NAG C Library Function Document

nag_3d_shep_eval (e01thc)

1 Purpose

nag_3d_shep_eval (e01thc) evaluates the three-dimensional interpolating function generated by
nag_3d_shep_interp (e01tgc) and its first partial derivatives.

2 Specification

```c
void nag_3d_shep_eval (Integer m, const double x[], const double y[],
const double z[], const double f[], const Integer iq[], const double rq[],
Integer n, const double u[], const double v[], const double w[], double q[],
double qx[], double qy[], double qz[], NagError *fail)
```

3 Description

This function takes as input the interpolant \( Q(x, y, z) \) of a set of scattered data points \( (x_r, y_r, z_r, f_r) \), for
\( r = 1, 2, \ldots, m \), as computed by nag_3d_shep_interp (e01tgc), and evaluates the interpolant and its first
partial derivatives at the set of points \( (u_i, v_i, w_i) \), for \( i = 1, 2, \ldots, n \).

nag_3d_shep_eval (e01thc) must only be called after a call to nag_3d_shep_interp (e01tgc).

This function is derived from the function QS3GRD described by Renka (1988b).

4 References

Renka R J (1988b) Algorithm 661: QSHEP3D: Quadratic Shepard method for trivariate interpolation of
scattered data  ACM Trans. Math. Software 14 151–152

5 Parameters

1:  \( m \) – Integer  
   \( m = \) Integer  
   \( \text{Input} \)

2:  \( x[m] \) – const double  
   \( x[m] = \) const double  
   \( \text{Input} \)

3:  \( y[m] \) – const double  
   \( y[m] = \) const double  
   \( \text{Input} \)

4:  \( z[m] \) – const double  
   \( z[m] = \) const double  
   \( \text{Input} \)

5:  \( f[m] \) – const double  
   \( f[m] = \) const double  
   \( \text{Input} \)

   On entry: \( m, x, y, z \) and \( f \) must be the same values as were supplied in the preceding call to
nag_3d_shep_interp (e01tgc).

6:  \( iq[dim] \) – const Integer  
   \( iq[dim] = \) const Integer  
   \( \text{Input} \)

   Note: the dimension, \( dim \), of the array \( iq \) must be at least \( 2 \times m + 1 \).

   On entry: \( iq \) must be unchanged from the value returned from a previous call to nag_3d_shep_interp
(e01tgc).

7:  \( rq[dim] \) – const double  
   \( rq[dim] = \) const double  
   \( \text{Input} \)

   Note: the dimension, \( dim \), of the array \( rq \) must be at least \( 10 \times m + 7 \).

   On entry: \( rq \) must be unchanged from the value returned from a previous call to
nag_3d_shep_interp (e01tgc).

8:  \( n \) – Integer  
   \( n = \) Integer  
   \( \text{Input} \)

   On entry: \( n \), the number of evaluation points.

   Constraint: \( n \geq 1 \).
9:  \( u[n] \) – const double \hspace{1cm} \text{Input}
10: \( v[n] \) – const double \hspace{1cm} \text{Input}
11: \( w[n] \) – const double \hspace{1cm} \text{Input}

On entry: \( u[i-1], v[i-1], w[i-1] \) must be set to the evaluation point \((u_i, v_i, w_i)\), for \( i = 1, 2, \ldots, n \).

12: \( q[n] \) – double \hspace{1cm} \text{Output}

On exit: \( q[i-1] \) contains the value of the interpolant at \((u_i, v_i, w_i)\), for \( i = 1, 2, \ldots, n \). If any of these evaluation points lie outside the region of definition of the interpolant the corresponding entries in \( q \) are set to the largest machine representable number (see nag_real_largest_number (X02ALC)), and nag_3d_shep_eval (e01thc) returns with fail.code = NE_BAD_POINT.

13: \( qx[n] \) – double \hspace{1cm} \text{Output}
14: \( qy[n] \) – double \hspace{1cm} \text{Output}
15: \( qz[n] \) – double \hspace{1cm} \text{Output}

On exit: \( qx[i-1], qy[i-1], qz[i-1] \) contains the value of the partial derivatives of the interpolant \( Q(x, y, z) \) at \((u_i, v_i, w_i)\), for \( i = 1, 2, \ldots, n \). If any of these evaluation points lie outside the region of definition of the interpolant, the corresponding entries in \( qx, qy \) and \( qz \) are set to the largest machine representable number (see nag_real_largest_number (X02ALC)), and nag_3d_shep_eval (e01thc) returns with fail.code = NE_BAD_POINT.

16: \( \text{fail} \) – NagError * \hspace{1cm} \text{Input/Output}

The NAG error parameter (see the Essential Introduction).

6 Error Indicators and Warnings

**NE_INT**

On entry, \( n = \langle \text{value} \rangle \).
Constraint: \( n \geq 1 \).

On entry, \( m = \langle \text{value} \rangle \).
Constraint: \( m \geq 10 \).

**NE_INT_ARRAY**

On entry, values in \( iq \) appear to be invalid. Check that \( iq \) has not been corrupted between calls to nag_3d_shep_interp (e01tgc) and nag_3d_shep_eval (e01thc).

**NE_BAD_POINT**

On entry, at least one evaluation point lies outside the region of definition of the interpolant. At all such points the corresponding values in \( q, qx, qy \) and \( qz \) have been set to nag_real_largest_number (X02ALC): nag_real_largest_number (X02ALC)(\) = \( \langle \text{value} \rangle \).

**NE_REAL_ARRAY**

On entry, values in \( rq \) appear to be invalid. Check that \( rq \) has not been corrupted between calls to nag_3d_shep_interp (e01tgc) and nag_3d_shep_eval (e01thc).

**NE_BAD_PARAM**

On entry, parameter \( \langle \text{value} \rangle \) had an illegal 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 consult NAG for assistance.
7  Accuracy
Computational errors should be negligible in most practical situations.

8  Further Comments
The time taken for a call to nag_3d_shep_eval (e01thc) will depend in general on the distribution of the data points. If \(x\), \(y\) and \(z\) are approximately uniformly distributed, then the time taken should be only \(O(n)\). At worst \(O(mn)\) time will be required.

9  Example
See Section 9 of the document for nag_3d_shep_interp (e01tgc).