

# NAG Library Function Document

## nag\_2d\_triang\_bary\_eval (e01ebc)

### 1 Purpose

nag\_2d\_triang\_bary\_eval (e01ebc) performs barycentric interpolation, at a given set of points, using a set of function values on a scattered grid and a triangulation of that grid computed by nag\_2d\_triangulate (e01eac).

### 2 Specification

```
#include <nag.h>
#include <nage01.h>

void nag_2d_triang_bary_eval (Integer m, Integer n, const double x[],
    const double y[], const double f[], const Integer triang[],
    const double px[], const double py[], double pf[], NagError *fail)
```

### 3 Description

nag\_2d\_triang\_bary\_eval (e01ebc) takes as input a set of scattered data points  $(x_r, y_r, f_r)$ , for  $r = 1, 2, \dots, n$ , and a Thiessen triangulation of the  $(x_r, y_r)$  computed by nag\_2d\_triangulate (e01eac), and interpolates at a set of points  $(px_i, py_i)$ , for  $i = 1, 2, \dots, m$ .

If the  $i$ th interpolation point  $(px_i, py_i)$  is equal to  $(x_r, y_r)$  for some value of  $r$ , the returned value will be equal to  $f_r$ ; otherwise a barycentric transformation will be used to calculate the interpolant.

For each point  $(px_i, py_i)$ , a triangle is sought which contains the point; the vertices of the triangle and  $f_r$  values at the vertices are then used to compute the value  $F(px_i, py_i)$ .

If any interpolation point lies outside the triangulation defined by the input arguments, the returned value is the value provided,  $f_s$ , at the closest node  $(x_s, y_s)$ .

nag\_2d\_triang\_bary\_eval (e01ebc) must only be called after a call to nag\_2d\_triangulate (e01eac).

### 4 References

Cline A K and Renka R L (1984) A storage-efficient method for construction of a Thiessen triangulation *Rocky Mountain J. Math.* **14** 119–139

Lawson C L (1977) Software for  $C^1$  surface interpolation *Mathematical Software III* (ed J R Rice) 161–194 Academic Press

Renka R L (1984) Algorithm 624: triangulation and interpolation of arbitrarily distributed points in the plane *ACM Trans. Math. Software* **10** 440–442

Renka R L and Cline A K (1984) A triangle-based  $C^1$  interpolation method *Rocky Mountain J. Math.* **14** 223–237

### 5 Arguments

- 1: **m** – Integer *Input*  
*On entry:*  $m$ , the number of points to interpolate.  
*Constraint:*  $m \geq 1$ .

- 2: **n** – Integer *Input*  
*On entry:*  $n$ , the number of data points. **n** must be unchanged from the previous call of `nag_2d_triangulate` (e01eac).  
*Constraint:*  $n \geq 3$ .
- 3: **x[n]** – const double *Input*  
4: **y[n]** – const double *Input*  
*On entry:* the coordinates of the  $r$ th data point,  $(x_r, y_r)$ , for  $r = 1, 2, \dots, n$ . **x** and **y** must be unchanged from the previous call of `nag_2d_triangulate` (e01eac).
- 5: **f[n]** – const double *Input*  
*On entry:* the function values  $f_r$  at  $(x_r, y_r)$ , for  $r = 1, 2, \dots, n$ .
- 6: **triang[7 × n]** – const Integer *Input*  
*On entry:* the triangulation computed by the previous call of `nag_2d_triangulate` (e01eac). See Section 9 in `nag_2d_triangulate` (e01eac) for details of how the triangulation used is encoded in **triang**.
- 7: **px[m]** – const double *Input*  
8: **py[m]** – const double *Input*  
*On entry:* the coordinates  $(px_i, py_i)$ , for  $i = 1, 2, \dots, m$ , at which interpolated function values are sought.
- 9: **pf[m]** – double *Output*  
*On exit:* the interpolated values  $F(px_i, py_i)$ , for  $i = 1, 2, \dots, m$ .
- 10: **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\_INT

On entry,  $m = \langle value \rangle$ .

Constraint:  $m \geq 1$ .

On entry,  $n = \langle value \rangle$ .

Constraint:  $n \geq 3$ .

### 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.

#### **NE\_TRIANG\_INVALID**

On entry, the triangulation information held in the array **triang** does not specify a valid triangulation of the data points. **triang** has been corrupted since the call to nag\_2d\_triangulate (e01eac).

#### **NW\_EXTRAPOLATION**

At least one evaluation point lies outside the nodal triangulation. For each such point the value returned in **pf** is that corresponding to a node on the closest boundary line segment.

### **7 Accuracy**

Not applicable.

### **8 Parallelism and Performance**

nag\_2d\_triang\_bary\_eval (e01ebc) is not threaded in any implementation.

### **9 Further Comments**

The time taken for a call of nag\_2d\_triang\_bary\_eval (e01ebc) is approximately proportional to the number of interpolation points,  $m$ .

### **10 Example**

See Section 10 in nag\_2d\_triangulate (e01eac).

---