

# NAG Fortran Library Routine Document

## G13BHF

**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

G13BHF produces forecasts of a time series (the output series) which depends on one or more other (input) series via a multi-input model which will usually have been fitted using G13BEF. The future values of the input series must be supplied. The original observations are not required. G13BHF uses as input either the original state set obtained from G13BEF, or the state set updated by a series of new observations from G13BGF. Standard errors of the forecasts are produced. If future values of some of the input series have been obtained as forecasts using ARIMA models for those series, this may be allowed for in the calculation of the standard errors.

### 2 Specification

```

SUBROUTINE G13BHF(STTF, NSTTF, MR, NSER, MT, PARA, NPARA, NFV, XXYN,
1                IXXYN, MRX, PARX, IPARX, RMSXY, KZEF, FVA, FSD, WA,
2                IWA, IFAIL)
    INTEGER       NSTTF, MR(7), NSER, MT(4,NSER), NPARA, NFV, IXXYN,
1                MRX(7,NSER), IPARX, KZEF, IWA, IFAIL
    real         STTF(NSTTF), PARA(NPARA), XXYN(IXXYN,NSER),
1                PARX(IPARX,NSER), RMSXY(NSER), FVA(NFV), FSD(NFV),
2                WA(IWA)

```

### 3 Description

The forecasts of the output series  $y_t$  are calculated for  $t = n + 1, n + 2, \dots, n + L$ , where  $n$  is the latest time point of the observations used to produce the state set and  $L$  is the maximum lead time of the forecasts.

First the new input series values  $x_t$  are used to form the input components  $z_t$  for  $t = n + 1, n + 2, \dots, n + L$  using the transfer function models:

$$z_t = \delta_1 z_{t-1} + \delta_2 z_{t-2} + \dots + \delta_p z_{t-p} + \omega_0 x_{t-b} - \omega_1 x_{t-b-1} - \dots - \omega_q x_{t-b-q}.$$

The output noise component  $n_t$  for  $t = n + 1, n + 2, \dots, n + L$  is then forecast by setting  $a_t = 0$  for  $t = n + 1, n + 2, \dots, n + L$  and using the ARIMA model equations:

$$e_t = \phi_1 e_{t-1} + \phi_2 e_{t-2} + \dots + \phi_p e_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_1 a_{t-q}$$

$$w_t = \Phi_1 w_{t-s} + \Phi_2 w_{t-2 \times s} + \dots + \Phi_P w_{t-P \times s} + e_t - \Theta_1 e_{t-s} - \Theta_2 e_{t-2 \times s} - \dots - \Theta_Q e_{t-Q \times s}$$

$$n_t = (\nabla^d \nabla_s^D)^{-1} (w_t + c).$$

This last step of 'integration' reverses the process of differencing. Finally the output forecasts are calculated as

$$y_t = z_{1,t} + z_{2,t} + \dots + z_{m,t} + n_t.$$

The forecast error variance of  $y_{t+l}$  (i.e., at lead time  $l$ ) is  $S_l^2$ , which is the sum of parts which arise from the various input series, and the output noise component. That part due to the output noise is

$$sn_l^2 = V_n \times (\psi_0^2 + \psi_1^2 + \dots + \psi_{l-1}^2),$$

where  $V_n$  is the estimated residual variance of the output noise ARIMA model, and  $\psi_0, \psi_1, \dots$  are the 'psi-weights' of this model as defined in Box and Jenkins (1976). They are calculated by applying the equations (b), (c) and (d) above for  $t = 0, 1, \dots, L$ , but with artificial values for the various series and with the constant  $c$  set to 0. Thus all values of  $a_t, e_t, w_t$  and  $n_t$  are taken as zero for  $t < 0$ ;  $a_t$  is taken to be 1

for  $t = 0$  and 0 for  $t > 0$ . The resulting values of  $n_t$  for  $t = 0, 1, \dots, L$  are precisely  $\psi_0, \psi_1, \dots, \psi_L$  as required.

Further contributions to  $S_t^2$  come only from those input series, for which future values are forecasts which have been obtained by applying input series ARIMA models. For such a series the contribution is

$$sz_t^2 = V_x \times (\nu_0^2 + \nu_1^2 + \dots + \nu_{t-1}^2),$$

where  $V_x$  is the estimated residual variance of the input series ARIMA model. The coefficients  $\nu_0, \nu_1, \dots$  are calculated by applying the transfer function model equation (a) above for  $t = 0, 1, \dots, L$ , but again with artificial values of the series. Thus all values of  $z_t$  and  $x_t$  for  $t < 0$  are taken to be zero, and  $x_0, x_1, \dots$  are taken to be the psi-weight sequence  $\psi_0, \psi_1, \dots$  for the **input series** ARIMA model. The resulting values of  $z_t$  for  $t = 0, 1, \dots, L$  are precisely  $\nu_0, \nu_1, \dots, \nu_L$  as required.

In adding such contributions  $sz_t^2$  to  $sn_t^2$  to make up the total forecast error variance  $S_t^2$ , it is assumed that the various input series with which these contributions are associated are statistically independent of each other.

When using the routine in practice an ARIMA model is required for all the input series. In the case of those inputs for which no such ARIMA model is available (or its effects are to be excluded), the corresponding orders and parameters and the estimated residual variance should be set to zero.

## 4 References

Box G E P and Jenkins G M (1976) *Time Series Analysis: Forecasting and Control* (Revised Edition) Holden-Day

## 5 Parameters

- 1: STTF(NSTTF) – *real* array *Input*  
*On entry:* the NSTTF values in the state set as returned by G13BEF or G13BGF.
- 2: NSTTF – INTEGER *Input*  
*On entry:* the exact number of values in the state set array STTF as returned by G13BEF or G13BGF.
- 3: MR(7) – INTEGER array *Input/Output*  
*On entry:* the orders vector  $(p, d, q, P, D, Q, s)$  of the ARIMA model for the output noise component.  
 $p, q, P$  and  $Q$  give respectively the number of autoregressive ( $\phi$ ), moving average ( $\theta$ ), seasonal autoregressive ( $\Phi$ ) and seasonal moving average ( $\Theta$ ) parameters.  
 $d, D$  and  $s$  refer respectively to the order of non-seasonal differencing, the order of seasonal differencing, and the seasonal period.  
*On exit:* used as internal workspace prior to being restored and hence is unchanged.
- 4: NSER – INTEGER *Input*  
*On entry:* the total number of input and output series. There may be any number of input series (including none), but only one output series.
- 5: MT(4,NSER) – INTEGER array *Input*  
*On entry:* the transfer function orders  $b, p$  and  $q$  of each of the input series. The data for input series  $i$  are held in column  $i$ . Row 1 holds the value  $b_i$ , row 2 holds the value  $q_i$  and row 3 holds the value  $p_i$ . For a simple input,  $b_i = q_i = p_i = 0$ .  
Row 4 holds the value  $r_i$ , where  $r_i = 1$  for a simple input,  $r_i = 2$  or 3 for a transfer function input. When  $r_i = 1$ , any non-zero contents of rows 1, 2 and 3 of column  $i$  are ignored. The choice of

$r_i = 2$  or  $r_i = 3$  is an option for use in model estimation and does not affect the operation of this routine.

*Constraint:*  $MT(4, i) = 1, 2$  or  $3$ , for  $i = 1, 2, \dots, NSER - 1$ .

- 6: PARA(NPARA) – *real* array *Input*  
*On entry:* estimates of the multi-input model parameters as returned by G13BEF. These are in order, firstly the ARIMA model parameters:  $p$  values of  $\phi$  parameters,  $q$  values of  $\theta$  parameters,  $P$  values of  $\Phi$  parameters and  $Q$  values of  $\Theta$  parameters. These are followed by the transfer function model parameter values  $\omega_0, \omega_1, \dots, \omega_{q_1}, \delta_1, \delta_2, \dots, \delta_{p_1}$  for the first of any input series and similar sets of values for any subsequent input series. The final component of PARA is the constant  $c$ .
- 7: NPARA – INTEGER *Input*  
*On entry:* the exact number of  $\phi, \theta, \Phi, \Theta, \omega, \delta$  and  $c$  parameters. ( $c$  must be included, whether its value was previously estimated or was set fixed).
- 8: NFV – INTEGER *Input*  
*On entry:* the number of forecast values required.
- 9: XXYN(IXXYN, NSER) – *real* array *Input/Output*  
*On entry:* the supplied NFV values for each of the input series required to produce the NFV output series forecasts. Column  $i$  contains the values for input series  $i$ . Column NSER need not be supplied.  
*On exit:* if KZEF = 0, then column NSER of XXYN contains the output series forecast values (as does FVA), but XXYN is otherwise unchanged.  
 If KZEF  $\neq$  0, then the columns of XXYN hold the corresponding values of the forecast components  $z_t$  for each of the input series and the values of the output noise component  $n_t$  in that order.
- 10: IXXYN – INTEGER *Input*  
*On entry:* the first dimension of the array XXYN as declared in the (sub)program from which G13BHF is called.  
*Constraint:*  $IXXYN \geq NFV$ .
- 11: MRX(7, NSER) – INTEGER array *Input/Output*  
*On entry:* the orders array for each of the input series ARIMA models. Thus, column  $i$  contains values of  $p, d, q, P, D, Q, s$  for input series  $i$ . In the case of those inputs for which no ARIMA model is available, the corresponding orders should be set to 0. (The model for any input series only affects the standard errors of the forecast values.)  
*On exit:* unchanged, apart from column NSER which is used for workspace.
- 12: PARX(IPARX, NSER) – *real* array *Input*  
*On entry:* values of the parameters ( $\phi, \theta, \Phi$  and  $\Theta$ ) for each of the input series ARIMA models. Thus column  $i$  contains  $MRX(1, i)$  values of  $\phi$  parameters,  $MRX(3, i)$  values of  $\theta$  parameters,  $MRX(4, i)$  values of  $\Phi$  parameters and  $MRX(6, i)$  values of  $\Theta$  parameters – in that order.  
 Values in the columns relating to those input series for which no ARIMA model is available are ignored. (The model for any input series only affects the standard errors of the forecast values.)
- 13: IPARX – INTEGER *Input*  
*On entry:* the dimension of the array PARX as declared in the (sub)program from which G13BHF is called.  
*Constraint:*  $IPARX \geq ncd$ , where  $ncd$  is the maximum number of parameters in any of the input series ARIMA models. If there are no input series,  $IPARX \geq 1$ .

- 14: RMSXY(NSER) – *real* array *Input*  
*On entry:* the estimated residual variances for each input series ARIMA model followed by that for the output noise ARIMA model. In the case of those inputs for which no ARIMA model is available, or when its effects are to be excluded in the calculation of forecast standard errors, the corresponding entry of RMSXY should be set to 0.
- 15: KZEF – INTEGER *Input*  
*On entry:* KZEF must not be set to 0, if the values of the input component series  $z_t$  and the values of the output noise component  $n_t$  are to overwrite the contents of XXYN on exit, and must be set to 0 if XXYN is to remain unchanged on exit, apart from the appearance of the forecast values in column NSER.
- 16: FVA(NFV) – *real* array *Output*  
*On exit:* the required forecast values for the output series.
- 17: FSD(NFV) – *real* array *Output*  
*On exit:* the standard errors for each of the forecast values.
- 18: WA(IWA) – *real* array *Workspace*  
 19: IWA – INTEGER *Input*  
*On entry:* the dimension of the array WA as declared in the (sub)program from which G13BHF is called.  
 A good, slightly conservative approximation to the required size of IWA is given by
 
$$IWA \geq 4 \times (NSTTF + NFV + ncf)$$
 where  $ncf$  is the largest number of ARIMA parameters in any one of the input or output series.  
 An exact value for the required size of IWA can be calculated as follows:  
 Let  $ncg = \max(p_i)$ ,  
 $nch = \max(b_i + q_i)$ ,  
 $nci = \max(b_i + q_i + p_i)$ ,  
 over each of the transfer function input series for which  $r_i > 1$ , where  $b_i, q_i, p_i$  are the orders held in rows 1 to 3 of array MT.  
 Let  $ncj = 1 + nci$ ,  
 $nck = NFV + \max(ncg, nch)$ ,  
 $ncl = \max(NSTTF, ncf, ncj, nck)$ ,  
 $ncm = \max(NSTTF + 4 \times ncf, ncl)$ ,  
 then  $IWA \geq ncm + 3 \times ncl + NFV$ .
- 20: 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, NSTTF is not consistent with the orders in arrays MR and MT.

$IFAIL = 2$

On entry, NPARA is not consistent with the orders in arrays MR and MT.

$IFAIL = 3$

On entry, IXXYN is too small.

$IFAIL = 4$

On entry, IWA is too small.

$IFAIL = 5$

On entry, IPARX is too small.

$IFAIL = 6$

On entry, one of the  $r_i$ , stored in  $MT(4, i)$ , for  $i = 1, 2, \dots, NSER - 1$ , does not equal 1, 2 or 3.

## 7 Accuracy

The computations are believed to be stable.

## 8 Further Comments

The time taken by the routine is approximately proportional to  $NFV \times NPARA$ .

## 9 Example

The example follows up that described in G13BGF and makes use of its data. These consist of output series orders and parameter values, input series transfer function orders and the updated state set.

Four new values of the input series are supplied, as are the orders and parameter values for the single input series ARIMA model (which has 2 values of  $\phi$ , 2 values of  $\theta$ , 1 value of  $\Theta$ , single seasonal differencing and a seasonal period of 4), and the estimated residual variances for the input series ARIMA model and the output noise ARIMA model.

Four forecast values and their standard errors are computed and printed; also the values of the components  $z_t$  and the output noise component  $n_t$  corresponding to the forecasts.

### 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.

```
*      G13BHF Example Program Text
*      Mark 14 Revised.  NAG Copyright 1989.
*      .. Parameters ..
  INTEGER          NSERMX, NFVMAX, NPMAX, IPARX, NSTTFM, IXXYN, IWA
  PARAMETER       (NSERMX=2, NFVMAX=40, NPMAX=10, IPARX=NPMAX,
+                NSTTFM=20, IXXYN=NFVMAX, IWA=100)
  INTEGER          NIN, NOUT
```

```

PARAMETER      (NIN=5,NOUT=6)
*   .. Local Scalars ..
INTEGER        I, IFAIL, J, KZEF, NFV, NPARA, NSER, NSTTF
*   .. Local Arrays ..
real         FSD(NFVMAX), FVA(NFVMAX), PARA(NPMAX),
+           PARX(IPARX,NSERMX), RMSXY(NSERMX), STTF(NSTTFM),
+           WA(IWA), XXYN(IXXYN,NSERMX)
INTEGER        MR(7), MRX(7,NSERMX), MT(4,NSERMX)
*   .. External Subroutines ..
EXTERNAL      G13BHF
*   .. Executable Statements ..
WRITE (NOUT,*) 'G13BHF Example Program Results'
*   Skip heading in data file
READ (NIN,*)
READ (NIN,*) NSTTF, NSER, NFV, KZEF
IF (NSTTF.GT.0 .AND. NSTTF.LE.NSTTFM .AND. NSER.GT.0 .AND.
+   NSER.LE.NSERMX .AND. NFV.GT.0 .AND. NFV.LE.NFVMAX) THEN
  READ (NIN,*) (MR(I),I=1,7)
  DO 20 I = 1, 4
    READ (NIN,*) (MT(I,J),J=1,NSER)
20  CONTINUE
  NPARA = 0
  DO 40 I = 1, NSER
    NPARA = NPARA + MT(2,I) + MT(3,I)
40  CONTINUE
  NPARA = NPARA + MR(1) + MR(3) + MR(4) + MR(6) + NSER
  READ (NIN,*) (STTF(I),I=1,NSTTF)
  IF (NPARA.LE.NPMAX) THEN
    READ (NIN,*) (PARA(I),I=1,NPARA)
    DO 60 I = 1, NFV
      READ (NIN,*) (XXYN(I,J),J=1,NSER)
60  CONTINUE
    DO 80 I = 1, 7
      READ (NIN,*) (MRX(I,J),J=1,NSER)
80  CONTINUE
    DO 100 I = 1, NPARA
      READ (NIN,*) (PARX(I,J),J=1,NSER)
100 CONTINUE
    READ (NIN,*) (RMSXY(I),I=1,NSER)
    IFAIL = 0
*
+   CALL G13BHF(STTF,NSTTF,MR,NSER,MT,PARA,NPARA,NFV,XXYN,IXXYN,
+             MRX,PARX,IPARX,RMSXY,KZEF,FVA,FSD,WA,IWA,IFAIL)
*
  WRITE (NOUT,*)
  WRITE (NOUT,*)
+   'The forecast values and their standard errors'
  WRITE (NOUT,*)
  WRITE (NOUT,*) ' I      FVA      FSD'
  WRITE (NOUT,*)
  DO 120 I = 1, NFV
    WRITE (NOUT,99999) I, FVA(I), FSD(I)
120 CONTINUE
  WRITE (NOUT,*)
  WRITE (NOUT,*) 'The values of z(t) and n(t)'
  DO 140 I = 1, NFV
    WRITE (NOUT,99999) I, (XXYN(I,J),J=1,NSER)
140 CONTINUE
  END IF
END IF
STOP
*
99999 FORMAT (1X,I4,2F10.4)
END

```

## 9.2 Program Data

G13BHF Example Program Data

```

10      2      4      1
  1      0      0      0      1      1      4
  1      0
  0      0
  1      0
  3      0
  6.7160 158.3022 -80.3352 -74.8937 -80.7694 -70.3022  0.8476 -2.0234
-5.8080 10.2943
  0.5158  0.9994  8.6343  0.6726 -0.3172
  6.923  0.0000
  6.939  0.0000
  6.705  0.0000
  6.914  0.0000
  2      0
  0      0
  2      0
  0      0
  1      0
  1      0
  4      0
  1.6743  0.0
-0.9505  0.0
  1.4605  0.0
-0.4862  0.0
  0.8993  0.0
  0.1720 22.9256

```

## 9.3 Program Results

G13BHF Example Program Results

The forecast values and their standard errors

| I | FVA      | FSD    |
|---|----------|--------|
| 1 | 88.2723  | 4.7881 |
| 2 | 99.9425  | 6.4690 |
| 3 | 100.6499 | 7.3175 |
| 4 | 95.0958  | 7.5534 |

The values of  $z(t)$  and  $n(t)$

|   |          |          |
|---|----------|----------|
| 1 | 164.4620 | -76.1897 |
| 2 | 170.3924 | -70.4499 |
| 3 | 174.5193 | -73.8694 |
| 4 | 175.2747 | -80.1789 |

---