NAG Toolbox for MATLAB

e02de

1 Purpose

e02de calculates values of a bicubic spline from its B-spline representation.

2 Syntax

```
[ff, ifail] = e02de(x, y, lamda, mu, c, 'm', m, 'px', px, 'py', py)
```

3 Description

e02de calculates values of the bicubic spline $s(x,y)$ at prescribed points (x_r,y_r), for $r = 1,2,\ldots,m$, from its augmented knot sets $\{\lambda\}$ and $\{\mu\}$ and from the coefficients c_{ij}, for $i = 1,2,\ldots,\mathbf{px}-4$; $j = 1,2,\ldots,\mathbf{py}-4$, in its B-spline representation

$$s(x,y) = \sum_{ij} c_{ij} M_i(x) N_j(y).$$

Here $M_i(x)$ and $N_j(y)$ denote normalized cubic B-splines, the former defined on the knots λ_i to λ_{i+4} and the latter on the knots μ_j to μ_{j+4}.

This function may be used to calculate values of a bicubic spline given in the form produced by e01da, e02da, e02dc and e02dd. It is derived from the function B2VRE in Anthony *et al.* 1982.

4 References

Anthony G T, Cox M G and Hayes J G 1982 *DASL – Data Approximation Subroutine Library* National Physical Laboratory

Cox M G 1978 The numerical evaluation of a spline from its B-spline representation *J. Inst. Math. Appl.* **21** 135–143

5 Parameters

5.1 Compulsory Input Parameters

1: **x(m) – double array**

2: **y(m) – double array**

x and **y** must contain x_r and y_r, for $r = 1,2,\ldots,m$, respectively. These are the co-ordinates of the points at which values of the spline are required. The order of the points is immaterial.

Constraint: **x** and **y** must satisfy

$$\mathbf{lamda}(4) \leq \mathbf{x}(r) \leq \mathbf{lamda}(\mathbf{px}-3)$$

and

$$\mathbf{mu}(4) \leq \mathbf{y}(r) \leq \mathbf{mu}(\mathbf{py}-3), \qquad r = 1,2,\ldots,m.$$

The spline representation is not valid outside these intervals.

3: **lamda**(**px**) – **double array**

4: **mu**(**py**) – **double array**

lamda and **mu** must contain the complete sets of knots $\{\lambda\}$ and $\{\mu\}$ associated with the x and y variables respectively.

Constraint: the knots in each set must be in nondecreasing order, with **lamda**(**px** $-$ 3) $>$ **lamda**(4) and **mu**(**py** $-$ 3) $>$ **mu**(4).

5: **c**((**px** $-$ **4**) $\times$ (**py** $-$ **4**)) – **double array**

c((**py** $-$ 4) $\times$ $(i-1)+j$) must contain the coefficient c_{ij} described in Section 3, for $i = 1, 2, \ldots,$ **px** $-$ 4 and $j = 1, 2, \ldots,$ **py** $-$ 4.

5.2 Optional Input Parameters

1: **m** – **int32 scalar**

Default: The dimension of the arrays **x**, **y**, **ff**. (An error is raised if these dimensions are not equal.)

m, the number of points at which values of the spline are required.

Constraint: **m** $\geq$ 1.

2: **px** – **int32 scalar**

3: **py** – **int32 scalar**

Default: For **px**, the dimension of the array **lamda**. For **py**, the dimension of the array **mu**.

px and **py** must specify the total number of knots associated with the variables x and y respectively. They are such that **px** $-$ 8 and **py** $-$ 8 are the corresponding numbers of interior knots.

Constraint: **px** $\geq$ 8 and **py** $\geq$ 8.

5.3 Input Parameters Omitted from the MATLAB Interface

wrk, iwrk

5.4 Output Parameters

1: **ff**(**m**) – **double array**

ff(r) contains the value of the spline at the point (x_r, y_r), for $r = 1, 2, \ldots, m$.

2: **ifail** – **int32 scalar**

0 unless the function detects an error (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

On entry, **m** $<$ 1,
or **py** $<$ 8,
or **px** $<$ 8.

ifail $= 2$

On entry, the knots in array **lamda**, or those in array **mu**, are not in nondecreasing order, or **lamda**(**px** $-$ 3) $\leq$ **lamda**(4), or **mu**(**py** $-$ 3) $\leq$ **mu**(4).

ifail $= 3$

On entry, at least one of the prescribed points (x_r, y_r) lies outside the rectangle defined by **lamda**(4), **lamda**$(\mathbf{px} - 3)$ and **mu**(4), **mu**$(\mathbf{py} - 3)$.

7 Accuracy

The method used to evaluate the B-splines is numerically stable, in the sense that each computed value of $s(x_r, y_r)$ can be regarded as the value that would have been obtained in exact arithmetic from slightly perturbed B-spline coefficients. See Cox 1978 for details.

8 Further Comments

Computation time is approximately proportional to the number of points, m, at which the evaluation is required.

9 Example

```
x = [1;
     1.1;
     1.5;
     1.6;
     1.9;
     1.9;
     2];
y = [0;
     0.1;
     0.7;
     0.4;
     0.3;
     0.8;
     1];
lamda = [1;
     1;
     1;
     1;
     1.3;
     1.5;
     1.6;
     2;
     2;
     2;
     2];
mu = [0;
     0;
     0;
     0;
     0.4;
     0.7;
     1;
     1;
     1;
     1];
c = [1;
     1.1333;
     1.3667;
     1.7;
     1.9;
     2;
     1.2;
     1.3333;
     1.5667;
     1.9;
     2.1;
```

```
     2.2;
     1.5833;
     1.7167;
     1.95;
     2.2833;
     2.4833;
     2.5833;
     2.1433;
     2.2767;
     2.51;
     2.8433;
     3.0433;
     3.1433;
     2.8667;
     3;
     3.2333;
     3.5667;
     3.7667;
     3.8667;
     3.4667;
     3.6;
     3.8333;
     4.1667;
     4.3667;
     4.4667;
     4;
     4.1333;
     4.3667;
     4.7;
     4.9;
     5];
[ff, ifail] = e02de(x, y, lamda, mu, c)
```

```
ff =
    1.0000
    1.3100
    2.9500
    2.9600
    3.9100
    4.4100
    5.0000
ifail =
           0
```