NAG Library Routine Document

C09EDF

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

C09EDF computes the inverse two-dimensional multi-level discrete wavelet transform (DWT). This routine reconstructs data from (possibly filtered or otherwise manipulated) wavelet transform coefficients calculated by C09ECF from an original input matrix. The initialization routine C09ABF must be called first to set up the DWT options.

2 Specification

```
SUBROUTINE C09EDF (NWL, LENC, C, M, N, B, LDB, ICOMM, IFAIL)

INTEGER            NWL, LENC, M, N, LDB, ICOMM(180), IFAIL
REAL (KIND=nag_wp) C(LENC), B(LDB,N)
```

3 Description

C09EDF performs the inverse operation of C09ECF. That is, given a set of wavelet coefficients, computed by C09ECF using a DWT as set up by the initialization routine C09ABF, on a real matrix, A, C09EDF will reconstruct A. The reconstructed matrix is referred to as B in the following since it will not be identical to A when the DWT coefficients have been filtered or otherwise manipulated prior to reconstruction. If the original input matrix is level 0, then it is possible to terminate reconstruction at a higher level by specifying fewer than the number of levels used in the call to C09ECF. This results in a partial reconstruction.

4 References

None.

5 Parameters

1: NWL – INTEGER *Input*

On entry: the number, n_l, of levels to be used in the inverse multi-level transform.

Constraint: $1 \leq \text{NWL} \leq n_{\text{fwd}}$, where n_{fwd} is the value used in a preceding call to C09ECF.

2: LENC – INTEGER *Input*

On entry: the dimension of the array C as declared in the (sub)program from which C09EDF is called.

Constraint: $\text{LENC} \geq n_{\text{ct}}$, where n_{ct} is the total number of coefficients that correspond to a transform with n_{fwd} levels and is unchanged from the preceding call to C09ECF.

3: C(LENC) – REAL (KIND=nag_wp) array *Input*

On entry: the coefficients of a multi-level wavelet transform of the original matrix, A, which may have been filtered or otherwise manipulated.

Let $q(i)$ be the number of coefficients (of each type) at level i, for $i = n_{\text{fwd}}, n_{\text{fwd}} - 1, \ldots, 1$. Then, setting $k_1 = q(n_{\text{fwd}})$ and $k_{j+1} = k_j + q(n_{\text{fwd}} - \lceil j/3 \rceil + 1)$, for $j = 1, 2, \ldots, 3n_{\text{fwd}}$, the coefficients are stored in C as follows:

C(i), for $i = 1, 2, \ldots, k_1$
Contains the level n_{fwd} approximation coefficients, $a_{n_{\text{fwd}}}$.

C(i), for $i = k_j + 1, \ldots, k_{j+1}$
Contains the level $n_{\text{fwd}} - \lceil j/3 \rceil + 1$ vertical, horizontal and diagonal coefficients. These are:

vertical coefficients if j mod $3 = 1$;

horizontal coefficients if j mod $3 = 2$;

diagonal coefficients if j mod $3 = 0$,

for $j = 1, \ldots, 3n_{\text{fwd}}$.

4: M – INTEGER *Input*

On entry: the number of elements, m, in the first dimension of the reconstructed matrix B. For a full reconstruction of n_{fwd} levels this is the same as parameter M in C09ECF. For a partial reconstruction of $n_l < n_{\text{fwd}}$ levels this will be equal to DWTLVM$(n_l + 1)$ as returned from C09ECF.

5: N – INTEGER *Input*

On entry: the number of elements, n, in the second dimension of the reconstructed matrix B. For a full reconstruction of n_{fwd} levels this is the same as parameter N in C09ECF. For a partial reconstruction of $n_l < n_{\text{fwd}}$ levels this will be equal to DWTLVN$(n_l + 1)$ as returned from C09ECF.

6: B(LDB,N) – REAL (KIND=nag_wp) array *Output*

On exit: the m by n reconstructed matrix, B, based on the input multi-level wavelet transform coefficients and the transform options supplied to the initialization routine C09ABF.

7: LDB – INTEGER *Input*

On entry: the first dimension of the array B as declared in the (sub)program from which C09EDF is called.

Constraint: LDB $\geq$ M.

8: ICOMM(180) – INTEGER array *Communication Array*

On entry: contains details of the discrete wavelet transform and the problem dimension as setup in the call to the initialization routine C09ABF.

9: 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 $= 1$

On entry, NWL < 1,
or NWL $>$ the number of levels used in the computation of the wavelet coefficients by a call to C09ECF.

IFAIL $= 2$

On entry, LDB $<$ M.

IFAIL $= 3$

On entry, LENC is too small. LENC must be at least the number of wavelet coefficients required for a transform operating on NWL levels. If NWL $= l_{\max}$, the maximum number of levels as returned in NWL by the initial call to C09ABF, then LENC must be at least n_{ct}, the value returned in NWCT by the same call to C09ABF.

IFAIL $= 4$

On entry, M is too small for the required level of reconstruction,
or N is too small for the required level of reconstruction.

IFAIL $= 6$

On entry, the initialization routine C09ABF has not been called first or it has been called with WTRANS $=$ 'S', or the communication array ICOMM has become corrupted.

7 Accuracy

The accuracy of the wavelet transform depends only on the floating point operations used in the convolution and downsampling and should thus be close to ***machine precision***.

8 Further Comments

None.

9 Example

See Section 9 in C09ECF.