NAG Library Routine Document

C09ECF

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

C09ECF computes the two-dimensional multi-level discrete wavelet transform (DWT). The initialization routine C09ABF must be called first to set up the DWT options.

2 Specification

```
SUBROUTINE C09ECF (M, N, A, LDA, LENC, C, NWL, DWTLVM, DWTLVN, ICOMM,      &
                   IFAIL)
INTEGER            M, N, LDA, LENC, NWL, DWTLVM(NWL), DWTLVN(NWL),         &
                   ICOMM(180), IFAIL
REAL (KIND=nag_wp) A(LDA,N), C(LENC)
```

3 Description

C09ECF computes the multi-level DWT of two-dimensional data. For a given wavelet and end extension method, C09ECF will compute a multi-level transform of a matrix A, using a specified number, n_{fwd}, of levels. The number of levels specified, n_{fwd}, must be no more than the value l_{max} returned in NWLMAX by the initialization routine C09ABF for the given problem. The transform is returned as a set of coefficients for the different levels (packed into a single array) and a representation of the multi-level structure.

The notation used here assigns level 0 to the input matrix, A. Level 1 consists of the first set of coefficients computed: the vertical (v_1), horizontal (h_1) and diagonal (d_1) coefficients are stored at this level while the approximation (a_1) coefficients are used as the input to a repeat of the wavelet transform at the next level. This process is continued until, at level n_{fwd}, all four types of coefficients are stored. The output array, C, stores these sets of coefficients in reverse order, starting with $a_{n_{\mathrm{fwd}}}$ followed by $v_{n_{\mathrm{fwd}}}, h_{n_{\mathrm{fwd}}}, d_{n_{\mathrm{fwd}}}, v_{n_{\mathrm{fwd}}-1}, h_{n_{\mathrm{fwd}}-1}, d_{n_{\mathrm{fwd}}-1}, \ldots, v_1, h_1, d_1$.

4 References

None.

5 Arguments

1: M – INTEGER *Input*

On entry: number of rows, m, of data matrix A.

Constraint: this must be the same as the value M passed to the initialization routine C09ABF.

2: N – INTEGER *Input*

On entry: number of columns, n, of data matrix A.

Constraint: this must be the same as the value N passed to the initialization routine C09ABF.

3: A(LDA, N) – REAL (KIND=nag_wp) array *Input*

On entry: the m by n data matrix A.

4: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which C09ECF is called.

Constraint: LDA $\geq$ M.

5: LENC – INTEGER *Input*

On entry: the dimension of the array C as declared in the (sub)program from which C09ECF is called. C must be large enough to contain, n_{ct}, wavelet coefficients. The maximum value of n_{ct} is returned in NWCT by the call to the initialization routine C09ABF and corresponds to the DWT being continued for the maximum number of levels possible for the given data set. When the number of levels, n_{fwd}, is chosen to be less than the maximum, $l_{\max}$, then n_{ct} is correspondingly smaller and LENC can be reduced by noting that the vertical, horizontal and diagonal coefficients are stored at every level and that in addition the approximation coefficients are stored for the final level only. The number of coefficients stored at each level is given by $3 \times \lceil \bar{m}/2 \rceil \times \lceil \bar{n}/2 \rceil$ for MODE = 'P' in C09ABF and $3 \times \lfloor (\bar{m} + n_f - 1)/2 \rfloor \times \lfloor (\bar{n} + n_f - 1)/2 \rfloor$ for MODE = 'H', 'W' or 'Z', where the input data is of dimension $\bar{m} \times \bar{n}$ at that level and n_f is the filter length NF provided by the call to C09ABF. At the final level the storage is $4/3$ times this value to contain the set of approximation coefficients.

Constraint: LENC $\geq n_{\text{ct}}$, where n_{ct} is the total number of coefficients that correspond to a transform with NWL levels.

6: C(LENC) – REAL (KIND=nag_wp) array *Output*

On exit: the coefficients of the discrete wavelet transform. If you need to access or modify the approximation coefficients or any specific set of detail coefficients then the use of C09EYF or C09EZF is recommended. For completeness the following description provides details of precisely how the coefficient are stored in C but this information should only be required in rare cases.

Let $q(i)$ denote the number of coefficients (of each type) at level i, for $i = 1, 2, \ldots, n_{\text{fwd}}$, such that $q(i) = \text{DWTLVM}(n_{\text{fwd}} - i + 1) \times \text{DWTLVN}(n_{\text{fwd}} - i + 1)$. Then, letting $k_1 = q(n_{\text{fwd}})$ 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 $j \bmod 3 = 1$;

  horizontal coefficients if $j \bmod 3 = 2$;

  diagonal coefficients if $j \bmod 3 = 0$,

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

7: NWL – INTEGER *Input*

On entry: the number of levels, n_{fwd}, in the multi-level resolution to be performed.

Constraint: $1 \leq \text{NWL} \leq l_{\max}$, where $l_{\max}$ is the value returned in NWLMAX (the maximum number of levels) by the call to the initialization routine C09ABF.

8: DWTLVM(NWL) – INTEGER array *Output*

On exit: the number of coefficients in the first dimension for each coefficient type at each level. DWTLVM(i) contains the number of coefficients in the first dimension (for each coefficient type computed) at the $(n_{\text{fwd}} - i + 1)$th level of resolution, for $i = 1, 2, \ldots, n_{\text{fwd}}$. Thus for the first

$n_{\mathrm{fwd}} - 1$ levels of resolution, DWTLVM$(n_{\mathrm{fwd}} - i + 1)$ is the size of the first dimension of the matrices of vertical, horizontal and diagonal coefficients computed at this level; for the final level of resolution, DWTLVM(1) is the size of the first dimension of the matrices of approximation, vertical, horizontal and diagonal coefficients computed.

9: DWTLVN(NWL) – INTEGER array *Output*

On exit: the number of coefficients in the second dimension for each coefficient type at each level. DWTLVN(i) contains the number of coefficients in the second dimension (for each coefficient type computed) at the $(n_{\mathrm{fwd}} - i + 1)$th level of resolution, for $i = 1, 2, \ldots, n_{\mathrm{fwd}}$. Thus for the first $n_{\mathrm{fwd}} - 1$ levels of resolution, DWTLVN$(n_{\mathrm{fwd}} - i + 1)$ is the size of the second dimension of the matrices of vertical, horizontal and diagonal coefficients computed at this level; for the final level of resolution, DWTLVN(1) is the size of the second dimension of the matrices of approximation, vertical, horizontal and diagonal coefficients computed.

10: 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.

On exit: contains additional information on the computed transform.

11: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation 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 argument, 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, M $= \langle value \rangle$.
Constraint: M $= \langle value \rangle$, the value of M on initialization (see C09ABF).

On entry, N $= \langle value \rangle$.
Constraint: N $= \langle value \rangle$, the value of N on initialization (see C09ABF).

IFAIL $= 2$

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

IFAIL $= 3$

On entry, LENC $= \langle value \rangle$.
Constraint: LENC $\geq \langle value \rangle$, the total number of coefficents to be generated.

IFAIL $= 5$

On entry, NWL $= \langle value \rangle$.
Constraint: NWL ≥ 1.

On entry, NWL $= \langle value \rangle$ and NWLMAX $= \langle value \rangle$ in C09ABF.
Constraint: NWL $\leq$ NWLMAX in C09ABF.

IFAIL $= 7$

Either the initialization routine has not been called first or ICOMM has been corrupted.

Either the initialization routine was called with WTRANS = 'S' or ICOMM has been corrupted.

IFAIL $= -99$

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

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -399$

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

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

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 Parallelism and Performance

C09ECF is not threaded in any implementation.

9 Further Comments

The wavelet coefficients at each level can be extracted from the output array C using the information contained in DWTLVM and DWTLVN on exit (see the descriptions of C, DWTLVM and DWTLVN in Section 5). For example, given an input data set, A, denoising can be carried out by applying a thresholding operation to the detail (vertical, horizontal and diagonal) coefficients at every level. The elements $\mathrm{C}(k_1+1)$ to $\mathrm{C}(k_{n_{\mathrm{fwd}}+1})$, as described in Section 5, contain the detail coefficients, $\hat{c}_{ij}$, for $i = n_{\mathrm{fwd}}, n_{\mathrm{fwd}}-1, \ldots, 1$ and $j = 1, 2, \ldots, 3q(i)$, where $q(i)$ is the number of each type of coefficient at level i and $\hat{c}_{ij} = c_{ij} + \sigma\epsilon_{ij}$ and $\sigma\epsilon_{ij}$ is the transformed noise term. If some threshold parameter α is chosen, a simple hard thresholding rule can be applied as

$$\bar{c}_{ij} = \begin{cases} 0, & \text{if } \left|\hat{c}_{ij}\right| \leq \alpha \\ \hat{c}_{ij}, & \text{if } \left|\hat{c}_{ij}\right| > \alpha, \end{cases}$$

taking $\bar{c}_{ij}$ to be an approximation to the required detail coefficient without noise, c_{ij}. The resulting coefficients can then be used as input to C09EDF in order to reconstruct the denoised signal. See Section 10 in C09EZF for a simple example of denoising.

See the references given in the introduction to this chapter for a more complete account of wavelet denoising and other applications.

10 Example

This example performs a multi-level resolution transform of a dataset using the Daubechies wavelet (see WAVNAM = 'DB2' in C09ABF) using half-point symmetric end extensions, the maximum possible number of levels of resolution, where the number of coefficients in each level and the coefficients themselves are not changed. The original dataset is then reconstructed using C09EDF.

10.1 Program Text

```
    Program c09ecfe

!     C09ECF Example Program Text
!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: c09abf, c09ecf, c09edf, c09eyf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter                :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                           :: i, i1, ifail, ilevel, itype_coeffs,  &
                                           j1, lda, ldb, ldcoefs, lenc, m, n,   &
                                           nf, nwcn, nwct, nwl, nwlinv, nwlmax
      Character (10)                    :: mode, wavnam, wtrans
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable   :: a(:,:), b(:,:), c(:), coefs(:,:)
      Integer, Allocatable              :: dwtlvm(:), dwtlvn(:)
      Integer                           :: icomm(180)
!     .. Executable Statements ..
      Write (nout,*) 'C09ECF Example Program Results'
      Write (nout,*)
!     Skip heading in data file
      Read (nin,*)
!     Read problem parameters
      Read (nin,*) m, n
      Read (nin,*) wavnam, mode
      lda = m
      ldb = m
      Allocate (a(lda,n),b(ldb,n))

      Write (nout,99999) wavnam, mode, m, n

!     Read data array and write it out

      Do i = 1, m
        Read (nin,*) a(i,1:n)
      End Do

      Write (nout,*) ' Input Data     A :'
      Do i = 1, m
        Write (nout,99998) a(i,1:n)
      End Do

!     Query wavelet filter dimensions
!     For Multi-Resolution Analysis, decomposition, wtrans = 'M'
      wtrans = 'Multilevel'
      ifail = 0
      Call c09abf(wavnam,wtrans,mode,m,n,nwlmax,nf,nwct,nwcn,icomm,ifail)

      lenc = nwct
      Allocate (c(lenc),dwtlvm(nwlmax),dwtlvn(nwlmax))

      nwl = nwlmax

!     Perform Discrete Wavelet transform
      ifail = 0
      Call c09ecf(m,n,a,lda,lenc,c,nwl,dwtlvm,dwtlvn,icomm,ifail)
```

```
      Write (nout,99997) nwl
      Write (nout,99996)
      Write (nout,99995) dwtlvm(1:nwl)
      Write (nout,99994)
      Write (nout,99995) dwtlvn(1:nwl)

!     Allocate an array in which to extract coefficients
      ldcoefs = dwtlvm(nwl)
      Allocate (coefs(ldcoefs,dwtlvn(nwl)))

!     Extract each set of coefficients, working from the deepest level
      Write (nout,99993)
      Do ilevel = nwl, 1, -1
        Write (nout,99992) ilevel, dwtlvm(nwl-ilevel+1), dwtlvn(nwl-ilevel+1)

        Do itype_coeffs = 0, 3
          Select Case (itype_coeffs)
          Case (0)
            If (ilevel==nwl) Then
              Write (nout,99991) 'Approximation coefficients '
            End If
          Case (1)
            Write (nout,99991) 'Vertical coefficients      '
          Case (2)
            Write (nout,99991) 'Horizontal coefficients    '
          Case (3)
            Write (nout,99991) 'Diagonal coefficients      '
          End Select
          If (itype_coeffs>0 .Or. ilevel==nwl) Then
!           Call the 2D extraction routine c09eaf
            Call c09eyf(ilevel,itype_coeffs,lenc,c,coefs,ldcoefs,icomm,ifail)
            Do i1 = 1, dwtlvm(nwl-ilevel+1)
              Write (nout,99989)(coefs(i1,j1),j1=1,dwtlvn(nwl-ilevel+1))
            End Do
          End If
        End Do
      End Do

      nwlinv = nwl

!     Reconstruct original data
      ifail = 0
      Call c09edf(nwlinv,lenc,c,m,n,b,ldb,icomm,ifail)

      Write (nout,99990)
      Do i = 1, m
        Write (nout,99998) b(i,1:n)
      End Do

99999 Format (1X,' MLDWT :: Wavelet  : ',A,/,1X,'          End mode : ',A,/,   &
        1X,'          M        : ',I4,/,1X,'          N        : ',I4,/)
99998 Format (8(F8.4,1X),:)
99997 Format (/,1X,' Number of Levels : ',I10)
99996 Format (1X,' Number of coefficients in 1st dimension for each level :')
99995 Format (8(I8,1X),:)
99994 Format (1X,' Number of coefficients in 2nd dimension for each level :')
99993 Format (/,1X,' Wavelet coefficients C : ')
99992 Format (1X,55('-'),/,1X,' Level : ',I10,'; output is ',I10,' by ',I10,/, &
        1X,55('-'))
99991 Format (1X,A28,': ')
99990 Format (/,1X,' Reconstruction           B : ')
99989 Format (4X,5(F8.4,1X),:)
    End Program c09ecfe
```

10.2 Program Data

```
C09ECF Example Program Data
 7, 8                              : m, n
 DB2 Half : wavnam, mode
  3.0000   7.0000   9.0000   1.0000   9.0000   9.0000   1.0000   0.0000
  9.0000   9.0000   3.0000   3.0000   4.0000   1.0000   2.0000   4.0000
  7.0000   8.0000   1.0000   3.0000   8.0000   9.0000   3.0000   3.0000
  1.0000   1.0000   1.0000   1.0000   2.0000   8.0000   4.0000   0.0000
  1.0000   2.0000   4.0000   6.0000   5.0000   6.0000   5.0000   4.0000
  2.0000   2.0000   5.0000   7.0000   3.0000   6.0000   6.0000   8.0000
  7.0000   9.0000   3.0000   1.0000   3.0000   4.0000   7.0000   2.0000
```

10.3 Program Results

```
 C09ECF Example Program Results

  MLDWT :: Wavelet  : DB2
           End mode : Half
           M        :    7
           N        :    8

  Input Data     A :
  3.0000   7.0000   9.0000   1.0000   9.0000   9.0000   1.0000   0.0000
  9.0000   9.0000   3.0000   3.0000   4.0000   1.0000   2.0000   4.0000
  7.0000   8.0000   1.0000   3.0000   8.0000   9.0000   3.0000   3.0000
  1.0000   1.0000   1.0000   1.0000   2.0000   8.0000   4.0000   0.0000
  1.0000   2.0000   4.0000   6.0000   5.0000   6.0000   5.0000   4.0000
  2.0000   2.0000   5.0000   7.0000   3.0000   6.0000   6.0000   8.0000
  7.0000   9.0000   3.0000   1.0000   3.0000   4.0000   7.0000   2.0000

  Number of Levels :          2
  Number of coefficients in 1st dimension for each level :
       4       5
  Number of coefficients in 2nd dimension for each level :
       4       5

  Wavelet coefficients C :
 -------------------------------------------------------
  Level :          2; output is         4 by         4
 -------------------------------------------------------
  Approximation coefficients :
     24.9724  25.6017  20.8900   7.9280
     27.6100  27.0955  18.7941   8.2804
     11.2663  11.0273  19.6410  18.6651
     27.6050  26.6443  14.5913  18.0835
  Vertical coefficients      :
     -2.5552  -6.1078  -4.0629   8.2136
     -1.6061  -7.2355  -3.3633   7.6075
     -0.2225  -1.6283  -0.5301   3.7415
     -0.9052  -6.5810   0.8023   1.8591
  Horizontal coefficients    :
     -3.8069  -3.0730   2.1121  -1.8525
     -2.7548  -4.5949  -0.8321  -4.8155
      4.8398   4.5104  -1.5308  -0.6456
     -6.4332  -4.5381   2.4753   6.8224
  Diagonal coefficients      :
     -0.8978  -0.2326  -1.2515   2.6346
      0.5708  -4.9783  -1.5309   6.4569
     -0.1854  -1.8430   0.2426  -0.0754
      0.0345   7.1864   1.5938  -5.9745
 -------------------------------------------------------
  Level :          1; output is         5 by         5
 -------------------------------------------------------
  Vertical coefficients      :
     -2.5981   4.6471   2.5392  -2.8415  -0.2165
     -1.3203  -0.0592   3.0490  -2.5837   1.0458
     -0.4330  -1.6405  -1.1752   0.2533  -2.3448
     -0.4118  -0.0682  -2.4608  -0.0167   0.4387
     -1.5368  -1.1450  -0.5547   4.5936  -3.6863
```

```
 Horizontal coefficients    :
    -4.3301  -1.8170   0.8023   5.7566  -2.8146
     4.3089   3.6908   0.8349   3.4653   1.7108
    -1.5311  -1.0736   1.5257   0.0212  -0.9608
     2.8873   3.1148  -1.9118  -0.4007  -1.5302
    -2.2377  -2.7611   2.4453  -0.3705   4.3448
 Diagonal coefficients      :
    -1.5000   4.4151  -0.0057  -0.8236  -1.1250
    -0.1953  -2.9530   1.8840  -1.7635   0.9877
    -0.4330   0.2745   1.1450   0.4632  -0.5547
    -0.3538  -0.3215   0.6462   1.3705  -1.2778
     0.7288   0.4587  -1.8873  -1.8828   2.4028

 Reconstruction           B :
 3.0000   7.0000   9.0000   1.0000   9.0000   9.0000   1.0000   0.0000
 9.0000   9.0000   3.0000   3.0000   4.0000   1.0000   2.0000   4.0000
 7.0000   8.0000   1.0000   3.0000   8.0000   9.0000   3.0000   3.0000
 1.0000   1.0000   1.0000   1.0000   2.0000   8.0000   4.0000   0.0000
 1.0000   2.0000   4.0000   6.0000   5.0000   6.0000   5.0000   4.0000
 2.0000   2.0000   5.0000   7.0000   3.0000   6.0000   6.0000   8.0000
 7.0000   9.0000   3.0000   1.0000   3.0000   4.0000   7.0000   2.0000
```