NAG Library Function Document

nag_idwt (c09cbc)

1 Purpose

nag_idwt (c09cbc) computes the inverse one-dimensional discrete wavelet transform (DWT) at a single level. The initialization function nag_wfilt (c09aac) must be called first to set up the DWT options.

2 Specification

```
#include <nag.h>
#include <nagc09.h>
void nag_idwt (Integer lenc, const double ca[], const double cd[], Integer n,
     double y[], const Integer icomm[], NagError *fail)
```

3 Description

nag_idwt (c09cbc) performs the inverse operation of nag_dwt (c09cac). That is, given sets of n_c approximation coefficients and detail coefficients, computed by nag_dwt (c09cac) using a DWT as set up by the initialization function nag_wfilt (c09aac), on a real data array of length n, nag_idwt (c09cbc) will reconstruct the data array y_i, for $i = 1, 2, \ldots, n$, from which the coefficients were derived.

4 References

None.

5 Arguments

1: **lenc** – Integer *Input*

On entry: the dimension of the arrays **ca** and **cd**.

Constraint: **lenc** $\geq n_c$, where n_c is the value returned in **nwc** by the call to the initialization function nag_wfilt (c09aac).

2: **ca**[**lenc**] – const double *Input*

On entry: the n_c approximation coefficients, C_a. These will normally be the result of some transformation on the coefficients computed by nag_dwt (c09cac).

3: **cd**[**lenc**] – const double *Input*

On entry: the n_c detail coefficients, C_d. These will normally be the result of some transformation on the coefficients computed by nag_dwt (c09cac).

4: **n** – Integer *Input*

On entry: n, the length of the original data array from which the wavelet coefficients were computed by nag_dwt (c09cac) and the length of the data array **y** that is to be reconstructed by this function.

Constraint: This must be the same as the value **n** passed to the initialization function nag_wfilt (c09aac).

5: **y**[**n**] – double *Output*

On exit: the reconstructed data based on approximation and detail coefficients C_a and C_d and the transform options supplied to the initialization function nag_wfilt (c09aac).

6: **icomm**[**100**] – const Integer *Communication Array*

On entry: contains details of the discrete wavelet transform and the problem dimension and, possibly, additional information on the previously computed forward transform.

7: **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_ARRAY_DIM_LEN

On entry, array dimension **lenc** not large enough: **lenc** = ⟨*value*⟩ but must be at least ⟨*value*⟩.

NE_BAD_PARAM

On entry, argument ⟨*value*⟩ had an illegal value.

NE_INITIALIZATION

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

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

On entry, **n** is inconsistent with the value passed to the initialization function: **n** = ⟨*value*⟩, **n** should be ⟨*value*⟩.

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

Not applicable.

9 Further Comments

None.

10 Example

See Section 10 in nag_dwt (c09cac).