# NAG Library Routine Document

# F02GCF

**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

F02GCF computes selected eigenvalues and eigenvectors of a complex general matrix.

## 2 Specification

```
SUBROUTINE F02GCF (CRIT, N, A, LDA, WL, WU, MEST, M, W, V, LDV, WORK,           &
                   LWORK, RWORK, IWORK, BWORK, IFAIL)

INTEGER                 N, LDA, MEST, M, LDV, LWORK, IWORK(N), IFAIL
REAL (KIND=nag_wp)      WL, WU, RWORK(2*N)
COMPLEX (KIND=nag_wp)   A(LDA,*), W(N), V(LDV,MEST), WORK(LWORK)
LOGICAL                 BWORK(N)
CHARACTER(1)            CRIT
```

## 3 Description

F02GCF computes selected eigenvalues and the corresponding right eigenvectors of a complex general matrix $A$:

$$Ax_i = \lambda_i x_i.$$

Eigenvalues $\lambda_i$ may be selected either by *modulus*, satisfying

$$w_l \leq |\lambda_i| \leq w_u,$$

or by *real part*, satisfying

$$w_l \leq \mathrm{Re}(\lambda_i) \leq w_u.$$

## 4 References

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

## 5 Parameters

1: CRIT – CHARACTER(1) *Input*

*On entry*: indicates the criterion for selecting eigenvalues.

CRIT = 'M'
: Eigenvalues are selected according to their moduli: $w_l \leq |\lambda_i| \leq w_u$.

CRIT = 'R'
: Eigenvalues are selected according to their real parts: $w_l \leq \mathrm{Re}(\lambda_i) \leq w_u$.

*Constraint*: CRIT = 'M' or 'R'.

2: N – INTEGER *Input*

*On entry*: $n$, the order of the matrix $A$.

*Constraint*: $\mathrm{N} \geq 0$.

3: A(LDA,$*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

**Note**: the second dimension of the array A must be at least $\max(1, \mathrm{N})$.

*On entry*: the $n$ by $n$ general matrix $A$.

*On exit*: contains the Hessenberg form of the balanced input matrix $A'$ (see Section 8).

4: LDA – INTEGER *Input*

*On entry*: the first dimension of the array A as declared in the (sub)program from which F02GCF is called.

*Constraint*: $\mathrm{LDA} \geq \max(1, \mathrm{N})$.

5: WL – REAL (KIND=nag_wp) *Input*
6: WU – REAL (KIND=nag_wp) *Input*

*On entry*: $w_l$ and $w_u$, the lower and upper bounds on the criterion for the selected eigenvalues (see CRIT).

*Constraint*: $\mathrm{WU} > \mathrm{WL}$.

7: MEST – INTEGER *Input*

*On entry*: the second dimension of the array V as declared in the (sub)program from which F02GCF is called. MEST must be an upper bound on $m$, the number of eigenvalues and eigenvectors selected. No eigenvectors are computed if $\mathrm{MEST} < m$.

*Constraint*: $\mathrm{MEST} \geq \max(1, m)$.

8: M – INTEGER *Output*

*On exit*: $m$, the number of eigenvalues actually selected.

9: W(N) – COMPLEX (KIND=nag_wp) array *Output*

*On exit*: the first M elements of W hold the selected eigenvalues; elements $\mathrm{M} + 1$ to N contain the other eigenvalues.

10: V(LDV,MEST) – COMPLEX (KIND=nag_wp) array *Output*

*On exit*: contains the selected eigenvectors, with the $i$th column holding the eigenvector associated with the eigenvalue $\lambda_i$ (stored in $\mathrm{W}(i)$).

11: LDV – INTEGER *Input*

*On entry*: the first dimension of the array V as declared in the (sub)program from which F02GCF is called.

*Constraint*: $\mathrm{LDV} \geq \max(1, \mathrm{N})$.

12: WORK(LWORK) – COMPLEX (KIND=nag_wp) array *Workspace*
13: LWORK – INTEGER *Input*

*On entry*: the dimension of the array WORK as declared in the (sub)program from which F02GCF is called.

*Constraint*: $\mathrm{LWORK} \geq \max(1, \mathrm{N} \times (\mathrm{N} + 2))$.

14: RWORK($2 \times N$) – REAL (KIND=nag_wp) array *Workspace*

15: IWORK(N) – INTEGER array *Workspace*

16: BWORK(N) – LOGICAL array *Workspace*

17: 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.

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.**

*On exit*: IFAIL $= 0$ unless the routine detects an error or a warning has been flagged (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 or warnings detected by the routine:

IFAIL $= 1$

On entry, CRIT $\neq$ 'M' or 'R',
or N $< 0$,
or LDA $< \max(1, N)$,
or WU $\leq$ WL,
or MEST $< 1$,
or LDV $< \max(1, N)$,
or LWORK $< \max(1, N \times (N + 2))$.

IFAIL $= 2$

The $QR$ algorithm failed to compute all the eigenvalues. No eigenvectors have been computed.

IFAIL $= 3$

There are more than MEST eigenvalues in the specified range. The actual number of eigenvalues in the range is returned in M. No eigenvectors have been computed. Rerun with the second dimension of V $=$ MEST $\geq$ M.

IFAIL $= 4$

Inverse iteration failed to compute all the specified eigenvectors. If an eigenvector failed to converge, the corresponding column of V is set to zero.

# 7 Accuracy

If $\lambda_i$ is an exact eigenvalue, and $\tilde{\lambda}_i$ is the corresponding computed value, then

$$\left|\tilde{\lambda}_i - \lambda_i\right| \leq \frac{c(n)\epsilon\left\|A'\right\|_2}{s_i},$$

where $c(n)$ is a modestly increasing function of $n$, $\epsilon$ is the ***machine precision***, and $s_i$ is the reciprocal condition number of $\lambda_i$; $A'$ is the balanced form of the original matrix $A$ (see Section 8), and $\left\|A'\right\| \leq \|A\|$.

If $x_i$ is the corresponding exact eigenvector, and $\tilde{x}_i$ is the corresponding computed eigenvector, then the angle $\theta(\tilde{x}_i, x_i)$ between them is bounded as follows:

$$\theta(\tilde{x}_i, x_i) \leq \frac{c(n)\epsilon\|A'\|_2}{sep_i}$$

where $sep_i$ is the reciprocal condition number of $x_i$.

The condition numbers $s_i$ and $sep_i$ may be computed from the Hessenberg form of the balanced matrix $A'$ which is returned in the array A. This requires calling F08PSF (ZHSEQR) with JOB = 'S' to compute the Schur form of $A'$, followed by F08QYF (ZTRSNA).

# 8 Further Comments

F02GCF calls routines from LAPACK in Chapter F08. It first balances the matrix, using a diagonal similarity transformation to reduce its norm; and then reduces the balanced matrix $A'$ to upper Hessenberg form $H$, using a unitary similarity transformation: $A' = QHQ^{\mathrm{H}}$. The routine uses the Hessenberg $QR$ algorithm to compute all the eigenvalues of $H$, which are the same as the eigenvalues of $A$. It computes the eigenvectors of $H$ which correspond to the selected eigenvalues, using inverse iteration. It premultiplies the eigenvectors by $Q$ to form the eigenvectors of $A'$; and finally transforms the eigenvectors to those of the original matrix $A$.

Each eigenvector $x$ is normalized so that $\|x\|_2 = 1$, and the element of largest absolute value is real and positive.

The inverse iteration routine may make a small perturbation to the real parts of close eigenvalues, and this may shift their moduli just outside the specified bounds. If you are relying on eigenvalues being within the bounds, you should test them on return from F02GCF.

The time taken by the routine is approximately proportional to $n^3$.

The routine can be used to compute *all* eigenvalues and eigenvectors, by setting WL large and negative, and WU large and positive.

# 9 Example

This example computes those eigenvalues of the matrix $A$ which lie in the range $[-5.5, +5.5]$, and their corresponding eigenvectors, where

$$A = \begin{pmatrix} -3.97 - 5.04i & -4.11 + 3.70i & -0.34 + 1.01i & 1.29 - 0.86i \\ 0.34 - 1.50i & 1.52 - 0.43i & 1.88 - 5.38i & 3.36 + 0.65i \\ 3.31 - 3.85i & 2.50 + 3.45i & 0.88 - 1.08i & 0.64 - 1.48i \\ -1.10 + 0.82i & 1.81 - 1.59i & 3.25 + 1.33i & 1.57 - 3.44i \end{pmatrix}.$$

## 9.1 Program Text

```
    Program f02gcfe

!     F02GCF Example Program Text

!     Mark 24 Release. NAG Copyright 2012.

!     .. Use Statements ..
      Use nag_library, Only: f02gcf, nag_wp, x04dbf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: wl, wu
      Integer                          :: i, ifail, lda, ldv, lwork, m, mest, n
!     .. Local Arrays ..
      Complex (Kind=nag_wp), Allocatable :: a(:,:), v(:,:), w(:), work(:)
      Real (Kind=nag_wp), Allocatable  :: rwork(:)
```

```
      Integer, Allocatable              :: iwork(:)
      Logical, Allocatable              :: bwork(:)
      Character (1)                     :: clabs(1), rlabs(1)
!     .. Executable Statements ..
      Write (nout,*) 'F02GCF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, mest, wl, wu
      lda = n
      ldv = n
      lwork = 64*n
      Allocate (a(lda,n),v(ldv,n),w(n),work(lwork),rwork(2*n),iwork(n), &
        bwork(n))
!     Read A from data file
      Read (nin,*)(a(i,1:n),i=1,n)

!     Compute selected eigenvalues and eigenvectors of A

!     ifail: behaviour on error exit
!             =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0
      Call f02gcf('Moduli',n,a,lda,wl,wu,mest,m,w,v,ldv,work,lwork,rwork, &
        iwork,bwork,ifail)

      Write (nout,*)
      Write (nout,*) 'Eigenvalues'
      Write (nout,99999) w(1:m)
      Write (nout,*)
      Flush (nout)
      ifail = 0
      Call x04dbf('General',' ',n,m,v,ldv,'Bracketed','F7.4','Eigenvectors', &
        'Integer',rlabs,'Integer',clabs,80,0,ifail)

99999 Format ((3X,4(' (',F7.4,',',F7.4,')':)))
    End Program f02gcfe
```

## 9.2 Program Data

```
F02GCF Example Program Data
  4  3  -5.5  5.5                                      : n, mest, wl, wu
 (-3.97,-5.04) (-4.11, 3.70) (-0.34, 1.01) ( 1.29,-0.86)
 ( 0.34,-1.50) ( 1.52,-0.43) ( 1.88,-5.38) ( 3.36, 0.65)
 ( 3.31,-3.85) ( 2.50, 3.45) ( 0.88,-1.08) ( 0.64,-1.48)
 (-1.10, 0.82) ( 1.81,-1.59) ( 3.25, 1.33) ( 1.57,-3.44)   : matrix A
```

## 9.3 Program Results

```
 F02GCF Example Program Results

 Eigenvalues
    (-5.0000, 2.0060) ( 3.0023,-3.9998)

 Eigenvectors
                    1                 2
 1  (-0.3865, 0.1732) (-0.0356,-0.1782)
 2  (-0.3539, 0.4529) ( 0.1264, 0.2666)
 3  ( 0.6124, 0.0000) ( 0.0129,-0.2966)
 4  (-0.0859,-0.3284) ( 0.8898, 0.0000)
```