NAG Library Function Document

nag_idwt_2d (c09ebc)

1 Purpose

nag_idwt_2d (c09ebc) computes the inverse two-dimensional discrete wavelet transform (DWT) at a single level. The initialization function nag_wfilt_2d (c09abc) must be called first to set up the DWT options.

2 Specification

```
#include <nag.h>
#include <nagc09.h>
```

```
void nag_idwt_2d (Integer m, Integer n, const double ca[], Integer ldca,
     const double ch[], Integer ldch, const double cv[], Integer ldcv,
     const double cd[], Integer ldcd, double b[], Integer ldb,
     const Integer icomm[], NagError *fail)
```

3 Description

nag_idwt_2d (c09ebc) performs the inverse operation of function nag_dwt_2d (c09eac). That is, given sets of approximation, horizontal, vertical and diagonal coefficients computed by function nag_dwt_2d (c09eac) using a DWT as set up by the initialization function nag_wfilt_2d (c09abc), on a real matrix, B, nag_idwt_2d (c09ebc) will reconstruct B.

4 References

None.

5 Arguments

1: **m** – Integer *Input*

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

Constraint: this must be the same as the value **m** passed to the initialization function nag_wfilt_2d (c09abc).

2: **n** – Integer *Input*

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

Constraint: this must be the same as the value **n** passed to the initialization function nag_wfilt_2d (c09abc).

3: **ca**[dim] – const double *Input*

Note: the dimension, *dim*, of the array **ca** must be at least **ldca** $\times n_{\mathrm{cn}}$ where n_{cn} is the argument **nwcn** returned by function nag_wfilt_2d (c09abc).

The (i,j)th element of the matrix is stored in **ca**$[(j-1)\times$ **ldca** $+ i - 1]$.

On entry: contains the n_{cm} by n_{cn} matrix of approximation coefficients, C_a. This array will normally be the result of some transformation on the coefficients computed by function nag_dwt_2d (c09eac).

4: **ldca** – Integer *Input*

On entry: the stride separating matrix row elements in the array **ca**.

Constraint: **ldca** $\geq n_{\text{cm}}$ where $n_{\text{cm}} = n_{\text{ct}}/(4n_{\text{cn}})$ and n_{cn}, n_{ct} are returned by the initialization function nag_wfilt_2d (c09abc).

5: **ch**[*dim*] – const double *Input*

Note: the dimension, *dim*, of the array **ch** must be at least **ldch** $\times\, n_{\text{cn}}$ where n_{cn} is the argument **nwcn** returned by function nag_wfilt_2d (c09abc).

The (i,j)th element of the matrix is stored in **ch**$[(j-1)\times$ **ldch** $+\, i-1]$.

On entry: contains the n_{cm} by n_{cn} matrix of horizontal coefficients, C_h. This array will normally be the result of some transformation on the coefficients computed by function nag_dwt_2d (c09eac).

6: **ldch** – Integer *Input*

On entry: the stride separating matrix row elements in the array **ch**.

Constraint: **ldch** $\geq n_{\text{cm}}$ where $n_{\text{cm}} = n_{\text{ct}}/(4n_{\text{cn}})$ and n_{cn}, n_{ct} are returned by the initialization function nag_wfilt_2d (c09abc).

7: **cv**[*dim*] – const double *Input*

Note: the dimension, *dim*, of the array **cv** must be at least **ldcv** $\times\, n_{\text{cn}}$ where n_{cn} is the argument **nwcn** returned by function nag_wfilt_2d (c09abc).

The (i,j)th element of the matrix is stored in **cv**$[(j-1)\times$ **ldcv** $+\, i-1]$.

On entry: contains the n_{cm} by n_{cn} matrix of vertical coefficients, C_v. This array will normally be the result of some transformation on the coefficients computed by function nag_dwt_2d (c09eac).

8: **ldcv** – Integer *Input*

On entry: the stride separating matrix row elements in the array **cv**.

Constraint: **ldcv** $\geq n_{\text{cm}}$ where $n_{\text{cm}} = n_{\text{ct}}/(4n_{\text{cn}})$ and n_{cn}, n_{ct} are returned by the initialization function nag_wfilt_2d (c09abc).

9: **cd**[*dim*] – const double *Input*

Note: the dimension, *dim*, of the array **cd** must be at least **ldcd** $\times\, n_{\text{cn}}$ where n_{cn} is the argument **nwcn** returned by function nag_wfilt_2d (c09abc).

The (i,j)th element of the matrix is stored in **cd**$[(j-1)\times$ **ldcd** $+\, i-1]$.

On entry: contains the n_{cm} by n_{cn} matrix of diagonal coefficients, C_d. This array will normally be the result of some transformation on the coefficients computed by function nag_dwt_2d (c09eac).

10: **ldcd** – Integer *Input*

On entry: the stride separating matrix row elements in the array **cd**.

Constraint: **ldcd** $\geq n_{\text{cm}}$ where $n_{\text{cm}} = n_{\text{ct}}/(4n_{\text{cn}})$ and n_{cn}, n_{ct} are returned by the initialization function nag_wfilt_2d (c09abc).

11: **b**[**ldb** × **n**] – double *Output*

Note: the (i,j)th element of the matrix B is stored in **b**$[(j-1)\times$ **ldb** $+\, i-1]$.

On exit: the m by n reconstructed matrix, B, based on the input approximation, horizontal, vertical and diagonal coefficients and the transform options supplied to the initialization function nag_wfilt_2d (c09abc).

12: **ldb** – Integer *Input*

On entry: the stride separating matrix row elements in the array **b**.

Constraint: **ldb** $\geq$ **m**.

13: **icomm**[**180**] – const Integer *Communication Array*

On entry: contains details of the discrete wavelet transform and the problem dimension as setup in the call to the initialization function nag_wfilt_2d (c09abc).

14: **fail** – NagError * *Input/Output*

The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

NE_ALLOC_FAIL

Dynamic memory allocation failed.
See Section 3.2.1.2 in the Essential Introduction for further information.

NE_BAD_PARAM

On entry, argument $\langle$*value*$\rangle$ had an illegal value.

NE_INITIALIZATION

Either the initialization function has not been called first or **icomm** has been corrupted.

Either the initialization function was called with **wtrans** = Nag_MultiLevel or **icomm** has been corrupted.

NE_INT

On entry, **ldca** = $\langle$*value*$\rangle$.
Constraint: **ldca** $\geq$ $\langle$*value*$\rangle$, the number of wavelet coefficients in the first dimension.

On entry, **ldcd** = $\langle$*value*$\rangle$.
Constraint: **ldcd** $\geq$ $\langle$*value*$\rangle$, the number of wavelet coefficients in the first dimension.

On entry, **ldch** = $\langle$*value*$\rangle$.
Constraint: **ldch** $\geq$ $\langle$*value*$\rangle$, the number of wavelet coefficients in the first dimension.

On entry, **ldcv** = $\langle$*value*$\rangle$.
Constraint: **ldcv** $\geq$ $\langle$*value*$\rangle$, the number of wavelet coefficients in the first dimension.

On entry, **m** = $\langle$*value*$\rangle$.
Constraint: **m** = $\langle$*value*$\rangle$, the value of **m** on initialization (see nag_wfilt_2d (c09abc)).

On entry, **n** = $\langle$*value*$\rangle$.
Constraint: **n** = $\langle$*value*$\rangle$, the value of **n** on initialization (see nag_wfilt_2d (c09abc)).

NE_INT_2

On entry, **ldb** = $\langle$*value*$\rangle$ and **m** = $\langle$*value*$\rangle$.
Constraint: **ldb** $\geq$ **m**.

NE_INTERNAL_ERROR

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.

An unexpected error has been triggered by this function. Please contact NAG.
See Section 3.6.6 in the Essential Introduction for further information.

NE_NO_LICENCE

Your licence key may have expired or may not have been installed correctly.
See Section 3.6.5 in the Essential Introduction 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

nag_idwt_2d (c09ebc) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this function. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

None.

10 Example

See Section 10 in nag_dwt_2d (c09eac).