NAG Toolbox for MATLAB

f02xu

1 Purpose

f02xu returns all, or part, of the singular value decomposition of a complex upper triangular matrix.

2 Syntax

```
[a, b, q, sv, rwork, ifail] = f02xu(a, b, wantq, wantp, 'n', n, 'ncolb',
ncolb)
```

3 Description

The n by n upper triangular matrix R is factorized as

$$R = QSP^{\mathrm{H}},$$

where Q and P are n by n unitary matrices and S is an n by n diagonal matrix with real nonnegative diagonal elements, $sv_1, sv_2, \ldots, sv_n$, ordered such that

$$sv_1 \geq sv_2 \geq \cdots \geq sv_n \geq 0.$$

The columns of Q are the left-hand singular vectors of R, the diagonal elements of S are the singular values of R and the columns of P are the right-hand singular vectors of R.

Either or both of Q and P^{H} may be requested and the matrix C given by

$$C = Q^{\mathrm{H}}B,$$

where B is an n by ncolb given matrix, may also be requested.

f02xu obtains the singular value decomposition by first reducing R to bidiagonal form by means of Givens plane rotations and then using the QR algorithm to obtain the singular value decomposition of the bidiagonal form.

Good background descriptions to the singular value decomposition are given in Dongarra *et al.* 1979, Hammarling 1985 and Wilkinson 1978.

Note that if K is any unitary diagonal matrix so that

$$KK^{\mathrm{H}} = I,$$

then

$$A = (QK)S(PK)^{H}$$

is also a singular value decomposition of A.

4 References

Dongarra J J, Moler C B, Bunch J R and Stewart G W 1979 *LINPACK Users' Guide* SIAM, Philadelphia

Hammarling S 1985 The singular value decomposition in multivariate statistics *SIGNUM Newsl.* **20 (3)** 2–25

Wilkinson J H 1978 Singular Value Decomposition – Basic Aspects *Numerical Software – Needs and Availability* (ed D A H Jacobs) Academic Press

5 Parameters

5.1 Compulsory Input Parameters

1: **a(lda,$*$) – complex array**

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

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

The leading n by n upper triangular part of the array **a** must contain the upper triangular matrix R.

2: **b(ldb,$*$) – complex array**

The first dimension, **ldb**, of the array **b** must satisfy

if $\mathbf{ncolb} > 0$, $\mathbf{ldb} \geq \max(1, \mathbf{n})$;
$\mathbf{ldb} \geq 1$ otherwise.

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

If $\mathbf{ncolb} > 0$, the leading n by $ncolb$ part of the array **b** must contain the matrix to be transformed.

3: **wantq – logical scalar**

Must be **true** if the matrix Q is required.

If **wantq** = **false** then the array **q** is not referenced.

4: **wantp – logical scalar**

Must be **true** if the matrix P^{H} is required, in which case P^{H} is returned in the array **a**, otherwise **wantp** must be **false**.

5.2 Optional Input Parameters

1: **n – int32 scalar**

Default: The second dimension of the array **a**.

n, the order of the matrix R.

If $\mathbf{n} = 0$, an immediate return is effected.

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

2: **ncolb – int32 scalar**

Default: The second dimension of the array **b**.

$ncolb$, the number of columns of the matrix B.

If $\mathbf{ncolb} = 0$, the array **b** is not referenced.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldq, cwork

5.4 Output Parameters

1: **a(lda,$*$) – complex array**

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

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

If **wantp** $=$ **true**, the n by n part of **a** will contain the n by n unitary matrix P^{H}, otherwise the n by n upper triangular part of **a** is used as internal workspace, but the strictly lower triangular part of **a** is not referenced.

2: **b(ldb,$*$) – complex array**

The first dimension, **ldb**, of the array **b** must satisfy

if **ncolb** > 0, **ldb** $\geq \max(1, \mathbf{n})$;
ldb ≥ 1 otherwise.

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

Contains the n by $ncolb$ matrix $Q^{\mathrm{H}}B$.

3: **q(ldq,$*$) – complex array**

The first dimension, **ldq**, of the array **q** must satisfy

if **wantq** $=$ **true**, **ldq** $\geq \max(1, \mathbf{n})$;
ldq ≥ 1 otherwise.

The second dimension of the array must be at least $\max(1, \mathbf{n})$ if **wantq** $=$ **true**, and at least 1 otherwise

If **wantq** $=$ **true**, the leading n by n part of the array **q** will contain the unitary matrix Q. Otherwise the array **q** is not referenced.

4: **sv($*$) – double array**

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

The n diagonal elements of the matrix S.

5: **rwork($*$) – double array**

Note: the dimension of the array **rwork** must be at least $\max(1, lrwork)$, where $lrwork$ must satisfy:

$lrwork = 2 \times (\mathbf{n} - 1)$, when **ncolb** $= 0$, **wantq** and **wantp** are **false**,

$lrwork = 3 \times (\mathbf{n} - 1)$, when **ncolb** $= 0$, **wantq** $=$ **false** and **wantp** $=$ **true**, or **ncolb** > 0 and **wantp** $=$ **false**, or **wantq** $=$ **true** and **wantp** $=$ **false**,

$lrwork = 5 \times (\mathbf{n} - 1)$, otherwise.

rwork(n) contains the total number of iterations taken by the QR algorithm.

The rest of the array is used as workspace.

6: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= -1$

On entry, **n** < 0,
or **lda** $<$ **n**,
or **ncolb** < 0,
or **ldb** $<$ **n** and **ncolb** > 0,
or **ldq** $<$ **n** and **wantq** $=$ **true**

ifail > 0

The QR algorithm has failed to converge in $50 \times$ **n** iterations. In this case **sv**(1), **sv**$(2), \ldots,$ **sv**(**ifail**) may not have been found correctly and the remaining singular values may not be the smallest. The matrix R will nevertheless have been factorized as $R = QEP^{\mathrm{H}}$, where E is a bidiagonal matrix with **sv**(1), **sv**$(2), \ldots,$ **sv**(n) as the diagonal elements and **rwork**(1), **rwork**$(2), \ldots,$ **rwork**$(n-1)$ as the superdiagonal elements.

This failure is not likely to occur.

7 Accuracy

The computed factors Q, S and P satisfy the relation

$$QSP^{\mathrm{H}} = A + E,$$

where

$$\|E\| \leq c\epsilon\|A\|,$$

ϵ is the ***machine precision***, c is a modest function of n and $\|.\|$ denotes the spectral (two) norm. Note that $\|A\| = sv_1$.

8 Further Comments

For given values of **ncolb**, **wantq** and **wantp**, the number of floating-point operations required is approximately proportional to n^3.

>Following the use of f02xu the rank of R may be estimated as follows:

```
tol = eps;
irank = 1;
while (irank <= numel(sv) && sv(irank) >= tol*sv(1) )
  irank = irank + 1;
end
```

returns the value k in `irank`, where k is the smallest integer for which **sv**$(k) < tol \times$ **sv**(1), where tol is typically the machine precision, so that `irank` is an estimate of the rank of S and thus also of R.

9 Example

```
a = [complex(1, +0), complex(1, +1), complex(1, +1);
     complex(0, 0), complex(-2, +0), complex(-1, -1);
     complex(0, 0), complex(0, 0), complex(-3, +0)];
b = [complex(1, +1); complex(-1,0); complex(-1,1)];
wantq = true;
wantp = true;
[aOut, bOut, q, sv, rwork, ifail] = f02xu(a, b, wantq, wantp)
```

```
aOut =
  -0.1275             -0.3899 - 0.2046i  -0.5289 - 0.7142i
  -0.2265             -0.3397 - 0.7926i   0.0000 + 0.4529i
   0.9656             -0.1311 - 0.2129i  -0.0698 + 0.0119i
bOut =
  -1.9656 - 0.7935i
   0.1132 - 0.3397i
   0.0915 + 0.6086i
q =
  -0.5005             -0.4529             0.7378
   0.5152 - 0.1514i   0.1132 - 0.5661i   0.4190 - 0.4502i
   0.4041 - 0.5457i  -0.0000 + 0.6794i   0.2741 + 0.0468i
sv =
    3.9263
    2.0000
    0.7641
```

```
rwork =
   -0.0000
    0.0000
    6.0000
    3.2500
    3.2500
   -0.0000
    0.9992
    1.0000
    0.0407
   -0.0000
ifail =
           0
```