NAG Library Routine Document

F08TSF (ZHPGST)

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

F08TSF (ZHPGST) reduces a complex Hermitian-definite generalized eigenproblem $Az = \lambda Bz$, $ABz = \lambda z$ or $BAz = \lambda z$ to the standard form $Cy = \lambda y$, where A is a complex Hermitian matrix and B has been factorized by F07GRF (ZPPTRF), using packed storage.

2   Specification

```
SUBROUTINE F08TSF (ITYPE, UPLO, N, AP, BP, INFO)

INTEGER            ITYPE, N, INFO
COMPLEX (KIND=nag_wp) AP(*), BP(*)
CHARACTER(1)       UPLO
```

The routine may be called by its LAPACK name ***zhpgst***.

3   Description

To reduce the complex Hermitian-definite generalized eigenproblem $Az = \lambda Bz$, $ABz = \lambda z$ or $BAz = \lambda z$ to the standard form $Cy = \lambda y$ using packed storage, F08TSF (ZHPGST) must be preceded by a call to F07GRF (ZPPTRF) which computes the Cholesky factorization of B; B must be positive definite.

The different problem types are specified by the parameter ITYPE, as indicated in the table below. The table shows how C is computed by the routine, and also how the eigenvectors z of the original problem can be recovered from the eigenvectors of the standard form.

ITYPE	Problem	UPLO	B	C	z
1	$Az = \lambda Bz$	'U' 'L'	$U^{\mathrm{H}}U$ LL^{H}	$U^{-\mathrm{H}}AU^{-1}$ $L^{-1}AL^{-\mathrm{H}}$	$U^{-1}y$ $L^{-\mathrm{H}}y$
2	$ABz = \lambda z$	'U' 'L'	$U^{\mathrm{H}}U$ LL^{H}	UAU^{H} $L^{\mathrm{H}}AL$	$U^{-1}y$ $L^{-\mathrm{H}}y$
3	$BAz = \lambda z$	'U' 'L'	$U^{\mathrm{H}}U$ LL^{H}	UAU^{H} $L^{\mathrm{H}}AL$	$U^{\mathrm{H}}y$ Ly

4   References

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

5   Parameters

1:    ITYPE – INTEGER   *Input*

On entry: indicates how the standard form is computed.

ITYPE = 1

if UPLO = 'U', $C = U^{-H}AU^{-1}$;

if UPLO = 'L', $C = L^{-1}AL^{-H}$.

ITYPE = 2 or 3

if UPLO = 'U', $C = UAU^{H}$;

if UPLO = 'L', $C = L^{H}AL$.

Constraint: ITYPE = 1, 2 or 3.

2:    UPLO – CHARACTER(1)   *Input*

On entry: indicates whether the upper or lower triangular part of A is stored and how B has been factorized.

UPLO = 'U'
     The upper triangular part of A is stored and $B = U^{H}U$.

UPLO = 'L'
     The lower triangular part of A is stored and $B = LL^{H}$.

Constraint: UPLO = 'U' or 'L'.

3:    N – INTEGER   *Input*

On entry: n, the order of the matrices A and B.

Constraint: N $\geq$ 0.

4:    AP($*$) – COMPLEX (KIND=nag_wp) array   *Input/Output*

Note: the dimension of the array AP must be at least $\max(1, N \times (N+1)/2)$.

On entry: the upper or lower triangle of the n by n Hermitian matrix A, packed by columns.

More precisely,

if UPLO = 'U', the upper triangle of A must be stored with element A_{ij} in AP$(i + j(j-1)/2)$ for $i \leq j$;

if UPLO = 'L', the lower triangle of A must be stored with element A_{ij} in AP$(i + (2n-j)(j-1)/2)$ for $i \geq j$.

On exit: the upper or lower triangle of AP is overwritten by the corresponding upper or lower triangle of C as specified by ITYPE and UPLO, using the same packed storage format as described above.

5:    BP($*$) – COMPLEX (KIND=nag_wp) array   *Input*

Note: the dimension of the array BP must be at least $\max(1, N \times (N+1)/2)$.

On entry: the Cholesky factor of B as specified by UPLO and returned by F07GRF (ZPPTRF).

6:    INFO – INTEGER   *Output*

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

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFO < 0

If $\text{INFO} = -i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

7 Accuracy

Forming the reduced matrix C is a stable procedure. However it involves implicit multiplication by B^{-1} if (ITYPE = 1) or B (if ITYPE = 2 or 3). When F08TSF (ZHPGST) is used as a step in the computation of eigenvalues and eigenvectors of the original problem, there may be a significant loss of accuracy if B is ill-conditioned with respect to inversion. See the document for F08SNF (ZHEGV) for further details.

8 Further Comments

The total number of real floating point operations is approximately $4n^3$.

The real analogue of this routine is F08TEF (DSPGST).

9 Example

This example computes all the eigenvalues of $Az = \lambda Bz$, where

$$
A = \begin{pmatrix}
-7.36 + 0.00i & 0.77 - 0.43i & -0.64 - 0.92i & 3.01 - 6.97i \\
0.77 + 0.43i & 3.49 + 0.00i & 2.19 + 4.45i & 1.90 + 3.73i \\
-0.64 + 0.92i & 2.19 - 4.45i & 0.12 + 0.00i & 2.88 - 3.17i \\
3.01 + 6.97i & 1.90 - 3.73i & 2.88 + 3.17i & -2.54 + 0.00i
\end{pmatrix}
$$

and

$$
B = \begin{pmatrix}
3.23 + 0.00i & 1.51 - 1.92i & 1.90 + 0.84i & 0.42 + 2.50i \\
1.51 + 1.92i & 3.58 + 0.00i & -0.23 + 1.11i & -1.18 + 1.37i \\
1.90 - 0.84i & -0.23 - 1.11i & 4.09 + 0.00i & 2.33 - 0.14i \\
0.42 - 2.50i & -1.18 - 1.37i & 2.33 + 0.14i & 4.29 + 0.00i
\end{pmatrix},
$$

using packed storage. Here B is Hermitian positive definite and must first be factorized by F07GRF (ZPPTRF). The program calls F08TSF (ZHPGST) to reduce the problem to the standard form $Cy = \lambda y$; then F08GSF (ZHPTRD) to reduce C to tridiagonal form, and F08JFF (DSTERF) to compute the eigenvalues.

9.1 Program Text

```
    Program f08tsfe

!       F08TSF Example Program Text

!       Mark 24 Release. NAG Copyright 2012.

!       .. Use Statements ..
        Use nag_library, Only: dsterf, nag_wp, zhpgst, zhptrd, zpptrf
!       .. Implicit None Statement ..
        Implicit None
!       .. Parameters ..
        Integer, Parameter              :: nin = 5, nout = 6
!       .. Local Scalars ..
        Integer                         :: i, info, j, n
        Character (1)                   :: uplo
!       .. Local Arrays ..
        Complex (Kind=nag_wp), Allocatable :: ap(:), bp(:), tau(:)
        Real (Kind=nag_wp), Allocatable  :: d(:), e(:)
```

```
!     .. Executable Statements ..
      Write (nout,*) 'F08TSF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n

      Allocate (ap(n*(n+1)/2),bp(n*(n+1)/2),tau(n),d(n),e(n-1))

!     Read A and B from data file

      Read (nin,*) uplo
      If (uplo=='U') Then
        Read (nin,*)((ap(i+j*(j-1)/2),j=i,n),i=1,n)
        Read (nin,*)((bp(i+j*(j-1)/2),j=i,n),i=1,n)
      Else If (uplo=='L') Then
        Read (nin,*)((ap(i+(2*n-j)*(j-1)/2),j=1,i),i=1,n)
        Read (nin,*)((bp(i+(2*n-j)*(j-1)/2),j=1,i),i=1,n)
      End If

!     Compute the Cholesky factorization of B
!     The NAG name equivalent of zpptrf is f07grf
      Call zpptrf(uplo,n,bp,info)

      Write (nout,*)
      If (info>0) Then
        Write (nout,*) 'B is not positive definite.'
      Else

!       Reduce the problem to standard form C*y = lambda*y, storing
!       the result in A
!       The NAG name equivalent of zhpgst is f08tsf
        Call zhpgst(1,uplo,n,ap,bp,info)

!       Reduce C to tridiagonal form T = (Q**H)*C*Q
!       The NAG name equivalent of zhptrd is f08gsf
        Call zhptrd(uplo,n,ap,d,e,tau,info)

!       Calculate the eigenvalues of T (same as C)
!       The NAG name equivalent of dsterf is f08jff
        Call dsterf(n,d,e,info)

        If (info>0) Then
          Write (nout,*) 'Failure to converge.'
        Else

!         Print eigenvalues

          Write (nout,*) 'Eigenvalues'
          Write (nout,99999) d(1:n)
        End If
      End If

99999 Format (3X,(9F8.4))
    End Program f08tsfe
```

9.2   Program Data

```
F08TSF Example Program Data
  4                                                         :Value of N
  'L'                                                       :Value of UPLO
 (-7.36, 0.00)
 ( 0.77, 0.43) ( 3.49, 0.00)
 (-0.64, 0.92) ( 2.19,-4.45) ( 0.12, 0.00)
 ( 3.01, 6.97) ( 1.90,-3.73) ( 2.88, 3.17) (-2.54, 0.00)  :End of matrix A
 ( 3.23, 0.00)
 ( 1.51, 1.92) ( 3.58, 0.00)
 ( 1.90,-0.84) (-0.23,-1.11) ( 4.09, 0.00)
 ( 0.42,-2.50) (-1.18,-1.37) ( 2.33, 0.14) ( 4.29, 0.00)  :End of matrix B
```

9.3   Program Results

```
F08TSF Example Program Results

Eigenvalues
    -5.9990 -2.9936   0.5047   3.9990
```

_____