

NAG Toolbox

nag_lapack_zhptri (f07pw)

1 Purpose

nag_lapack_zhptri (f07pw) computes the inverse of a complex Hermitian indefinite matrix A , where A has been factorized by nag_lapack_zhptrf (f07pr), using packed storage.

2 Syntax

```
[ap, info] = nag_lapack_zhptri(uplo, ap, ipiv, 'n', n)
[ap, info] = f07pw(uplo, ap, ipiv, 'n', n)
```

3 Description

nag_lapack_zhptri (f07pw) is used to compute the inverse of a complex Hermitian indefinite matrix A , the function must be preceded by a call to nag_lapack_zhptrf (f07pr), which computes the Bunch–Kaufman factorization of A , using packed storage.

If **uplo** = 'U', $A = PUDU^H P^T$ and A^{-1} is computed by solving $U^H P^T XPU = D^{-1}$ for X .

If **uplo** = 'L', $A = PLDL^H P^T$ and A^{-1} is computed by solving $L^H P^T XPL = D^{-1}$ for X .

4 References

Du Croz J J and Higham N J (1992) Stability of methods for matrix inversion *IMA J. Numer. Anal.* **12** 1–19

5 Parameters

5.1 Compulsory Input Parameters

1: **uplo** – CHARACTER(1)

Specifies how A has been factorized.

uplo = 'U'

$A = PUDU^H P^T$, where U is upper triangular.

uplo = 'L'

$A = PLDL^H P^T$, where L is lower triangular.

Constraint: **uplo** = 'U' or 'L'.

2: **ap(:)** – COMPLEX (KIND=nag_wp) array

The dimension of the array **ap** must be at least $\max(1, \mathbf{n} \times (\mathbf{n} + 1)/2)$

The factorization of A stored in packed form, as returned by nag_lapack_zhptrf (f07pr).

3: **ipiv(:)** – INTEGER array

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

Details of the interchanges and the block structure of D , as returned by nag_lapack_zhptrf (f07pr).

5.2 Optional Input Parameters

1: **n** – INTEGER

Default: the dimension of the array **ipiv**.

n , the order of the matrix A .

Constraint: $n \geq 0$.

5.3 Output Parameters

1: **ap(:)** – COMPLEX (KIND=nag_wp) array

The dimension of the array **ap** will be $\max(1, n \times (n + 1)/2)$

The factorization stores the n by n matrix A^{-1} .

More precisely,

if **uplo** = 'U', the upper triangle of A^{-1} must be stored with element A_{ij} in $\mathbf{ap}(i + j(j - 1)/2)$ for $i \leq j$;

if **uplo** = 'L', the lower triangle of A^{-1} must be stored with element A_{ij} in $\mathbf{ap}(i + (2n - j)(j - 1)/2)$ for $i \geq j$.

2: **info** – INTEGER

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

6 Error Indicators and Warnings

info < 0

If **info** = $-i$, argument i had an illegal value. An explanatory message is output, and execution of the program is terminated.

info > 0 (*warning*)

Element $\langle value \rangle$ of the diagonal is exactly zero. D is singular and the inverse of A cannot be computed.

7 Accuracy

The computed inverse X satisfies a bound of the form

if **uplo** = 'U', $|DU^T P^T XPU - I| \leq c(n)\epsilon(|D||U^T|P^T|X|P|U| + |D||D^{-1}|)$;

if **uplo** = 'L', $|DL^T P^T XPL - I| \leq c(n)\epsilon(|D||L^T|P^T|X|P|L| + |D||D^{-1}|)$,

$c(n)$ is a modest linear function of n , and ϵ is the *machine precision*.

8 Further Comments

The total number of real floating-point operations is approximately $\frac{8}{3}n^3$.

The real analogue of this function is nag_lapack_dsptri (f07pj).

9 Example

This example computes the inverse of the matrix A , where

$$A = \begin{pmatrix} -1.36 + 0.00i & 1.58 + 0.90i & 2.21 - 0.21i & 3.91 + 1.50i \\ 1.58 - 0.90i & -8.87 + 0.00i & -1.84 - 0.03i & -1.78 + 1.18i \\ 2.21 + 0.21i & -1.84 + 0.03i & -4.63 + 0.00i & 0.11 + 0.11i \\ 3.91 - 1.50i & -1.78 - 1.18i & 0.11 - 0.11i & -1.84 + 0.00i \end{pmatrix}.$$

Here A is Hermitian indefinite, stored in packed form, and must first be factorized by nag_lapack_zhptrf (f07pr).

9.1 Program Text

```
function f07pw_example

fprintf('f07pw example results\n\n');

% Hermitian indefinite matrix A (Lower triangular part stored in packed form)
uplo = 'L';
n = nag_int(4);
ap = [-1.36 + 0i; 1.58 - 0.9i; 2.21 + 0.21i; 3.91 - 1.5i;
       -8.87 + 0i; -1.84 + 0.03i; -1.78 - 1.18i;
       -4.63 + 0i; 0.11 - 0.11i;
       -1.84 + 0i];

% Factorize
[apf, ipiv, info] = f07pr( ...
    uplo, n, ap);

% Invert
[ainv, info] = f07pw( ...
    uplo, apf, ipiv);

[ifail] = x04dc( ...
    uplo, 'Non-unit', n, ainv, 'Inverse');
```

9.2 Program Results

```
f07pw example results

Inverse
      1         2         3         4
1   0.0826
     0.0000
2   -0.0335 -0.1408
     0.0440  0.0000
3   0.0603  0.0422 -0.2007
     -0.0105 -0.0222  0.0000
4   0.2391  0.0304  0.0982  0.0073
     -0.0926  0.0203 -0.0635  0.0000
```
