

# NAG Fortran Library Routine Document

## G13BBF

**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

G13BBF filters a time series by a transfer function model.

### 2 Specification

```
SUBROUTINE G13BBF(Y, NY, MR, NMR, PAR, NPAR, CY, WA, IWA, B, NB, IFAIL)
INTEGER          NY, MR(NMR), NMR, NPAR, IWA, NB, IFAIL
real           Y(NY), PAR(NPAR), CY, WA(IWA), B(NB)
```

### 3 Description

From a given series  $y_1, y_2, \dots, y_n$  a new series  $b_1, b_2, \dots, b_n$  is calculated using a supplied (filtering) transfer function model according to the equation

$$b_t = \delta_1 b_{t-1} + \delta_2 b_{t-2} + \dots + \delta_p b_{t-p} + \omega_0 y_{t-b} - \omega_1 y_{t-b-1} - \dots - \omega_q y_{t-b-q}. \quad (1)$$

As in the use of G13BAF, large transient errors may arise in the early values of  $b_t$  due to ignorance of  $y_t$  for  $t < 0$ , and two possibilities are allowed.

- (i) The equation (1) is applied from  $t = 1 + b + q, \dots, n$  so all terms in  $y_t$  on the right-hand side of (1) are known, the unknown set of values  $b_t$  for  $t = b + q, \dots, b + q + 1 - p$  being taken as zero.
- (ii) The unknown values of  $y_t$  for  $t \leq 0$  are estimated by backforecasting exactly as for G13BAF.

### 4 References

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

### 5 Parameters

- 1:  $Y(NY)$  – **real** array *Input*  
*On entry:* the  $Q'_y$  backforecasts starting with backforecast at time  $1 - Q'_y$  to backforecast at time 0 followed by the time series starting at time 1, where  $Q'_y = MR(6) + MR(9) \times MR(10)$ . If there are no backforecasts either because the ARIMA model for the time series is not known or because it is known but has no moving average terms, then the time series starts at the beginning of Y.
- 2: NY – INTEGER *Input*  
*On entry:* the total number of backforecasts and time series data points in array Y.  
*Constraint:*  $NY \geq \max(1 + Q'_y, NPAR)$ .
- 3: MR(NMR) – INTEGER array *Input*  
*On entry:* the orders vector for the filtering TF model followed by the orders vector for the ARIMA model for the time series if the latter is known. The TF model orders appear in the standard form  $(b, q, p)$  as given in the G13 Chapter Introduction. Note that if the ARIMA model for the time series is supplied then the routine will assume that the first  $Q'_y$  values of the array Y are backforecasts.

*Constraints:* the filtering model is restricted in the following way:

$$\text{MR}(1), \text{MR}(2), \text{MR}(3) \geq 0.$$

the ARIMA model for the time series is restricted in the following ways:

$$\text{MR}(k) \geq 0; k = 4, \dots, 10;$$

$$\text{MR}(7) + \text{MR}(8) + \text{MR}(9) = 0 \text{ if and only if } \text{MR}(10) = 0;$$

$$\text{MR}(10) \neq 1.$$

- 4: NMR – INTEGER *Input*
- On entry:* the number of values supplied in the array MR. It takes the value 3 if no ARIMA model for the time series is supplied but otherwise it takes the value 10. Thus NMR acts as an indicator as to whether backforecasting can be carried out.
- Constraint:* NMR = 3 or 10.
- 5: PAR(NPAR) – *real* array *Input*
- On entry:* the parameters of the filtering TF model followed by the parameters of the ARIMA model for the time series. In the TF model the parameters are in the standard order of MA-like followed by AR-like operator parameters. In the ARIMA model the parameters are in the standard order of non-seasonal AR and MA followed by seasonal AR and MA.
- 6: NPAR – INTEGER *Input*
- On entry:* the total number of parameters held in array PAR.
- Constraints:*
- if NMR = 3, NPAR = MR(2) + MR(3) + 1,  
if NMR = 10, NPAR = MR(2) + MR(3) + 1 + MR(4) + MR(6) + MR(7) + MR(9).
- 7: CY – *real* *Input*
- On entry:* if the ARIMA model is known (i.e., NMR=10), CY must specify the constant term of the ARIMA model for the time series. If this model is not known (i.e., NMR=3) then CY is not used.
- 8: WA(IWA) – *real* array *Workspace*
- 9: IWA – INTEGER *Input*
- On entry:* the dimension of the array WA as declared in the (sub)program from which G13BBF is called.
- Constraints:*
- let  $K = \text{MR}(3) + \text{MR}(4) + \text{MR}(5) + (\text{MR}(7) + \text{MR}(8)) \times \text{MR}(10)$ , then  
if NMR = 3, IWA  $\geq \text{MR}(1) + \text{NPAR}$ ,  
if NMR = 10, IWA  $\geq \text{MR}(1) + \text{NPAR} + K \times (K + 2)$ .
- 10: B(NB) – *real* array *Output*
- On exit:* the filtered output series. If the ARIMA model for the time series was known and hence  $Q'_y$  backforecasts were supplied in Y then B contains  $Q'_y$  ‘filtered’ backforecasts followed by the filtered series. Otherwise the filtered series begins at the start of B just as the original series began at the start of Y. In either case if the value of the series at time  $t$  is held in  $Y(t)$  then the filtered value at time  $t$  is held in  $B(t)$ .

11: NB – INTEGER *Input*

*On entry:* the dimension of the array B as declared in the (sub)program from which G13BBF is called.

In addition to holding the returned filtered series, B is also used as an intermediate work array if the ARIMA model for the time series is known.

*Constraints:*

if  $NMR = 3$ ,  $NB \geq NY$ ,  
if  $NMR = 10$ ,  $NB \geq NY + \max(MR(1) + MR(2), MR(3))$ .

12: 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,  $NMR \neq 3$  and  $NMR \neq 10$ ,  
or  $MR(i) < 0$ ,  $i = 1, 2, \dots, NMR$ ,  
or  $NMR = 10$  and  $MR(10) = 1$ ,  
or  $NMR = 10$  and  $MR(10) = 0$  and  $MR(7) + MR(8) + MR(9) \neq 0$ ,  
or  $NMR = 10$  and  $MR(10) \neq 0$ , and  $MR(7) + MR(8) + MR(9) = 0$ ,  
or NPAR is inconsistent with the contents of MR,  
or WA is too small,  
or B is too small.

IFAIL = 2

A supplied model has parameter values which have failed the validity test.

IFAIL = 3

The supplied time series is too short to carry out the requested filtering successfully.

IFAIL = 4

This only occurs when an ARIMA model for the time series has been supplied. The matrix which is used to solve for the starting values for MA filtering is singular.

## 7 Accuracy

Accuracy and stability are high except when the AR-like parameters are close to the invertibility boundary. All calculations are performed in *basic precision* except for one inner product type calculation which on machines of low precision is performed in *additional precision*.

## 8 Further Comments

The time taken by the routine is roughly proportional to the product of the length of the series and number of parameters in the filtering model with appreciable increase if an ARIMA model is supplied for the time series.

## 9 Example

The example program reads a time series of length 296. It reads one univariate ARIMA (1,1,0,0,1,1,12) model for the series and the (0,13,12) transfer function filtering model. 12 initial backforecasts are required and these are calculated by a call to G13AJF. The backforecasts are inserted at the start of the series and G13BBF is called to perform the filtering.

### 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.

```

*      G13BBF Example Program Text
*      Mark 14 Revised.  NAG Copyright 1989.
*      .. Parameters ..
      INTEGER          NXMAX, NPMAX, ISTMAX, IFVMAX, IW, IQXDAX, NYMAX,
+                    NBMAX
      PARAMETER        (NXMAX=200, NPMAX=30, ISTMAX=30, IFVMAX=12, IW=2000,
+                    IQXDAX=15, NYMAX=NXMAX+IQXDAX, NBMAX=NYMAX+20)
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5, NOUT=6)
*      .. Local Scalars ..
      real            A1, A2, CX, CY, RMS
      INTEGER          I, IDD, IFAIL, II, IJ, IQXD, IWA, J, K, N, NB,
+                    NI, NMR, NPAR, NPARX, NST, NX, NY
*      .. Local Arrays ..
      real            B(NXMAX), FSD(IFVMAX), FVA(IFVMAX), PAR(NPMAX),
+                    PARX(NPMAX), ST(NPMAX), WA(IW), X(NXMAX),
+                    Y(NYMAX)
      INTEGER          ISF(4), MR(10), MRX(7)
*      .. External Subroutines ..
      EXTERNAL         G13AJF, G13BBF
*      .. Intrinsic Functions ..
      INTRINSIC        MAX, MIN, MOD
*      .. Executable Statements ..
      WRITE (NOUT,*) 'G13BBF Example Program Results'
*      Skip heading in data file
      READ (NIN,*)
      READ (NIN,*) NX
      WRITE (NOUT,*)
      IF (NX.GT.0 .AND. NX.LE.NXMAX) THEN
      READ (NIN,*) (X(I),I=1,NX)
*      Read univariate ARIMA for series
      READ (NIN,*) (MRX(I),I=1,7)
      READ (NIN,*) CX
      NPARX = MRX(1) + MRX(3) + MRX(4) + MRX(6)
      IF (NPARX.GT.0 .AND. NPARX.LE.NPMAX) THEN
      READ (NIN,*) (PARX(I),I=1,NPARX)
*      Read model by which to filter series
      READ (NIN,*) (MR(I),I=1,3)
      NPAR = MR(2) + MR(3) + 1
      IF (NPAR.GT.0 .AND. NPAR+NPARX.LE.NPMAX) THEN
      READ (NIN,*) (PAR(I),I=1,NPAR)
*      Initially backforecast QY values
*      (1) Reverse series in situ
      N = NX/2
      NI = NX
      DO 20 I = 1, N
          A1 = X(I)
          A2 = X(NI)
          X(I) = A2

```

```

                X(NI) = A1
                NI = NI - 1
20      CONTINUE
        IDD = MRX(2) + MRX(5)
*       (2) Possible sign reversal for ARIMA constant
        IF (MOD(IDD,2).NE.0) CX = -CX
*       (3) Calculate number of backforecasts required
        IQXD = MRX(3) + MRX(6)*MRX(7)
        IFAIL = 0
        IF (IQXD.NE.0) CALL G13AJF(MRX,PARX,NPARX,CX,0,X,NX,RMS,
+                               ST,ISTMAX,NST,IQXD,FVA,FSD,
+                               IFVMAX,ISF,WA,IW,IFAIL)
*       Move backforecasts to start of Y array
        J = IQXD
        DO 40 I = 1, IQXD
            Y(I) = FVA(J)
            J = J - 1
40      CONTINUE
*       Move series into Y
        J = IQXD + 1
        K = NX
        DO 60 I = 1, NX
            IF (J.GT.NYMAX) STOP
            Y(J) = X(K)
            J = J + 1
            K = K - 1
60      CONTINUE
        END IF
*       Calculate series length
        NY = NX + IQXD
*       Move ARIMA for series into MR
        DO 80 I = 1, 7
            MR(3+I) = MRX(I)
80      CONTINUE
*       Move parameters of ARIMA for Y into PAR
        DO 100 I = 1, NPARX
            PAR(NPAR+I) = PARX(I)
100     CONTINUE
        NPAR = NPAR + NPARX
*       Move constant and reset sign reversal
        CY = CX
        IF (MOD(IDD,2).NE.0) CY = -CY
*       Set parameters for call to filter routine G13BBF
        NMR = 10
        IWA = MR(3) + MR(4) + MR(5) + (MR(7)+MR(8))*MR(10)
        IWA = NPAR + IWA*(IWA+2)
        NB = NY + MAX(MR(1)+MR(2),MR(3))
        IF (IWA.LE.IW .AND. NB.LE.NBMAX) THEN
            IFAIL = 0
*
*       Filter series by call to G13BBF
        CALL G13BBF(Y,NY,MR,NMR,PAR,NPAR,CY,WA,IWA,B,NB,IFAIL)
*
        WRITE (NOUT,*)
+       '
+       ' Original          Filtered'
        WRITE (NOUT,*)
+       ' Backforecasts  y-series      series'
        IF (IQXD.NE.0) THEN
            IJ = -IQXD
            DO 120 I = 1, IQXD
                WRITE (NOUT,99999) IJ, Y(I), B(I)
                IJ = IJ + 1
120         CONTINUE
            WRITE (NOUT,*)
            WRITE (NOUT,*)
+       ' Filtered          Filtered          Filtered          Filtered'
            WRITE (NOUT,*)
+       ' series            series            series            series'
            DO 140 I = IQXD + 1, NY, 4
                WRITE (NOUT,99998) (II-IQXD,B(II),II=I,MIN(NY,I+3))
140         CONTINUE

```

```

                END IF
            END IF
        END IF
    END IF
STOP
*
99999 FORMAT (1X,I8,F17.1,F16.1)
99998 FORMAT (1X,I5,F10.1,I6,F10.1,I6,F10.1,I6,F10.1)
END

```

## 9.2 Program Data

G13BBF Example Program Data

```

158
5312. 5402. 4960. 4717. 4383. 3828. 3665. 3718.
3744. 3994. 4150. 4064. 4324. 4256. 3986. 3670.
3292. 2952. 2765. 2813. 2850. 3085. 3256. 3213.
3514. 3386. 3205. 3124. 2804. 2536. 2445. 2649.
2761. 3183. 3456. 3529. 4067. 4079. 4082. 4029.
3887. 3684. 3707. 3923. 4068. 4557. 4975. 5197.
6054. 6471. 6277. 5529. 5059. 4539. 4236. 4305.
4299. 4478. 4561. 4470. 4712. 4512. 4129. 3942.
3572. 3149. 3026. 3141. 3145. 3322. 3384. 3373.
3630. 3555. 3413. 3127. 2966. 2685. 2642. 2789.
2867. 3032. 3125. 3176. 3359. 3265. 3053. 2915.
2690. 2518. 2523. 2737. 3074. 3671. 4355. 4648.
5232. 5349. 5228. 5172. 4932. 4637. 4642. 4930.
5033. 5223. 5482. 5560. 5960. 5929. 5697. 5583.
5316. 5039. 4972. 5169. 5138. 5316. 5409. 5375.
5803. 5736. 5643. 5416. 5059. 4810. 4937. 5166.
5187. 5348. 5483. 5626. 6077. 6033. 5996. 5860.
5499. 5210. 5421. 5609. 5586. 3663. 5829. 6005.
6693. 6792. 6966. 7227. 7089. 6823. 7286. 7621.
7758. 8000. 8393. 8592. 9186. 9175.
    1      1      0      0      1      1      12
    0.000
    0.620  0.820
    0      13     12
    1.0131  0.0806 -0.0150 -0.0150 -0.0150 -0.0150
-0.0150 -0.0150 -0.0150 -0.0150 -0.0150 -0.0150
    0.9981 -0.0956  0.0000  0.0000  0.0000  0.0000
    0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
    0.0000  0.8200

```

## 9.3 Program Results

G13BBF Example Program Results

Backforecasts	Original y-series	Filtered series					
-12	5159.0	4549.2					
-11	5165.9	4550.9					
-10	4947.5	4552.8					
-9	4729.8	4554.9					
-8	4424.5	4557.4					
-7	4072.5	4560.7					
-6	3995.5	4565.0					
-5	4142.7	4571.1					
-4	4219.7	4580.0					
-3	4452.1	4593.5					
-2	4758.0	4614.3					
-1	4834.6	4647.1					
	Filtered series	Filtered series	Filtered series	Filtered series		Filtered series	
1	4699.2	2 4782.2	3 4552.8	4 4550.4			
5	4525.7	6 4324.8	7 4256.9	8 4169.7			
9	4127.9	10 4154.6	11 4011.3	12 3878.7			
13	3705.1	14 3619.1	15 3603.1	16 3496.1			
17	3422.6	18 3463.5	19 3349.8	20 3262.1			

21	3225.9	22	3218.1	23	3103.6	24	3023.5
25	2905.9	26	2758.5	27	2828.2	28	2958.4
29	2926.2	30	3019.8	31	3010.7	32	3082.8
33	3111.7	34	3286.3	35	3279.3	36	3324.4
37	3461.7	38	3468.3	39	3709.0	40	3839.6
41	4004.4	42	4146.3	43	4265.3	44	4344.6
45	4419.8	46	4647.2	47	4802.6	48	4999.5
49	5446.0	50	5861.0	51	5855.9	52	5310.7
53	5202.5	54	5046.6	55	4857.1	56	4812.3
57	4740.7	58	4631.1	59	4447.5	60	4317.7
61	4079.8	62	3833.7	63	3667.7	64	3774.8
65	3709.9	66	3648.5	67	3645.3	68	3619.8
69	3549.4	70	3439.2	71	3250.3	72	3209.2
73	3005.2	74	2912.4	75	2994.1	76	2947.9
77	3103.7	78	3168.1	79	3226.0	80	3224.1
81	3233.0	82	3119.2	83	2992.5	84	3014.8
85	2763.7	86	2671.3	87	2664.9	88	2778.2
89	2823.8	90	2989.0	91	3072.2	92	3132.1
93	3394.6	94	3717.4	95	4180.5	96	4405.9
97	4605.2	98	4733.0	99	4830.9	100	5030.8
101	5079.0	102	5125.0	103	5236.7	104	5392.7
105	5396.7	106	5300.7	107	5312.1	108	5336.6
109	5347.9	110	5331.2	111	5322.0	112	5444.8
113	5468.7	114	5532.9	115	5555.9	116	5603.4
117	5483.2	118	5406.8	119	5250.5	120	5171.9
121	5217.4	122	5162.3	123	5296.1	124	5268.2
125	5204.9	126	5290.7	127	5500.0	128	5552.3
129	5503.3	130	5419.2	131	5335.6	132	5447.6
133	5495.1	134	5475.1	135	5643.8	136	5713.1
137	5655.1	138	5691.9	139	5958.4	140	5959.0
141	5884.8	142	3714.7	143	5877.8	144	5814.1
145	6095.6	146	6210.7	147	6560.5	148	7013.9
149	7174.8	150	7230.8	151	7726.7	152	7880.0
153	7997.4	154	8428.5	155	8264.1	156	8443.1
157	8615.4	158	8644.6				

---