NAG Library Routine Document

F08WVF (ZGGBAL)

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

F08WVF (ZGGBAL) balances a pair of complex square matrices (A, B) of order n. Balancing usually improves the accuracy of computed generalized eigenvalues and eigenvectors.

2 Specification

```
SUBROUTINE F08WVF (JOB, N, A, LDA, B, LDB, ILO, IHI, LSCALE, RSCALE,      &
                   WORK, INFO)

INTEGER                 N, LDA, LDB, ILO, IHI, INFO
REAL (KIND=nag_wp)      LSCALE(N), RSCALE(N), WORK(6*N)
COMPLEX (KIND=nag_wp) A(LDA,*), B(LDB,*)
CHARACTER(1)            JOB
```

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

3 Description

Balancing may reduce the 1-norm of the matrices and improve the accuracy of the computed eigenvalues and eigenvectors in the complex generalized eigenvalue problem

$$Ax = \lambda Bx.$$

F08WVF (ZGGBAL) is usually the first step in the solution of the above generalized eigenvalue problem. Balancing is optional but it is highly recommended.

The term 'balancing' covers two steps, each of which involves similarity transformations on A and B. The routine can perform either or both of these steps. Both steps are optional.

1. The routine first attempts to permute A and B to block upper triangular form by a similarity transformation:

$$PAP^{\mathrm{T}} = F = \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ & F_{22} & F_{23} \\ & & F_{33} \end{pmatrix}$$

$$PBP^{\mathrm{T}} = G = \begin{pmatrix} G_{11} & G_{12} & G_{13} \\ & G_{22} & G_{23} \\ & & G_{33} \end{pmatrix}$$

   where P is a permutation matrix, F_{11}, F_{33}, G_{11} and G_{33} are upper triangular. Then the diagonal elements of the matrix pairs (F_{11}, G_{11}) and (F_{33}, G_{33}) are generalized eigenvalues of (A, B). The rest of the generalized eigenvalues are given by the matrix pair (F_{22}, G_{22}) which are in rows and columns i_{lo} to i_{hi}. Subsequent operations to compute the generalized eigenvalues of (A, B) need only be applied to the matrix pair (F_{22}, G_{22}); this can save a significant amount of work if $i_{\mathrm{lo}} > 1$ and $i_{\mathrm{hi}} < n$. If no suitable permutation exists (as is often the case), the routine sets $i_{\mathrm{lo}} = 1$ and $i_{\mathrm{hi}} = n$.

2. The routine applies a diagonal similarity transformation to (F, G), to make the rows and columns of (F_{22}, G_{22}) as close in norm as possible:

$$DF\hat{D} = \begin{pmatrix} I & 0 & 0 \\ 0 & D_{22} & 0 \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ & F_{22} & F_{23} \\ & & F_{33} \end{pmatrix} \begin{pmatrix} I & 0 & 0 \\ 0 & \hat{D}_{22} & 0 \\ 0 & 0 & I \end{pmatrix}$$

$$DGD^{-1} = \begin{pmatrix} I & 0 & 0 \\ 0 & D_{22} & 0 \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} G_{11} & G_{12} & G_{13} \\ & G_{22} & G_{23} \\ & & G_{33} \end{pmatrix} \begin{pmatrix} I & 0 & 0 \\ 0 & \hat{D}_{22} & 0 \\ 0 & 0 & I \end{pmatrix}$$

This transformation usually improves the accuracy of computed generalized eigenvalues and eigenvectors.

4 References

Ward R C (1981) Balancing the generalized eigenvalue problem *SIAM J. Sci. Stat. Comp.* **2** 141–152

5 Parameters

1: JOB – CHARACTER(1) *Input*

On entry: specifies the operations to be performed on matrices A and B.

JOB = 'N'
: No balancing is done. Initialize ILO $= 1$, IHI $=$ N, LSCALE$(i) = 1.0$ and RSCALE$(i) = 1.0$, for $i = 1, 2, \ldots, n$.

JOB = 'P'
: Only permutations are used in balancing.

JOB = 'S'
: Only scalings are are used in balancing.

JOB = 'B'
: Both permutations and scalings are used in balancing.

Constraint: JOB = 'N', 'P', 'S' or 'B'.

2: N – INTEGER *Input*

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

Constraint: N ≥ 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 matrix A.

On exit: A is overwritten by the balanced matrix. If JOB = 'N', A is not referenced.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F08WVF (ZGGBAL) is called.

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

5: B(LDB, $*$) – COMPLEX (KIND=nag_wp) array *Input/Output*

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

On entry: the n by n matrix B.

On exit: B is overwritten by the balanced matrix. If JOB = 'N', B is not referenced.

6: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which F08WVF (ZGGBAL) is called.

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

7: ILO – INTEGER *Output*

8: IHI – INTEGER *Output*

On exit: i_{lo} and i_{hi} are set such that $\mathrm{A}(i,j) = 0$ and $\mathrm{B}(i,j) = 0$ if $i > j$ and $1 \leq j < i_{\mathrm{lo}}$ or $i_{\mathrm{hi}} < i \leq n$.

If JOB = 'N' or 'S', $i_{\mathrm{lo}} = 1$ and $i_{\mathrm{hi}} = n$.

9: LSCALE(N) – REAL (KIND=nag_wp) array *Output*

On exit: details of the permutations and scaling factors applied to the left side of the matrices A and B. If P_i is the index of the row interchanged with row i and d_i is the scaling factor applied to row i, then

$\mathrm{LSCALE}(i) = P_i$, for $i = 1, 2, \ldots, i_{\mathrm{lo}} - 1$;

$\mathrm{LSCALE}(i) = d_i$, for $i = i_{\mathrm{lo}}, \ldots, i_{\mathrm{hi}}$;

$\mathrm{LSCALE}(i) = P_i$, for $i = i_{\mathrm{hi}} + 1, \ldots, n$.

The order in which the interchanges are made is n to $i_{\mathrm{hi}} + 1$, then 1 to $i_{\mathrm{lo}} - 1$.

10: RSCALE(N) – REAL (KIND=nag_wp) array *Output*

On exit: details of the permutations and scaling factors applied to the right side of the matrices A and B.

If P_j is the index of the column interchanged with column j and $\hat{d}_j$ is the scaling factor applied to column j, then

$\mathrm{RSCALE}(j) = P_j$, for $j = 1, 2, \ldots, i_{\mathrm{lo}} - 1$;

$\mathrm{RSCALE}(j) = \hat{d}_j$, for $j = i_{\mathrm{lo}}, \ldots, i_{\mathrm{hi}}$;

$\mathrm{RSCALE}(j) = P_j$, for $j = i_{\mathrm{hi}} + 1, \ldots, n$.

The order in which the interchanges are made is n to $i_{\mathrm{hi}} + 1$, then 1 to $i_{\mathrm{lo}} - 1$.

11: WORK($6 \times \mathrm{N}$) – REAL (KIND=nag_wp) array *Workspace*

12: INFO – INTEGER *Output*

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

6 Error Indicators and Warnings

INFO < 0

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

7 Accuracy

The errors are negligible, compared to those in subsequent computations.

8 Parallelism and Performance

F08WVF (ZGGBAL) is not threaded by NAG in any implementation.

F08WVF (ZGGBAL) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

F08WVF (ZGGBAL) is usually the first step in computing the complex generalized eigenvalue problem but it is an optional step. The matrix B is reduced to the triangular form using the QR factorization routine F08ASF (ZGEQRF) and the unitary transformation Q is applied to the matrix A by calling F08AUF (ZUNMQR). This is followed by F08WSF (ZGGHRD) which reduces the matrix pair into the generalized Hessenberg form.

If the matrix pair (A, B) is balanced by this routine, then any generalized eigenvectors computed subsequently are eigenvectors of the balanced matrix pair. In that case, to compute the generalized eigenvectors of the original matrix, F08WWF (ZGGBAK) must be called.

The total number of floating-point operations is approximately proportional to n^2.

The real analogue of this routine is F08WHF (DGGBAL).

10 Example

See Section 10 in F08XSF (ZHGEQZ) and F08YXF (ZTGEVC).