NAG Toolbox for MATLAB

f08qv

1 Purpose

f08qv solves the complex triangular Sylvester matrix equation.

2 Syntax

```
[c, scal, info] = f08qv(trana, tranb, isgn, a, b, c, 'm', m, 'n', n)
```

3 Description

f08qv solves the complex Sylvester matrix equation

$$\mathrm{op}(A)X \pm X\,\mathrm{op}(B) = \alpha C,$$

where $\mathrm{op}(A) = A$ or A^{H}, and the matrices A and B are upper triangular; α is a scale factor (≤ 1) determined by the function to avoid overflow in X; A is m by m and B is n by n while the right-hand side matrix C and the solution matrix X are both m by n. The matrix X is obtained by a straightforward process of back-substitution (see Golub and Van Loan 1996).

Note that the equation has a unique solution if and only if $\alpha_i \pm \beta_j \neq 0$, where $\{\alpha_i\}$ and $\{\beta_j\}$ are the eigenvalues of A and B respectively and the sign ($+$ or $-$) is the same as that used in the equation to be solved.

4 References

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

Higham N J 1992 Perturbation theory and backward error for $AX - XB = C$ *Numerical Analysis Report* University of Manchester

5 Parameters

5.1 Compulsory Input Parameters

1: **trana – string**

Specifies the option $\mathrm{op}(A)$.

trana = 'N'

$\mathrm{op}(A) = A$.

trana = 'C'

$\mathrm{op}(A) = A^{\mathrm{H}}$.

Constraint: **trana** = 'N' or 'C'.

2: **tranb – string**

Specifies the option $\mathrm{op}(B)$.

tranb = 'N'

$\mathrm{op}(B) = B$.

tranb = 'C'

$\mathrm{op}(B) = B^{\mathrm{H}}$.

Constraint: **tranb** = 'N' or 'C'.

3: **isgn – int32 scalar**

Indicates the form of the Sylvester equation.

isgn $= +1$

The equation is of the form $\mathrm{op}(A)X + X\,\mathrm{op}(B) = \alpha C$.

isgn $= -1$

The equation is of the form $\mathrm{op}(A)X - X\,\mathrm{op}(B) = \alpha C$.

Constraint: **isgn** $= +1$ or -1.

4: **a(lda,$*$) – complex array**

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

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

The m by m upper triangular matrix A.

5: **b(ldb,$*$) – complex array**

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

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

The n by n upper triangular matrix B.

6: **c(ldc,$*$) – complex array**

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

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

The m by n right-hand side matrix C.

5.2 Optional Input Parameters

1: **m – int32 scalar**

Default: The first dimension of the array **a** and the second dimension of the arrays **a**, **c**. (An error is raised if these dimensions are not equal.)

m, the order of the matrix A, and the number of rows in the matrices X and C.

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

2: **n – int32 scalar**

Default: The first dimension of the array **b** and the second dimension of the array **b**. (An error is raised if these dimensions are not equal.)

n, the order of the matrix B, and the number of columns in the matrices X and C.

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

5.3 Input Parameters Omitted from the MATLAB Interface

lda, ldb, ldc

5.4 Output Parameters

1: **c(ldc,$*$) – complex array**

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

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

c contains the solution matrix X.

2: **scal – double scalar**

The value of the scale factor α.

3: **info – int32 scalar**

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

6 Error Indicators and Warnings

info $= -i$

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

1: **trana**, 2: **tranb**, 3: **isgn**, 4: **m**, 5: **n**, 6: **a**, 7: **lda**, 8: **b**, 9: **ldb**, 10: **c**, 11: **ldc**, 12: **scal**, 13: **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 $= 1$

A and B have common or close eigenvalues, perturbed values of which were used to solve the equation.

7 Accuracy

Consider the equation $AX - XB = C$. (To apply the remarks to the equation $AX + XB = C$, simply replace B by $-B$.)

Let $\tilde{X}$ be the computed solution and R the residual matrix:

$$R = C - \left(A\tilde{X} - \tilde{X}B\right).$$

Then the residual is always small:

$$\|R\|_F = O(\epsilon)\left(\|A\|_F + \|B\|_F\right)\left\|\tilde{X}\right\|_F.$$

However, $\tilde{X}$ is **not** necessarily the exact solution of a slightly perturbed equation; in other words, the solution is not backwards stable.

For the forward error, the following bound holds:

$$\left\|\tilde{X} - X\right\|_F \leq \frac{\|R\|_F}{sep(A, B)}$$

but this may be a considerable over estimate. See Golub and Van Loan 1996 for a definition of $sep(A, B)$, and Higham 1992 for further details.

These remarks also apply to the solution of a general Sylvester equation, as described in Section 8.

8 Further Comments

The total number of real floating-point operations is approximately $4mn(m + n)$.

To solve the **general** complex Sylvester equation

$$AX \pm XB = C$$

where A and B are general matrices, A and B must first be reduced to Schur form (by calling f08pn, for example):

$$A = Q_1 \tilde{A} Q_1^{\mathrm{H}} \qquad \text{and} \qquad B = Q_2 \tilde{B} Q_2^{\mathrm{H}}$$

where $\tilde{A}$ and $\tilde{B}$ are upper triangular and Q_1 and Q_2 are unitary. The original equation may then be transformed to:

$$\tilde{A}\tilde{X} \pm \tilde{X}\tilde{B} = \tilde{C}$$

where $\tilde{X} = Q_1^{\mathrm{H}} X Q_2$ and $\tilde{C} = Q_1^{\mathrm{H}} C Q_2$. $\tilde{C}$ may be computed by matrix multiplication; f08qv may be used to solve the transformed equation; and the solution to the original equation can be obtained as $X = Q_1 \tilde{X} Q_2^{\mathrm{H}}$.

The real analogue of this function is f08qh.

9 Example

```
trana = 'No transpose';
tranb = 'No transpose';
isgn = int32(1);
a = [complex(-6, -7), complex(0.36, -0.36), complex(-0.19, +0.48),
complex(0.88, -0.25);
       complex(0, +0), complex(-5, +2), complex(-0.03, -0.72), complex(-
0.23, +0.13);
                complex(0, +0), complex(0, +0), complex(8, -1),
complex(0.9399999999999999, +0.53);
     complex(0, +0), complex(0, +0), complex(0, +0), complex(3, -4)];
b = [complex(0.5, -0.2), complex(-0.29, -0.16), complex(-0.37, +0.84),
complex(-0.55, +0.73);
     complex(0, +0), complex(-0.4, +0.9), complex(0.06, +0.22), complex(-
0.43, +0.17);
      complex(0, +0), complex(0, +0), complex(-0.9, -0.1), complex(-0.89,
-0.42);
     complex(0, +0), complex(0, +0), complex(0, +0), complex(0.3, -0.7)];
c = [complex(0.63, +0.35), complex(0.45, -0.5600000000000001),
complex(0.08, -0.14), complex(-0.17, -0.23);
          complex(-0.17, +0.09), complex(-0.07000000000000001, -0.31),
complex(0.27, -0.54), complex(0.35, +1.21);
      complex(-0.93, -0.44), complex(-0.33, -0.35), complex(0.41, -0.03),
complex(0.57, +0.84);
      complex(0.54, +0.25), complex(-0.62, -0.05), complex(-0.52, -0.13),
complex(0.11, -0.08)];
[cOut, scal, info] = f08qv(trana, tranb, isgn, a, b, c)
```

```
cOut =
   -0.0611 + 0.0249i   -0.0031 + 0.0798i   -0.0062 + 0.0165i    0.0054 -
0.0063i
    0.0215 - 0.0003i   -0.0155 + 0.0570i   -0.0665 + 0.0718i    0.0290 -
0.2636i
   -0.0949 - 0.0785i   -0.0415 - 0.0298i    0.0357 + 0.0244i    0.0284 +
0.1108i
    0.0281 + 0.1052i   -0.0970 - 0.1214i   -0.0271 - 0.0940i    0.0402 +
0.0048i
scal =
     1
info =
           0
```