

NAG Library Routine Document

F01KHF

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

F01KHF computes the Fréchet derivative $L(A, E)$ of the matrix exponential of a complex n by n matrix A applied to the complex n by n matrix E . The matrix exponential e^A is also returned.

2 Specification

```
SUBROUTINE F01KHF (N, A, LDA, E, LDE, IFAIL)
  INTEGER          N, LDA, LDE, IFAIL
  COMPLEX (KIND=nag_wp) A(LDA,*), E(LDE,*)
```

3 Description

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

$$e^{A+E} - e^A - L(A, E) = o(\|E\|).$$

The derivative describes the first-order effect of perturbations in A on the exponential e^A .

F01KHF uses the algorithms of Al-Mohy and Higham (2009a) and Al-Mohy and Higham (2009b) to compute e^A and $L(A, E)$. The matrix exponential e^A is computed using a Padé approximant and the scaling and squaring method. The Padé approximant is then differentiated in order to obtain the Fréchet derivative $L(A, E)$.

4 References

Al-Mohy A H and Higham N J (2009a) A new scaling and squaring algorithm for the matrix exponential *SIAM J. Matrix Anal.* **31(3)** 970–989

Al-Mohy A H and Higham N J (2009b) Computing the Fréchet derivative of the matrix exponential, with an application to condition number estimation *SIAM J. Matrix Anal. Appl.* **30(4)** 1639–1657

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

Moler C B and Van Loan C F (2003) Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later *SIAM Rev.* **45** 3–49

5 Parameters

- 1: N – INTEGER *Input*
On entry: n , the order of the matrix A .
Constraint: $N \geq 0$.
- 2: A(LDA,*) – COMPLEX (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 matrix exponential e^A .

- 3: LDA – INTEGER *Input*
On entry: the first dimension of the array A as declared in the (sub)program from which F01KHF is called.
Constraint: $LDA \geq N$.
- 4: E(LDE, *) – COMPLEX (KIND=nag_wp) array *Input/Output*
Note: the second dimension of the array E must be at least N.
On entry: the n by n matrix E
On exit: the Fréchet derivative $L(A, E)$
- 5: LDE – INTEGER *Input*
On entry: the first dimension of the array E as declared in the (sub)program from which F01KHF is called.
Constraint: $LDE \geq N$.
- 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

The linear equations to be solved for the Padé approximant are singular; it is likely that this routine has been called incorrectly.

IFAIL = 2

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

IFAIL = 3

An unexpected internal error has occurred. Please contact NAG.

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 = -5

On entry, LDE = $\langle value \rangle$ and N = $\langle value \rangle$.
Constraint: LDE \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

For a normal matrix A (for which $A^H A = A A^H$) the computed matrix, e^A , is guaranteed to be close to the exact matrix, that is, the method is forward stable. No such guarantee can be given for non-normal matrices. See Section 10.3 of Higham (2008), Al-Mohy and Higham (2009a) and Al-Mohy and Higham (2009b) for details and further discussion.

8 Parallelism and Performance

F01KHF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

F01KHF 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 cost of the algorithm is $O(n^3)$ and the complex allocatable memory required is approximately $9n^2$; see Al-Mohy and Higham (2009a) and Al-Mohy and Higham (2009b).

If the matrix exponential alone is required, without the Fréchet derivative, then F01FCF should be used.

If the condition number of the matrix exponential is required then F01KGF should be used.

As well as the excellent book Higham (2008), the classic reference for the computation of the matrix exponential is Moler and Van Loan (2003).

10 Example

This example finds the matrix exponential e^A and the Fréchet derivative $L(A, E)$, where

$$A = \begin{pmatrix} 1 + i & 2 + i & 2 + i & 2 + i \\ 3 + 2i & 1 & 1 & 2 + i \\ 3 + 2i & 2 + i & 1 & 2 + i \\ 3 + 2i & 3 + 2i & 3 + 2i & 1 + i \end{pmatrix} \quad \text{and} \quad E = \begin{pmatrix} 1 & 2 + i & 2 & 4 + i \\ 3 + 2i & 0 & 1 & 0 + i \\ 0 + 2i & 0 + i & 1 & 0 \\ 1 + i & 2 + 2i & 0 + 3i & 1 \end{pmatrix}.$$

10.1 Program Text

```

Program f01khfe

!      F01KHF Example Program Text
!      Mark 25 Release. NAG Copyright 2014.

!      .. Use Statements ..
!      Use nag_library, Only: f01khf, nag_wp, x04daf
!      .. Implicit None Statement ..
!      Implicit None
!      .. Parameters ..
!      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
!      Integer                     :: i, ierr, ifail, lda, lde, n
!      .. Local Arrays ..
!      Complex (Kind=nag_wp), Allocatable :: a(:,,:), e(:,,:)
!      .. Executable Statements ..
!      Write (nout,*) 'F01KHF Example Program Results'
!      Write (nout,*)
!      Flush (nout)
!      Skip heading in data file
!      Read (nin,*)
!      Read (nin,*) n

!      lda = n
!      lde = n
!      Allocate (a(lda,n))
!      Allocate (e(lde,n))

!      Read A from data file

!      Read (nin,*)(a(i,1:n),i=1,n)

!      Read E from data file

!      Read (nin,*)(e(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 exp( A ) and L_exp(A,E)
!      Call f01khf(n,a,lda,e,lde,ifail)

!      Print solution

!      ierr = 0
!      Call x04daf('General',' ',n,n,a,lda,'Exp(A)',ierr)
!      Write (nout,*)
!      Call x04daf('General',' ',n,n,e,lde,'L_exp(A,E)',ierr)

!      End Program f01khfe

```

10.2 Program Data

F01KHF Example Program Data

```

4                                     :Value of N

(1.0,1.0) (2.0,1.0) (2.0,1.0) (2.0,1.0)
(3.0,2.0) (1.0,0.0) (1.0,0.0) (2.0,1.0)
(3.0,2.0) (2.0,1.0) (1.0,0.0) (2.0,1.0)
(3.0,2.0) (3.0,2.0) (3.0,2.0) (1.0,1.0) :End of matrix A

(1.0,0.0) (2.0,1.0) (2.0,0.0) (4.0,1.0)
(3.0,2.0) (0.0,0.0) (1.0,0.0) (0.0,1.0)
(0.0,2.0) (0.0,1.0) (1.0,0.0) (0.0,0.0)
(1.0,1.0) (2.0,2.0) (0.0,3.0) (1.0,0.0) :End of matrix E

```

10.3 Program Results

F01KHF Example Program Results

Exp(A)

	1	2	3	4
1	-157.9003	-194.6526	-186.5627	-155.7669
	-754.3717	-555.0507	-475.4533	-520.1876
2	-206.8899	-225.4985	-212.4414	-186.5627
	-694.7443	-505.3938	-431.0611	-475.4533
3	-208.7476	-238.4962	-225.4985	-194.6526
	-808.2090	-590.8045	-505.3938	-555.0507
4	-133.3958	-208.7476	-206.8899	-157.9003
	-1085.5496	-808.2090	-694.7443	-754.3717

L_exp(A,E)

	1	2	3	4
1	1571.5852	778.4238	500.2085	740.7485
	-4640.2429	-3719.8308	-3246.0234	-3424.1963
2	1472.7846	731.6608	473.2569	692.0895
	-4273.5048	-3432.5961	-2990.9285	-3148.4635
3	1996.4848	1107.9174	782.1266	1031.5808
	-4568.8881	-3714.9923	-3249.1926	-3400.8557
4	3327.1347	2015.2763	1514.3130	1873.9421
	-5829.0773	-4810.2591	-4234.6812	-4404.0163
