

# NAG Library Function Document

## nag\_wav\_3d\_coeff\_ext (c09fyc)

### 1 Purpose

nag\_wav\_3d\_coeff\_ext (c09fyc) extracts a selected set of discrete wavelet transform (DWT) coefficients from the full set of coefficients stored in compact form, as computed by nag\_dwt\_3d (c09fac) (single level three-dimensional DWT) or nag\_mldwt\_3d (c09fcc) (multi-level three-dimensional DWT).

### 2 Specification

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

void nag_wav_3d_coeff_ext (Integer ilev, Integer cindex, Integer lenc,
    const double c[], double d[], Integer ldd, Integer sdd, Integer icomm[],
    NagError *fail)
```

### 3 Description

nag\_wav\_3d\_coeff\_ext (c09fyc) is intended to be used after a call to either nag\_dwt\_3d (c09fac) (single level three-dimensional DWT) or nag\_mldwt\_3d (c09fcc) (multi-level three-dimensional DWT), either of which must be preceded by a call to nag\_wfilt\_3d (c09acc) (three-dimensional wavelet filter initialization). Given an initial three-dimensional data set *A*, a prior call to nag\_dwt\_3d (c09fac) or nag\_mldwt\_3d (c09fcc) computes the approximation coefficients (at the highest requested level in the case of nag\_mldwt\_3d (c09fcc)) and seven sets of detail coefficients (at all levels in the case of nag\_mldwt\_3d (c09fcc)) and stores these in compact form in a one-dimensional array *c*. nag\_wav\_3d\_coeff\_ext (c09fyc) can then extract either the approximation coefficients or one of the sets of detail coefficients (at one of the levels following nag\_mldwt\_3d (c09fcc)) into a three-dimensional data set stored in *d*.

If a multi-level DWT was performed by a prior call to nag\_mldwt\_3d (c09fcc) then the dimensions of the three-dimensional data stored in *d* depend on the level extracted and are available from the arrays **dwtlvm**, **dwtlvn** and **dwtlvfr** as returned by nag\_mldwt\_3d (c09fcc) which contain the first, second and third dimensions respectively.

If a single level DWT was performed by a prior call to nag\_dwt\_3d (c09fac) then the dimensions of the three-dimensional data stored in *d* can be determined from **nwct**, **nwcn** and **nwcfr** as returned by the setup function nag\_wfilt\_3d (c09acc).

See Section 2.1 in the c09 Chapter Introduction for a discussion of the three-dimensional DWT.

### 4 References

None.

### 5 Arguments

**Note:** the following notation is used in this section:

$n_{cm}$  is the number of wavelet coefficients in the first dimension. Following a call to nag\_dwt\_3d (c09fac) (i.e., when **ilev** = 0) this is equal to **nwct**/(8 × **nwcn** × **nwcfr**) as returned by nag\_wfilt\_3d (c09acc). Following a call to nag\_mldwt\_3d (c09fcc) transforming **nwl** levels, and when extracting at level **ilev** > 0, this is equal to **dwtlvm**[**nwl** – **ilev**].

$n_{cn}$  is the number of wavelet coefficients in the second dimension. Following a call to nag\_dwt\_3d (c09fac) (i.e., when **ilev** = 0) this is equal to **nwcn** as returned by nag\_wfilt\_3d

(c09acc). Following a call to nag\_mldwt\_3d (c09fcc) transforming **nwl** levels, and when extracting at level **ilev** > 0, this is equal to **dwtlvn[nwl – ilev]**.

$n_{\text{cfr}}$  is the number of wavelet coefficients in the third dimension. Following a call to nag\_dwt\_3d (c09fac) (i.e., when **ilev** = 0) this is equal to **nwcfr** as returned by nag\_wfilt\_3d (c09acc). Following a call to nag\_mldwt\_3d (c09fcc) transforming **nwl** levels, and when extracting at level **ilev** > 0, this is equal to **dwtlvfr[nwl – ilev]**.

- 1: **ilev** – Integer *Input*

*On entry:* the level at which coefficients are to be extracted.

If **ilev** = 0, it is assumed that the coefficient array **c** was produced by a preceding call to the single level function nag\_dwt\_3d (c09fac).

If **ilev** > 0, it is assumed that the coefficient array **c** was produced by a preceding call to the multi-level function nag\_mldwt\_3d (c09fcc).

*Constraints:*

**ilev** = 0 (following a call to nag\_dwt\_3d (c09fac));  
 $0 \leq \text{ilev} \leq \text{nwl}$ , where **nwl** is as used in a preceding call to nag\_mldwt\_3d (c09fcc);  
 if **cindex** = 0, **ilev** = **nwl** (following a call to nag\_mldwt\_3d (c09fcc)).

- 2: **cindex** – Integer *Input*

*On entry:* identifies which coefficients to extract. The coefficients are identified as follows:

**cindex** = 0

The approximation coefficients, produced by application of the low pass filter over columns, rows and frames of *A* (LLL). After a call to the multi-level transform function nag\_mldwt\_3d (c09fcc) (which implies that **ilev** > 0) the approximation coefficients are available only for **ilev** = **nwl**, where **nwl** is the value used in a preceding call to nag\_mldwt\_3d (c09fcc).

**cindex** = 1

The detail coefficients produced by applying the low pass filter over columns and rows of *A* and the high pass filter over frames (LLH).

**cindex** = 2

The detail coefficients produced by applying the low pass filter over columns, high pass filter over rows and low pass filter over frames of *A* (LHL).

**cindex** = 3

The detail coefficients produced by applying the low pass filter over columns of *A* and high pass filter over rows and frames (LHH).

**cindex** = 4

The detail coefficients produced by applying the high pass filter over columns of *A* and low pass filter over rows and frames (HLL).

**cindex** = 5

The detail coefficients produced by applying the high pass filter over columns, low pass filter over rows and high pass filter over frames of *A* (HLH).

**cindex** = 6

The detail coefficients produced by applying the high pass filter over columns and rows of *A* and the low pass filter over frames (HHL).

**cindex** = 7

The detail coefficients produced by applying the high pass filter over columns, rows and frames of *A* (HHH).

*Constraints:*

if **ilev** = 0,  $0 \leq \mathbf{cindex} \leq 7$ ;  
 if **ilev** = **nwl**, following a call to nag\_mldwt\_3d (c09fcc) transforming **nwl** levels,  
 $0 \leq \mathbf{cindex} \leq 7$ ;  
 otherwise  $1 \leq \mathbf{cindex} \leq 7$ .

3: **lenc** – Integer *Input*

*On entry:* the dimension of the array **c**.

*Constraint:* **lenc** must be unchanged from the value used in the preceding call to either nag\_dwt\_3d (c09fac) or nag\_mldwt\_3d (c09fcc)..

4: **c[lenc]** – const double *Input*

*On entry:* DWT coefficients, as computed by nag\_dwt\_3d (c09fac) or nag\_mldwt\_3d (c09fcc).

5: **d[dim]** – double *Output*

**Note:** the dimension, *dim*, of the array **d** must be at least  $\mathbf{ldd} \times \mathbf{sdd} \times n_{\text{cfr}}$ .

*On exit:* the requested coefficients.

If the DWT coefficients were computed by nag\_dwt\_3d (c09fac) then

if **cindex** = 0, the approximation coefficients are stored in **d**[(*k* - 1) × **ldd** × **sdd** + (*j* - 1) × **ldd** + *i* - 1], for *i* = 1, 2, ..., *n<sub>cm</sub>*, *j* = 1, 2, ..., *n<sub>cn</sub>* and *k* = 1, 2, ..., *n<sub>cfr</sub>*;

if  $1 \leq \mathbf{cindex} \leq 7$ , the detail coefficients, as indicated by **cindex**, are stored in **d**[(*k* - 1) × **ldd** × **sdd** + (*j* - 1) × **ldd** + *i* - 1], for *i* = 1, 2, ..., *n<sub>cm</sub>*, *j* = 1, 2, ..., *n<sub>cn</sub>* and *k* = 1, 2, ..., *n<sub>cfr</sub>*.

If the DWT coefficients were computed by nag\_mldwt\_3d (c09fcc) then

if **cindex** = 0 and **ilev** = **nwl**, the approximation coefficients are stored in **d**[(*k* - 1) × **ldd** × **sdd** + (*j* - 1) × **ldd** + *i* - 1], for *i* = 1, 2, ..., *n<sub>cm</sub>*, *j* = 1, 2, ..., *n<sub>cn</sub>* and *k* = 1, 2, ..., *n<sub>cfr</sub>*;

if  $1 \leq \mathbf{cindex} \leq 7$ , the detail coefficients, as indicated by **cindex**, for level **ilev** are stored in **d**[(*k* - 1) × **ldd** × **sdd** + (*j* - 1) × **ldd** + *i* - 1], for *i* = 1, 2, ..., *n<sub>cm</sub>*, *j* = 1, 2, ..., *n<sub>cn</sub>* and *k* = 1, 2, ..., *n<sub>cfr</sub>*.

6: **ldd** – Integer *Input*

*On entry:* the stride separating row elements of each of the sets of frame coefficients in the three-dimensional data stored in **d**.

*Constraint:* **ldd** ≥ *n<sub>cm</sub>*.

7: **sdd** – Integer *Input*

*On entry:* the stride separating corresponding coefficients of consecutive frames in the three-dimensional data stored in **d**.

*Constraint:* **sdd** ≥ *n<sub>cn</sub>*.

8: **icomm[260]** – 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\_3d (c09acc).

9: **fail** – NagError \* *Input/Output*

The NAG error argument (see Section 2.7 in How to Use the NAG Library and its Documentation).

## 6 Error Indicators and Warnings

### NE\_ALLOC\_FAIL

Dynamic memory allocation failed.

See Section 2.3.1.2 in How to Use the NAG Library and its Documentation 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.

### NE\_INT

On entry, **cindex** =  $\langle value \rangle$ .

Constraint: **cindex**  $\leq 7$ .

On entry, **cindex** =  $\langle value \rangle$ .

Constraint: **cindex**  $\geq 0$ .

On entry, **ilev** =  $\langle value \rangle$ .

Constraint: **ilev** = 0 following a call to the single level function nag\_dwt\_3d (c09fac).

On entry, **ilev** =  $\langle value \rangle$ .

Constraint: **ilev** > 0 following a call to the multi-level function nag\_mldwt\_3d (c09fcc).

### NE\_INT\_2

On entry, **ilev** =  $\langle value \rangle$  and **nwl** =  $\langle value \rangle$ .

Constraint: **ilev**  $\leq$  **nwl**, where **nwl** is the number of levels used in the call to nag\_mldwt\_3d (c09fcc).

On entry, **ldd** =  $\langle value \rangle$  and  $n_{cm}$  =  $\langle value \rangle$ .

Constraint: **ldd**  $\geq n_{cm}$ , where  $n_{cm}$  is the number of DWT coefficients in the first dimension following the single level transform.

On entry, **lenc** =  $\langle value \rangle$  and  $n_{ct}$  =  $\langle value \rangle$ .

Constraint: **lenc**  $\geq n_{ct}$ , where  $n_{ct}$  is the number of DWT coefficients computed in the preceding call to nag\_dwt\_3d (c09fac).

On entry, **lenc** =  $\langle value \rangle$  and  $n_{ct}$  =  $\langle value \rangle$ .

Constraint: **lenc**  $\geq n_{ct}$ , where  $n_{ct}$  is the number of DWT coefficients computed in the preceding call to nag\_mldwt\_3d (c09fcc).

On entry, **sdd** =  $\langle value \rangle$  and  $n_{cn}$  =  $\langle value \rangle$ .

Constraint: **sdd**  $\geq n_{cn}$ , where  $n_{cn}$  is the number of DWT coefficients in the second dimension following the single level transform.

### NE\_INT\_3

On entry, **ilev** =  $\langle value \rangle$  and **nwl** =  $\langle value \rangle$ , but **cindex** = 0.

Constraint: **cindex** > 0 when **ilev** < **nwl** in the preceding call to nag\_mldwt\_3d (c09fcc).

On entry, **ldd** =  $\langle value \rangle$  and  $n_{cm}$  =  $\langle value \rangle$ .

Constraint: **ldd**  $\geq n_{cm}$ , where  $n_{cm}$  is the number of DWT coefficients in the first dimension at the selected level **ilev**.

On entry, **sdd** =  $\langle value \rangle$  and  $n_{cn}$  =  $\langle value \rangle$ .

Constraint: **sdd**  $\geq n_{cn}$ , where  $n_{cn}$  is the number of DWT coefficients in the second dimension at the selected level **ilev**.

**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 2.7.6 in How to Use the NAG Library and its Documentation for further information.

**NE\_NO\_LICENCE**

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

See Section 2.7.5 in How to Use the NAG Library and its Documentation for further information.

**7 Accuracy**

Not applicable.

**8 Parallelism and Performance**

nag\_wav\_3d\_coeff\_ext (c09fyc) is not threaded in any implementation.

**9 Further Comments**

None.

**10 Example**

See Section 10 in nag\_wfilt\_3d (c09acc), nag\_dwt\_3d (c09fac), nag\_mldwt\_3d (c09fcc) and nag\_wav\_3d\_coeff\_ins (c09fzc).

---