NAG Toolbox for MATLAB

f08jy

1 Purpose

f08jy computes all the eigenvalues and, optionally, all the eigenvectors of a real n by n symmetric tridiagonal matrix.

2 Syntax

```
[d, e, m, w, z, isuppz, info] = f08jy(jobz, range, d, e, vl, vu, il, iu,
abstol, 'n', n)
```

3 Description

f08jy computes all the eigenvalues, and optionally the eigenvectors, of a real symmetric tridiagonal matrix T. That is, the function computes the spectral factorization of T given by

$$T = Z\Lambda Z^{\mathrm{T}},$$

where Λ is a diagonal matrix whose diagonal elements are the eigenvalues, λ_i, of T and Z is an orthogonal matrix whose columns are the eigenvectors, z_i, of T. Thus

$$Tz_i = \lambda_i z_i, \qquad i = 1, 2, \ldots, n.$$

This function uses the dqds and the **Relatively Robust Representation** algorithms to compute the eigenvalues and eigenvectors respectively; see for example Parlett and Dhillon 2000 and Dhillon and Parlett 2004 for further details. f08jy can usually compute all the eigenvalues and eigenvectors in $O\left(n^2\right)$ floating point operations and so, for large matrices, is often considerably faster than the other symmetric tridiagonal functions in this chapter when all the eigenvectors are required, particularly so compared to those functions that are based on the QR algorithm.

In the future this function may be extended to allow for the computation of selected eigenvalues and eigenvectors.

4 References

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J J, Du Croz J J, Greenbaum A, Hammarling S, McKenney A and Sorensen D 1999 *LAPACK Users' Guide* (3rd Edition) SIAM, Philadelphia URL: http://www.netlib.org/lapack/lug

Barlow J and Demmel J W 1990 Computing accurate eigensystems of scaled diagonally dominant matrices *SIAM J. Numer. Anal.* **27** 762–791

Dhillon I S and Parlett B N 2004 Orthogonal eigenvectors and relative gaps. *SIAM J. Appl. Math.* **25** 858–899

Parlett B N and Dhillon I S 2000 Relatively robust representations of symmetric tridiagonals *Linear Algebra Appl.* **309** 121–151

5 Parameters

5.1 Compulsory Input Parameters

1: **jobz – string**

Indicates whether eigenvectors are computed.

jobz = 'N'

Only eigenvalues are computed.

jobz = 'V'

Eigenvalues and eigenvectors are computed.

Constraint: **jobz** = 'N' or 'V'.

2: **range – string**

Indicates which eigenvalues should be returned.

range = 'A'

All eigenvalues will be found.

range = 'V'

All eigenvalues in the half-open interval $(\mathbf{vl}, \mathbf{vu}]$ will be found.

range = 'I'

The **il**th through **iu**th eigenvectors will be found.

Constraint: **range** = 'A', 'V' or 'I'.

3: **d**$(*)$ **– double array**

Note: the dimension of the array **d** must be at least $\max(1, \mathbf{n})$.

The n diagonal elements of the tridiagonal matrix T.

4: **e**$(*)$ **– double array**

Note: the dimension of the array **e** must be at least $\max(1, \mathbf{n} - 1)$.

The $(n - 1)$ subdiagonal elements of the tridiagonal matrix T.

5: **vl – double scalar**

6: **vu – double scalar**

If **range** = 'V', **vl** and **vu** contain the lower and upper bounds respectively of the interval to be searched for eigenvalues. If **range** = 'A' or 'I', **vl** and **vl** ar not referenced.

Constraint: **vl** $<$ **vu**.

7: **il – int32 scalar**

8: **iu – int32 scalar**

If **range** = 'I', **il** and **iu** contains the indices (in ascending order) of the smallest and largest eigenvalues to be returned, respectively. If **range** = 'A' or 'V', **il** and **iu** are not referenced.

Constraints:

if $\mathbf{n} > 0$, $1 \leq \mathbf{il} \leq \mathbf{iu} \leq \mathbf{n}$;
if $\mathbf{n} = 0$, $\mathbf{il} = 1$ and $\mathbf{iu} = 0$.

9: **abstol – double scalar**

In earlier versions, this argument was the absolute error tolerance for the eigenvalues/eigenvectors. It is now deprecated, and only included for backwards-compatibility.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The dimension of the array **d**.

n, the order of the matrix T.

Constraint: $\mathbf{n} \geq 0$.

5.3 Input Parameters Omitted from the MATLAB Interface

ldz, work, lwork, iwork, liwork

5.4 Output Parameters

1: **d**$(*)$ **– double array**

Note: the dimension of the array **d** must be at least $\max(1, \mathbf{n})$.

d is overwritten.

2: **e**$(*)$ **– double array**

Note: the dimension of the array **e** must be at least $\max(1, \mathbf{n} - 1)$.

e is overwritten.

3: **m – int32 scalar**

The total number of eigenvalues found.

If **range** = 'A', **m** = **n**.

4: **w**$(*)$ **– double array**

Note: the dimension of the array **w** must be at least $\max(1, \mathbf{n})$.

The eigenvalues in ascending order.

5: **z**(**ldz**,$*$) **– complex array**

The first dimension, **ldz**, of the array **z** must satisfy

if **jobz** = 'V', **ldz** $\geq \max(1, \mathbf{n})$;
ldz ≥ 1 otherwise.

The second dimension of the array must be at least $\max(1, \mathbf{n})$

If **jobz** = 'V', then if **info** = 0, the columns of **z** contain the orthonormal eigenvectors of the matrix T, with the ith column of Z holding the eigenvector associated with **w**(i).

If **jobz** = 'N', **z** is not referenced.

6: **isuppz**$(*)$ **– int32 array**

Note: the dimension of the array **isuppz** must be at least $\max(1, 2 \times \mathbf{n})$.

The support of the eigenvectors in Z, i.e., the indices indicating the nonzero elements in Z. The ith eigenvector is nonzero only in elements **isuppz**$(2 \times i - 1)$ through **isuppz**$(2 \times i)$.

7: **info – int32 scalar**

info = 0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

info $= -i$

If **info** $= -i$, parameter i had an illegal value on entry. The parameters are numbered as follows:

1: **jobz**, 2: **range**, 3: **n**, 4: **d**, 5: **e**, 6: **vl**, 7: **vu**, 8: **il**, 9: **iu**, 10: **abstol**, 11: **m**, 12: **w**, 13: **z**, 14: **ldz**, 15: **isuppz**, 16: **work**, 17: **lwork**, 18: **iwork**, 19: **liwork**, 20: **info**.

It is possible that **info** refers to a parameter that is omitted from the MATLAB interface. This usually indicates that an error in one of the other input parameters has caused an incorrect value to be inferred.

info > 0

If **info** $= 1$, the dqds algorithm failed to converge, if **info** $= 2$, inverse iteration failed to converge.

7 Accuracy

See the description for **abstol**. See also Section 4.7 of Anderson *et al.* 1999 and Barlow and Demmel 1990 for further details.

8 Further Comments

The total number of floating point operations required to compute all the eigenvalues and eigenvectors is approximately proportional to n^2.

The real analogue of this function is f08jl. The specification of f08jy differs from that of f08jl only in that the array **z** is declared as **complex*16** in f08jy.

9 Example

```
jobz = 'V';
range = 'A';
d = [1;
     4;
     9;
     16];
e = [1;
     2;
     3];
vl = 0;
vu = 0;
il = int32(0);
iu = int32(0);
abstol = 0;
[dOut, eOut, m, w, z, isuppz, info] = f08jy(jobz, range, d, e, vl, vu,
il, iu, abstol)
```

```
dOut =
    0.3524
    0.5151
    0.5862
    0.0000
eOut =
    2.8374
    3.8831
    5.1175
m =
           4
w =
    0.6476
    3.5470
    8.6578
   17.1477
z =
    0.9396    0.3388    0.0494    0.0034
   -0.3311    0.8628    0.3781    0.0545
    0.0853   -0.3648    0.8558    0.3568
```

```
   -0.0167    0.0879   -0.3497    0.9326
isuppz =
           1
           4
           1
           4
           1
           4
           1
           4
info =
           0
```