NAG Library Routine Document

C09DCF

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

C09DCF computes the one-dimensional multi-level maximal overlap discrete wavelet transform (MODWT). The initialization routine C09AAF must be called first to set up the MODWT options.

2 Specification

```
SUBROUTINE C09DCF (N, X, KEEPA, LENC, C, NWL, NA, ICOMM, IFAIL)
INTEGER            N, LENC, NWL, NA, ICOMM(100), IFAIL
REAL (KIND=nag_wp) X(N), C(LENC)
CHARACTER(1)       KEEPA
```

3 Description

C09DCF computes the multi-level MODWT for a data set, x_i, for $i = 1, 2, \ldots, n$, in one dimension. For a chosen number of levels, n_l, with $n_l \leq l_{\max}$, where $l_{\max}$ is returned by the initialization routine C09AAF in NWLMAX, the transform is returned as a set of coefficients for the different levels stored in a single array. Periodic reflection is currently the only available end extension method to reduce the edge effects caused by finite data sets.

The argument KEEPA can be set to retain both approximation and detail coefficients at each level resulting in $n_l \times (n_a + n_d)$ coefficients being returned in the output array, C, where n_a is the number of approximation coefficients and n_d is the number of detail coefficients. Otherwise, only the detail coefficients are stored for each level along with the approximation coefficients for the final level, in which case the length of the output array, C, is $n_a + n_l \times n_d$. In the present implementation, for simplicity, n_a and n_d are chosen to be equal by adding zero padding to the wavelet filters where necessary.

4 References

Percival D B and Walden A T (2000) *Wavelet Methods for Time Series Analysis* Cambridge University Press

5 Arguments

1: N – INTEGER *Input*

On entry: the number of elements, n, in the data array x.

Constraint: this must be the same as the value N passed to the initialization routine C09AAF.

2: X(N) – REAL (KIND=nag_wp) array *Input*

On entry: X contains the input dataset x_i, for $i = 1, 2, \ldots, n$.

3: KEEPA – CHARACTER(1) *Input*

On entry: determines whether the approximation coefficients are stored in array C for every level of the computed transform or else only for the final level. In both cases, the detail coefficients are stored in C for every level computed.

KEEPA = 'A'
Retain approximation coefficients for all levels computed.

KEEPA = 'F'
Retain approximation coefficients for only the final level computed.

Constraint: KEEPA = 'A' or 'F'.

4: LENC – INTEGER *Input*

On entry: the dimension of the array C as declared in the (sub)program from which C09DCF is called. C must be large enough to contain the number of wavelet coefficients.

If KEEPA = 'F', the total number of coefficients, n_c, is returned in NWC by the call to the initialization routine C09AAF and corresponds to the MODWT being continued for the maximum number of levels possible for the given data set. When the number of levels, n_l, is chosen to be less than the maximum, then the number of stored coefficients is correspondingly smaller and LENC can be reduced by noting that n_d detail coefficients are stored at each level, with the storage increased at the final level to contain the n_a approximation coefficients.

If KEEPA = 'A', n_d detail coefficients and n_a approximation coefficients are stored for each level computed, requiring LENC $\geq n_l \times (n_a + n_d) = 2 \times n_l \times n_a$, since the numbers of stored approximation and detail coefficients are equal. The number of approximation (or detail) coefficients at each level, n_a, is returned in NA.

Constraints:

if KEEPA = 'F', LENC $\geq (n_l + 1) \times n_a$;
if KEEPA = 'A', LENC $\geq 2 \times n_l \times n_a$.

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

On exit: the coefficients of a multi-level wavelet transform of the dataset.

The coefficients are stored in C as follows:

If KEEPA = 'F',

C$(1 : n_a)$
Contains the level n_l approximation coefficients;

C$(n_a + (i-1) \times n_d + 1 : n_a + i \times n_d)$
Contains the level $(n_l - i + 1)$ detail coefficients, for $i = 1, 2, \ldots, n_l$;

If KEEPA = 'A',

C$((i-1) \times n_a + 1 : i \times n_a)$
Contains the level $(n_l - i + 1)$ approximation coefficients, for $i = 1, 2, \ldots, n_l$;

C$(n_l \times n_a + (i-1) \times n_d + 1 : n_l \times n_a + i \times n_d)$
Contains the level i detail coefficients, for $i = 1, 2, \ldots, n_l$;

The values n_a and n_d denote the numbers of approximation and detail coefficients respectively, which are equal and returned in NA.

6: NWL – INTEGER *Input*

On entry: the number of levels, n_l, in the multi-level resolution to be performed.

Constraint: $1 \leq$ 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 C09AAF.

7: NA – INTEGER *Output*

On exit: NA contains the number of approximation coefficients, n_a, at each level which is equal to the number of detail coefficients, n_d. With periodic end extension (MODE = 'P' in C09AAF) this is the same as the length, N, of the data array, X.

8: ICOMM(100) – 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 C09AAF.

On exit: contains additional information on the computed transform.

9: 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, N is inconsistent with the value passed to the initialization routine: N = ⟨*value*⟩, N should be ⟨*value*⟩.

IFAIL = 2

On entry, KEEPA = ⟨*value*⟩ was an illegal value.

IFAIL = 4

On entry, LENC is set too small: LENC = ⟨*value*⟩.
Constraint: LENC $\geq$ ⟨*value*⟩.

IFAIL = 6

On entry, NWL = ⟨*value*⟩.
Constraint: NWL ≥ 1.

On entry, NWL is larger than the maximum number of levels returned by the initialization function: NWL = ⟨*value*⟩, maximum = ⟨*value*⟩.

IFAIL = 8

On entry, the initialization routine C09AAF has not been called first or it has not been called with WTRANS = 'U', or the communication array ICOMM has become 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

C09DCF 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 NA on exit.

10 Example

A set of data values (N = 64) is decomposed using the MODWT over two levels and then the inverse (C09DDF) is applied to restore the original data set.

10.1 Program Text

```
    Program c09dcfe

!     C09DCF Example Program Text

!     Mark 26 Release. NAG Copyright 2016.

!     .. Use Statements ..
      Use nag_library, Only: c09aaf, c09dcf, c09ddf, nag_wp
!     .. Implicit None Statement ..
      Implicit None
!     .. Parameters ..
      Integer, Parameter               :: nin = 5, nout = 6
!     .. Local Scalars ..
      Integer                          :: ifail, lenc, n, na, nf, nwc, nwl, ny
      Character (10)                   :: keepa, mode, wavnam, wtrans
!     .. Local Arrays ..
      Real (Kind=nag_wp), Allocatable  :: c(:), x(:), y(:)
      Integer                          :: icomm(100)
!     .. Executable Statements ..
      Write (nout,*) 'C09DCF Example Program Results'
!     Skip heading in data file
      Read (nin,*)
!     Read problem parameters
      Read (nin,*) n
      Read (nin,*) wavnam, mode, keepa
      Allocate (x(n),y(n))

      Write (nout,99999) wavnam, mode, n
      Write (nout,99998) keepa

!     Read data array and write it out
      Read (nin,*) x(1:n)
```

```
      Write (nout,*) ' Input Data      X :'
      Write (nout,99997) x(1:n)

!     Query wavelet filter dimensions
!     For Multi-Resolution Analysis, decomposition, wtrans = 'U'
      wtrans = 'U'

!     ifail: behaviour on error exit
!            =0 for hard exit, =1 for quiet-soft, =-1 for noisy-soft
      ifail = 0
      Call c09aaf(wavnam,wtrans,mode,n,nwl,nf,nwc,icomm,ifail)
!     Choose to decompose over two levels
      nwl = 2
      nwc = (nwl+1)*n
      If (keepa=='Final') Then
        lenc = nwc
      Else
        lenc = 2*n*nwl
      End If

      Allocate (c(lenc))

!     Perform Discrete Wavelet transform
      ifail = 0
      Call c09dcf(n,x,keepa,lenc,c,nwl,na,icomm,ifail)

      Write (nout,99996) nwl
      Write (nout,99995) na
      Write (nout,99994)
      Write (nout,99997) c(1:lenc)

!     Reconstruct original data
      ny = n

      ifail = 0
      Call c09ddf(nwl,keepa,lenc,c,ny,y,icomm,ifail)

      Write (nout,99993)
      Write (nout,99997) y(1:ny)

99999 Format (1X,' MLMODWT :: Wavelet : ',A10,', End mode : ',A10,' N = ',I10)
99998 Format (1X,'          :: Keepa   : ',A10)
99997 Format (8(F8.4,1X),:)
99996 Format (1X,' Number of Levels : ',I10)
99995 Format (1X,' Number of coefficients in each level : ',I10)
99994 Format (1X,' Wavelet coefficients C : ')
99993 Format (1X,' Reconstruction              Y : ')
    End Program c09dcfe
```

10.2 Program Data

```
C09DCF Example Program Data
 64                                  : n
 DB4  Periodic  All                  : wavnam, mode, keepa
 6.5271 6.512  6.5016 6.5237 6.4625
 6.3496 6.4025 6.4035 6.4407 6.4746
 6.5095 6.6551 6.61   6.5969 6.6083
 6.652  6.7113 6.7227 6.7196 6.7649
 6.7794 6.8037 6.8308 6.7712 6.7067
 6.769  6.7068 6.7024 6.6463 6.6098
 6.59   6.596  6.5457 6.547  6.5797
 6.5895 6.6275 6.6795 6.6598 6.6925
 6.6873 6.7223 6.7205 6.6843 6.703
 6.647  6.6008 6.6061 6.6097 6.6485
 6.6394 6.6571 6.6357 6.6224 6.6073
 6.6075 6.6379 6.6294 6.5906 6.6258
 6.6369 6.6515 6.6826 6.7042         : x(1:n)
```

10.3 Program Results

```
 C09DCF Example Program Results
  MLMODWT :: Wavelet : DB4       , End mode : Periodic   N =         64
          :: Keepa   : All
  Input Data      X :
  6.5271   6.5120   6.5016   6.5237   6.4625   6.3496   6.4025   6.4035
  6.4407   6.4746   6.5095   6.6551   6.6100   6.5969   6.6083   6.6520
  6.7113   6.7227   6.7196   6.7649   6.7794   6.8037   6.8308   6.7712
  6.7067   6.7690   6.7068   6.7024   6.6463   6.6098   6.5900   6.5960
  6.5457   6.5470   6.5797   6.5895   6.6275   6.6795   6.6598   6.6925
  6.6873   6.7223   6.7205   6.6843   6.7030   6.6470   6.6008   6.6061
  6.6097   6.6485   6.6394   6.6571   6.6357   6.6224   6.6073   6.6075
  6.6379   6.6294   6.5906   6.6258   6.6369   6.6515   6.6826   6.7042
  Number of Levels :          2
  Number of coefficients in each level :         64
  Wavelet coefficients C :
  6.6448   6.6505   6.6415   6.6090   6.5631   6.5119   6.4657   6.4371
  6.4162   6.4041   6.4062   6.4235   6.4652   6.5191   6.5744   6.6170
  6.6375   6.6496   6.6575   6.6741   6.7038   6.7335   6.7633   6.7849
  6.7939   6.7970   6.7868   6.7649   6.7407   6.7102   6.6814   6.6571
  6.6269   6.5993   6.5773   6.5598   6.5574   6.5688   6.5881   6.6173
  6.6492   6.6741   6.6941   6.7052   6.7078   6.7083   6.7001   6.6842
  6.6616   6.6338   6.6146   6.6072   6.6139   6.6306   6.6428   6.6459
  6.6384   6.6252   6.6147   6.6113   6.6143   6.6189   6.6264   6.6361
  6.6719   6.5883   6.4958   6.4890   6.5103   6.4695   6.3900   6.3656
  6.4065   6.4444   6.4727   6.5273   6.6057   6.6409   6.6102   6.6001
  6.6469   6.7019   6.7288   6.7330   6.7501   6.7824   6.8064   6.8147
  6.7846   6.7332   6.7239   6.7297   6.6971   6.6508   6.6127   6.5897
  6.5818   6.5636   6.5476   6.5657   6.5980   6.6284   6.6627   6.6803
  6.6821   6.6941   6.7131   6.7182   6.7020   6.6824   6.6562   6.6140
  6.5942   6.6126   6.6378   6.6502   6.6498   6.6403   6.6233   6.6086
  6.6099   6.6260   6.6300   6.6112   6.6094   6.6358   6.6581   6.6778
  0.0107   0.0084   0.0003  -0.0065  -0.0000   0.0196   0.0191  -0.0152
 -0.0369  -0.0291  -0.0131   0.0227   0.0461   0.0005  -0.0488  -0.0145
  0.0518   0.0503  -0.0038  -0.0243  -0.0087  -0.0111  -0.0316  -0.0191
  0.0323   0.0461  -0.0001  -0.0300  -0.0107   0.0164   0.0112  -0.0156
 -0.0225  -0.0091   0.0090   0.0244   0.0050  -0.0281  -0.0150   0.0146
  0.0145   0.0034  -0.0019   0.0058   0.0188   0.0074  -0.0133  -0.0127
 -0.0062  -0.0008   0.0077   0.0022  -0.0151  -0.0192  -0.0041   0.0091
  0.0136   0.0230   0.0203  -0.0081  -0.0274  -0.0179  -0.0013   0.0074
 -0.0150   0.0126   0.0048  -0.0276  -0.0227   0.0639  -0.0184  -0.0048
 -0.0303   0.0180   0.0327  -0.0343   0.0119  -0.0046   0.0167   0.0025
 -0.0524   0.0369   0.0029   0.0055  -0.0070  -0.0134   0.0099   0.0088
 -0.0095   0.0103  -0.0114  -0.0181   0.0269   0.0132  -0.0371   0.0250
 -0.0186   0.0138   0.0022  -0.0058  -0.0112   0.0207  -0.0058  -0.0054
  0.0115  -0.0089  -0.0106   0.0180  -0.0096   0.0107  -0.0156   0.0068
  0.0074  -0.0242   0.0169   0.0075  -0.0045   0.0031  -0.0108   0.0092
 -0.0115   0.0061  -0.0002   0.0078  -0.0012  -0.0168   0.0074   0.0157
  Reconstruction              Y :
  6.5271   6.5120   6.5016   6.5237   6.4625   6.3496   6.4025   6.4035
  6.4407   6.4746   6.5095   6.6551   6.6100   6.5969   6.6083   6.6520
  6.7113   6.7227   6.7196   6.7649   6.7794   6.8037   6.8308   6.7712
  6.7067   6.7690   6.7068   6.7024   6.6463   6.6098   6.5900   6.5960
  6.5457   6.5470   6.5797   6.5895   6.6275   6.6795   6.6598   6.6925
  6.6873   6.7223   6.7205   6.6843   6.7030   6.6470   6.6008   6.6061
  6.6097   6.6485   6.6394   6.6571   6.6357   6.6224   6.6073   6.6075
  6.6379   6.6294   6.5906   6.6258   6.6369   6.6515   6.6826   6.7042
```