NAG Library Function Document

nag_sum_fft_hermitian_3d (c06pzc)

1 Purpose

nag_sum_fft_hermitian_3d (c06pzc) computes the three-dimensional inverse discrete Fourier transform of a trivariate Hermitian sequence of complex data values.

2 Specification

```
#include <nag.h>
#include <nagc06.h>
void nag_sum_fft_hermitian_3d (Integer n1, Integer n2, Integer n3,
     const Complex y[], double x[], NagError *fail)
```

3 Description

nag_sum_fft_hermitian_3d (c06pzc) computes the three-dimensional inverse discrete Fourier transform of a trivariate Hermitian sequence of complex data values $z_{j_1 j_2 j_3}$, for $j_1 = 0, 1, \ldots, n_1 - 1$, $j_2 = 0, 1, \ldots, n_2 - 1$ and $j_3 = 0, 1, \ldots, n_3 - 1$.

The discrete Fourier transform is here defined by

$$\hat{x}_{k_1 k_2 k_3} = \frac{1}{\sqrt{n_1 n_2 n_3}} \sum_{j_1=0}^{n_1-1} \sum_{j_2=0}^{n_2-1} \sum_{j_3=0}^{n_3-1} z_{j_1 j_2 j_3} \times \exp\left(2\pi i \left(\frac{j_1 k_1}{n_1} + \frac{j_2 k_2}{n_2} + \frac{j_3 k_3}{n_3}\right)\right),$$

where $k_1 = 0, 1, \ldots, n_1 - 1$, $k_2 = 0, 1, \ldots, n_2 - 1$ and $k_3 = 0, 1, \ldots, n_3 - 1$. (Note the scale factor of $\frac{1}{\sqrt{n_1 n_2 n_3}}$ in this definition.)

Because the input data satisfies conjugate symmetry (i.e., $z_{k_1 k_2 k_3}$ is the complex conjugate of $z_{(n_1 - k_1) k_2 k_3}$), the transformed values $\hat{x}_{k_1 k_2 k_3}$ are real.

A call of nag_sum_fft_real_3d (c06pyc) followed by a call of nag_sum_fft_hermitian_3d (c06pzc) will restore the original data.

This function performs multiple one-dimensional discrete Fourier transforms by the fast Fourier transform (FFT) algorithm in Brigham (1974) and Temperton (1983).

4 References

Brigham E O (1974) *The Fast Fourier Transform* Prentice–Hall

Temperton C (1983) Fast mixed-radix real Fourier transforms *J. Comput. Phys.* **52** 340–350

5 Arguments

1: **n1** – Integer *Input*

On entry: n_1, the first dimension of the transform.

Constraint: $\mathbf{n1} \geq 1$.

2: **n2** – Integer *Input*

On entry: n_2, the second dimension of the transform.

Constraint: $\mathbf{n2} \geq 1$.

3: **n3** – Integer *Input*

On entry: n_3, the third dimension of the transform.

Constraint: **n3** ≥ 1.

4: **y**[*dim*] – const Complex *Input*

Note: the dimension, *dim*, of the array **y** must be at least $(\mathbf{n1}/2 + 1) \times \mathbf{n2} \times \mathbf{n3}$.

On entry: the Hermitian sequence of complex input dataset z, where $z_{j_1 j_2 j_3}$ is stored in $\mathbf{y}[j_3 \times (n_1/2 + 1)n_2 + j_2 \times (n_1/2 + 1) + j_1]$, for $j_1 = 0, 1, \ldots, n_1/2$, $j_2 = 0, 1, \ldots, n_2 - 1$ and $j_3 = 0, 1, \ldots, n_3 - 1$.

5: **x**[**n1** × **n2** × **n3**] – double *Output*

On exit: the real output dataset $\hat{x}$, where $\hat{x}_{k_1 k_2 k_3}$ is stored in $\mathbf{x}[k_3 \times n_1 n_2 + k_2 \times n_1 + k_1]$, for $k_1 = 0, 1, \ldots, n_1 - 1$, $k_2 = 0, 1, \ldots, n_2 - 1$ and $k_3 = 0, 1, \ldots, n_3 - 1$.

6: **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 3.2.1.2 in How to Use the NAG Library and its Documentation for further information.

NE_BAD_PARAM

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

NE_INT

On entry, **n1** = ⟨*value*⟩.
Constraint: **n1** ≥ 1.

On entry, **n2** = ⟨*value*⟩.
Constraint: **n2** ≥ 1.

On entry, **n3** = ⟨*value*⟩.
Constraint: **n3** ≥ 1.

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

7 Accuracy

Some indication of accuracy can be obtained by performing a forward transform using nag_sum_fft_real_3d (c06pyc) and a backward transform using nag_sum_fft_hermitian_3d (c06pzc), and comparing the results with the original sequence (in exact arithmetic they would be identical).

8 Parallelism and Performance

nag_sum_fft_hermitian_3d (c06pzc) is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

nag_sum_fft_hermitian_3d (c06pzc) makes calls to BLAS and/or LAPACK routines, which may be threaded within the vendor library used by this implementation. Consult the documentation for the vendor library for further information.

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' Notefor your implementation for any additional implementation-specific information.

9 Further Comments

The time taken by nag_sum_fft_hermitian_3d (c06pzc) is approximately proportional to $n_1 n_2 n_3 \log(n_1 n_2 n_3)$, but also depends on the factors of n_1, n_2 and n_3. nag_sum_fft_hermitian_3d (c06pzc) is fastest if the only prime factors of n_1, n_2 and n_3 are 2, 3 and 5, and is particularly slow if one of the dimensions is a large prime, or has large prime factors.

Workspace is internally allocated by nag_sum_fft_hermitian_3d (c06pzc). The total size of these arrays is approximately proportional to $n_1 n_2 n_3$.

10 Example

See Section 10 in nag_sum_fft_real_3d (c06pyc).