# NAG Library Routine Document

# F01JEF

**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

F01JEF computes an estimate of the relative condition number $\kappa_{A^p}$ of the $p$th power (where $p$ is real) of a real $n$ by $n$ matrix $A$, in the 1-norm. The principal matrix power $A^p$ is also returned.

## 2 Specification

```
SUBROUTINE F01JEF (N, A, LDA, P, CONDPA, IFAIL)
INTEGER              N, LDA, IFAIL
REAL (KIND=nag_wp) A(LDA,*), P, CONDPA
```

## 3 Description

For a matrix $A$ with no eigenvalues on the closed negative real line, $A^p$ ($p \in \mathbb{R}$) can be defined as

$$A^p = \exp(p \log(A))$$

where $\log(A)$ is the principal logarithm of $A$ (the unique logarithm whose spectrum lies in the strip $\{z : -\pi < \mathrm{Im}(z) < \pi\}$).

The Fréchet derivative of the matrix $p$th power of $A$ is the unique linear mapping $E \mapsto L(A, E)$ such that for any matrix $E$

$$(A+E)^p - A^p - L(A, E) = o(\|E\|).$$

The derivative describes the first-order effect of perturbations in $A$ on the matrix power $A^p$.

The relative condition number of the matrix $p$th power can be defined by

$$\kappa_{A^p} = \frac{\|L(A)\|\|A\|}{\|A^p\|},$$

where $\|L(A)\|$ is the norm of the Fréchet derivative of the matrix power at $A$.

F01JEF uses the algorithms of Higham and Lin (2011) and Higham and Lin (2013) to compute $\kappa_{A^p}$ and $A^p$. The real number $p$ is expressed as $p = q + r$ where $q \in (-1, 1)$ and $r \in \mathbb{Z}$. Then $A^p = A^q A^r$. The integer power $A^r$ is found using a combination of binary powering and, if necessary, matrix inversion. The fractional power $A^q$ is computed using a Schur decomposition, a Padé approximant and the scaling and squaring method.

To obtain an estimate of $\kappa_{A^p}$, F01JEF first estimates $\|L(A)\|$ by computing an estimate $\gamma$ of a quantity $K \in \left[n^{-1}\|L(A)\|_1, n\|L(A)\|_1\right]$, such that $\gamma \le K$. This requires multiple Fréchet derivatives to be computed. Fréchet derivatives of $A^q$ are obtained by differentiating the Padé approximant. Fréchet derivatives of $A^p$ are then computed using a combination of the chain rule and the product rule for Fréchet derivatives.

## 4 References

Higham N J (2008) *Functions of Matrices: Theory and Computation* SIAM, Philadelphia, PA, USA

Higham N J and Lin L (2011) A Schur–Padé algorithm for fractional powers of a matrix *SIAM J. Matrix Anal. Appl.* **32(3)** 1056–1078

Higham N J and Lin L (2013) An improved Schur–Padé algorithm for fractional powers of a matrix and their Fréchet derivatives *MIMS Eprint 2013.1* Manchester Institute for Mathematical Sciences, School of Mathematics, University of Manchester http://eprints.ma.man.ac.uk/

## 5 Parameters

1: N – INTEGER *Input*

*On entry*: $n$, the order of the matrix $A$.

*Constraint*: $N \geq 0$.

2: $A(LDA, *)$ – REAL (KIND=nag_wp) array *Input/Output*

**Note**: the second dimension of the array A must be at least N.

*On entry*: the $n$ by $n$ matrix $A$.

*On exit*: the $n$ by $n$ principal matrix $p$th power, $A^p$.

3: LDA – INTEGER *Input*

*On entry*: the first dimension of the array A as declared in the (sub)program from which F01JEF is called.

*Constraint*: $LDA \geq N$.

4: P – REAL (KIND=nag_wp) *Input*

*On entry*: the required power of $A$.

5: CONDPA – REAL (KIND=nag_wp) *Output*

*On exit*: if $IFAIL = 0$ or 3, an estimate of the relative condition number of the matrix $p$th power, $\kappa_{A^p}$. Alternatively, if $IFAIL = 4$, the absolute condition number of the matrix $p$th power.

6: IFAIL – INTEGER *Input/Output*

*On entry*: IFAIL must be set to 0, $-1$ or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value $-1$ or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value $-1$ or 1 is used it is essential to test the value of IFAIL on exit.**

*On exit*: $IFAIL = 0$ unless the routine detects an error or a warning has been flagged (see Section 6).

## 6 Error Indicators and Warnings

If on entry IFAIL $= 0$ or $-1$, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL $= 1$

$A$ has eigenvalues on the negative real line. The principal $p$th power is not defined in this case; F01KEF can be used to find a complex, non-principal $p$th power.

IFAIL $= 2$

$A$ is singular so the $p$th power cannot be computed.

IFAIL $= 3$

$A^p$ has been computed using an IEEE double precision Padé approximant, although the arithmetic precision is higher than IEEE double precision.

IFAIL $= 4$

The relative condition number is infinite. The absolute condition number was returned instead.

IFAIL $= 5$

An unexpected internal error occurred. This failure should not occur and suggests that the routine has been called incorrectly.

IFAIL $= -1$

On entry, N $= \langle$*value*$\rangle$.
Constraint: N $\geq 0$.

IFAIL $= -3$

On entry, LDA $= \langle$*value*$\rangle$ and N $= \langle$*value*$\rangle$.
Constraint: LDA $\geq$ N.

IFAIL $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

Your licence key may have expired or may not have been installed correctly.

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

## 7 Accuracy

F01JEF uses the norm estimation routine F04YDF to produce an estimate $\gamma$ of a quantity $K \in \left[n^{-1}\|L(A)\|_1, n\|L(A)\|_1\right]$, such that $\gamma \leq K$. For further details on the accuracy of norm estimation, see the documentation for F04YDF.

For a normal matrix $A$ (for which $A^{\mathrm{T}}A = AA^{\mathrm{T}}$), the Schur decomposition is diagonal and the computation of the fractional part of the matrix power reduces to evaluating powers of the eigenvalues of

$A$ and then constructing $A^p$ using the Schur vectors. This should give a very accurate result. In general, however, no error bounds are available for the algorithm. See Higham and Lin (2011) and Higham and Lin (2013) for details and further discussion.

## 8 Parallelism and Performance

F01JEF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F01JEF 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

The amount of real allocatable memory required by the algorithm is typically of the order $10 \times n^2$.

The cost of the algorithm is $O(n^3)$ floating-point operations; see Higham and Lin (2013).

If the matrix $p$th power alone is required, without an estimate of the condition number, then F01EQF should be used. If the Fréchet derivative of the matrix power is required then F01JFF should be used. If $A$ has negative real eigenvalues then F01KEF can be used to return a complex, non-principal $p$th power and its condition number.

## 10 Example

This example estimates the relative condition number of the matrix power $A^p$, where $p = 0.2$ and

$$A = \begin{pmatrix} 3 & 3 & 2 & 1 \\ 1 & 1 & 0 & 2 \\ 1 & 4 & 4 & 2 \\ 3 & 1 & 3 & 1 \end{pmatrix}.$$

### 10.1 Program Text

```
    Program f01jefe

!     F01JEF Example Program Text

!     Mark 25 Release. NAG Copyright 2014.

!     .. Use Statements ..
      Use nag_library, Only: f01jef, nag_wp, x04caf
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Real (Kind=nag_wp)               :: condpa, p
      Integer                          :: i, ifail, lda, n
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: a(:,:)
!     .. Executable Statements ..
      Write (nout,*) 'F01JEF Example Program Results'
      Write (nout,*)
      Flush (nout)
!     Skip heading in data file
      Read (nin,*)
      Read (nin,*) n, p
      lda = n
      Allocate (a(lda,n))
```

```
!     Read A from data file
      Read (nin,*)(a(i,1:n),i=1,n)

!     ifail: behaviour on error exit
!             =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0

!     Find A^p
      Call f01jef(n,a,lda,p,condpa,ifail)

!     Print solution
      Call x04caf('General',' ',n,n,a,lda,'A^p',ifail)

      Write (nout,*)
      Write (nout,99999) 'Estimated condition number is: ', condpa
99999 Format (1X,A,F6.2)
    End Program f01jefe
```

## 10.2 Program Data

```
F01JEF Example Program Data

4     0.2              : Values of N and P

3.0   3.0   2.0   1.0
1.0   1.0   0.0   2.0
1.0   4.0   4.0   2.0
3.0   1.0   3.0   1.0  : End of matrix A
```

## 10.3 Program Results

```
 F01JEF Example Program Results

 A^p
             1          2          3          4
 1      1.2368     0.1977     0.1749    -0.0314
 2     -0.0543     1.1643    -0.0947     0.3145
 3      0.0537     0.3514     1.3254     0.0214
 4      0.3339    -0.2125     0.1880     1.0581

 Estimated condition number is:   2.75
```