat{\phi}(\omega)$ are Normal with variance

$$\frac{1}{d}\left(\frac{1}{W(\omega)} - 1\right).$$

Although the estimate of $\phi(\omega)$ is always given in the range $[0, 2\pi)$, no attempt is made to restrict its confidence limits to this range.

4 References

Jenkins G M and Watts D G (1968) *Spectral Analysis and its Applications* Holden-Day

Bloomfield P (1976) *Fourier Analysis of Time Series: An Introduction* Wiley

5 Parameters

1: XG(NG) – ***real*** array *Input*

On entry: the NG univariate spectral estimates, $f_{xx}(\omega)$, for the x series.

2: YG(NG) – ***real*** array *Input*

On entry: the NG univariate spectral estimates, $f_{yy}(\omega)$, for the y series.

3: XYRG(NG) – ***real*** array *Input*

On entry: the real parts, $cf(\omega)$ of the NG bivariate spectral estimates for the x and y series. The x series leads the y series.

4: XYIG(NG) – ***real*** array *Input*

On entry: the imaginary parts, $qf(\omega)$, of the NG bivariate spectral estimates for the x and y series. The x series leads the y series.

Note: the two univariate and the bivariate spectra must each have been calculated using the same method of smoothing. For rectangular, Bartlett, Tukey or Parzen smoothing windows, the same cut-off point of lag window and the same frequency division of the spectral estimates must be used. For the trapezium frequency smoothing window, the frequency width and the shape of the window and the frequency division of the spectral estimates must be the same. The spectral estimates and statistics must also be unlogged.

5: NG – INTEGER *Input*

On entry: the number of spectral estimates in each of the arrays XG, YG, XYRG and XYIG. It is also the number of gain and phase estimates.

Constraint: NG $\geq$ 1.

6: STATS(4) – ***real*** array *Input*

On entry: the four associated statistics for the univariate spectral estimates for the x and y series. STATS(1) contains the degrees of freedom, STATS(2) and STATS(3) contain the lower and upper bound multiplying factors respectively and STATS(4) holds the bandwidth.

Constraint: STATS$(1) \geq 3.0$.

7: GN(NG) – ***real*** array *Output*

On exit: the NG gain estimates, $\hat{G}(\omega)$, at each frequency ω.

8: GNLW(NG) – ***real*** array *Output*

On exit: the NG lower bounds for the NG gain estimates.

9: GNUP(NG) – ***real*** array *Output*

On exit: the NG upper bounds for the NG gain estimates.

10: PH(NG) – ***real*** array *Output*

On exit: the NG phase estimates, $\hat{\phi}(\omega)$, at each frequency ω.

11: PHLW(NG) – ***real*** array *Output*

On exit: the NG lower bounds for the NG phase estimates.

12: PHUP(NG) – ***real*** array *Output*

On exit: the NG upper bounds for the NG phase estimates.

13: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 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, for users 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, NG < 1,
or STATS(1) < 3.0.

IFAIL $= 2$

A bivariate spectral estimate is zero. For this frequency the gain and the phase and their bounds are set to zero.

IFAIL $= 3$

A univariate spectral estimate is negative. For this frequency the gain and the phase and their bounds are set to zero.

IFAIL $= 4$

A univariate spectral estimate is zero. For this frequency the gain and the phase and their bounds are set to zero.

IFAIL $= 5$

A calculated value of the squared coherency exceeds 1.0. For this frequency the squared coherency is reset to 1.0 in the formulae for the gain and phase bounds.

If more than one failure of types 2, 3, 4 and 5 occurs then the failure type which occurred at lowest frequency is returned in IFAIL. However the actions indicated above are also carried out for failures at higher frequencies.

7 Accuracy

All computations are very stable and yield good accuracy.

8 Further Comments

The time taken by the routine is approximately proportional to NG.

9 Example

The example program reads the set of univariate spectrum statistics, the two univariate spectra and the cross spectrum at a frequency division of $\frac{2\pi}{20}$ for a pair of time series. It calls G13CFF to calculate the gain and the phase and their bounds and prints the results.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G13CFF Example Program Text
*     Mark 14 Revised.  NAG Copyright 1989.
*     .. Parameters ..
      INTEGER          NGMAX
      PARAMETER        (NGMAX=9)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
*     .. Local Scalars ..
      INTEGER          I, IFAIL, J, NG
*     .. Local Arrays ..
      real             GN(NGMAX), GNLW(NGMAX), GNUP(NGMAX), PH(NGMAX),
     +                 PHLW(NGMAX), PHUP(NGMAX), STATS(4), XG(NGMAX),
     +                 XYIG(NGMAX), XYRG(NGMAX), YG(NGMAX)
*     .. External Subroutines ..
      EXTERNAL         G13CFF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G13CFF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) NG
      IF (NG.GT.0 .AND. NG.LE.NGMAX) THEN
         READ (NIN,*) (STATS(I),I=1,4)
         READ (NIN,*) (XG(I),YG(I),XYRG(I),XYIG(I),I=1,NG)
         IFAIL = 1
*
         CALL G13CFF(XG,YG,XYRG,XYIG,NG,STATS,GN,GNLW,GNUP,PH,PHLW,PHUP,
     +               IFAIL)
*
         WRITE (NOUT,*)
         IF (IFAIL.NE.0) THEN
            WRITE (NOUT,99999) 'G13CFF fails. IFAIL =', IFAIL
            WRITE (NOUT,*)
         END IF
         IF (IFAIL.NE.1) THEN
            WRITE (NOUT,*) '              The gain'
            WRITE (NOUT,*)
            WRITE (NOUT,*) '                    Lower     Upper'
            WRITE (NOUT,*) '          Value     bound     bound'
            DO 20 J = 1, NG
               WRITE (NOUT,99998) J - 1, GN(J), GNLW(J), GNUP(J)
   20       CONTINUE
            WRITE (NOUT,*)
            WRITE (NOUT,*) '             The phase'
            WRITE (NOUT,*)
            WRITE (NOUT,*) '                    Lower     Upper'
            WRITE (NOUT,*) '          Value     bound     bound'
            DO 40 J = 1, NG
               WRITE (NOUT,99998) J - 1, PH(J), PHLW(J), PHUP(J)
   40       CONTINUE
         END IF
      END IF
      STOP
*
99999 FORMAT (1X,A,I3)
99998 FORMAT (1X,I5,3F10.4)
      END
```

9.2 Program Data

```
G13CFF Example Program Data
     9
 30.00000   .63858  1.78670   .33288
  2.03490 21.97712 -6.54995  0.00000
   .51554  3.29761   .34107 -1.19030
   .07640   .28782   .12335   .04087
   .01068   .02480  -.00514   .00842
   .00093   .00285  -.00033   .00032
   .00100   .00203  -.00039  -.00001
   .00076   .00125  -.00026   .00018
   .00037   .00107   .00011  -.00016
   .00021   .00191   .00007  0.00000
```

9.3 Program Results

```
 G13CFF Example Program Results

               The gain

                     Lower     Upper
         Value       bound     bound
     0    3.2188     2.9722    3.4859
     1    2.4018     2.1138    2.7290
     2    1.7008     1.3748    2.1042
     3    0.9237     0.5558    1.5350
     4    0.4943     0.1327    1.8415
     5    0.3901     0.1002    1.5196
     6    0.4161     0.1346    1.2863
     7    0.5248     0.1591    1.7306
     8    0.3333     0.0103   10.8301

              The phase

                     Lower     Upper
         Value       bound     bound
     0    3.1416     3.0619    3.2213
     1    4.9915     4.8637    5.1192
     2    0.3199     0.1071    0.5328
     3    2.1189     1.6109    2.6268
     4    2.3716     1.0563    3.6868
     5    3.1672     1.8075    4.5270
     6    2.5360     1.4074    3.6647
     7    5.3147     4.1214    6.5079
     8    0.0000    -3.4809    3.4809
```