NAG Toolbox

nag_ode_bvp_ps_lin_grid_vals (d02uw)

1 Purpose

nag_ode_bvp_ps_lin_grid_vals (d02uw) interpolates from a set of function values on a supplied grid onto a set of values for a uniform grid on the same range. The interpolation is performed using barycentric Lagrange interpolation. nag_ode_bvp_ps_lin_grid_vals (d02uw) is primarily a utility function to map a set of function values specified on a Chebyshev Gauss–Lobatto grid onto a uniform grid.

2 Syntax

```
[xip, fip, ifail] = nag_ode_bvp_ps_lin_grid_vals(n, nip, x, f)
[xip, fip, ifail] = d02uw(n, nip, x, f)
```

3 Description

nag_ode_bvp_ps_lin_grid_vals (d02uw) interpolates from a set of $n+1$ function values, $f(x_i)$, on a supplied grid, x_i, for $i = 0, 1, \ldots, n$, onto a set of m values, $\hat{f}(\hat{x}_j)$, on a uniform grid, $\hat{x}_j$, for $j = 1, 2, \ldots, m$. The image $\hat{x}$ has the same range as x, so that $\hat{x}_j = x_{\min} + ((j-1)/(m-1)) \times (x_{\max} - x_{\min})$, for $j = 1, 2, \ldots, m$. The interpolation is performed using barycentric Lagrange interpolation as described in Berrut and Trefethen (2004).

nag_ode_bvp_ps_lin_grid_vals (d02uw) is primarily a utility function to map a set of function values specified on a Chebyshev Gauss–Lobatto grid computed by nag_ode_bvp_ps_lin_cgl_grid (d02uc) onto an evenly-spaced grid with the same range as the original grid.

4 References

Berrut J P and Trefethen L N (2004) Barycentric lagrange interpolation *SIAM Rev.* **46(3)** 501–517

5 Parameters

5.1 Compulsory Input Parameters

1: **n** – INTEGER

n, where the number of grid points for the input data is $n+1$.

Constraint: **n** > 0 and **n** is even.

2: **nip** – INTEGER

The number, m, of grid points in the uniform mesh $\hat{x}$ onto which function values are interpolated. If **nip** $= 1$ then on successful exit from nag_ode_bvp_ps_lin_grid_vals (d02uw), **fip**(1) will contain the value $f(x_n)$.

Constraint: **nip** > 0.

3: **x**$(\mathbf{n}+\mathbf{1})$ – REAL (KIND=nag_wp) array

The grid points, x_i, for $i = 0, 1, \ldots, n$, at which the function is specified.

Usually this should be the array of Chebyshev Gauss–Lobatto points returned in nag_ode_bvp_ps_lin_cgl_grid (d02uc).

4: **f**(**n** + **1**) – REAL (KIND=nag_wp) array

The function values, $f(x_i)$, for $i = 0, 1, \ldots, n$.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **xip**(**nip**) – REAL (KIND=nag_wp) array

The evenly-spaced grid points, $\hat{x}_j$, for $j = 1, 2, \ldots, m$.

2: **fip**(**nip**) – REAL (KIND=nag_wp) array

The set of interpolated values $\hat{f}(\hat{x}_j)$, for $j = 1, 2, \ldots, m$. Here $\hat{f}(\hat{x}_j) \approx f(x = \hat{x}_j)$.

3: **ifail** – INTEGER

ifail $= 0$ unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail $= 1$

Constraint: **n** > 0.

Constraint: **n** is even.

ifail $= 2$

Constraint: **nip** > 0.

ifail $= -99$

An unexpected error has been triggered by this routine. Please contact NAG.

ifail $= -399$

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

ifail $= -999$

Dynamic memory allocation failed.

7 Accuracy

nag_ode_bvp_ps_lin_grid_vals (d02uw) is intended, primarily, for use with Chebyshev Gauss–Lobatto input grids. For such input grids and for well-behaved functions (no discontinuities, peaks or cusps), the accuracy should be a small multiple of ***machine precision***.

8 Further Comments

None.

9 Example

This example interpolates the function $x + \cos(5x)$, as specified on a 65-point Gauss–Lobatto grid on $[-1, 1]$, onto a coarse uniform grid.

9.1 Program Text

```
     function d02uw_example

fprintf('d02uw example results\n\n');

n   = nag_int(64);
a = -1;
b =  1;

% Set up Chebyshev grid
[x, ifail] = d02uc(n, a, b);

% Set up function on grid
f = x + cos(5*x);

% Interpolate onto smaller equally spaced grid
nip = nag_int(17);
[xip, fip, ifail] = d02uw(n, nip, x, f);

% Display interpolated values
fprintf('\nInterpolated function values\n');
fprintf('      x          F\n');
fprintf('%10.4f %10.4f \n', [xip fip]');
```

9.2 Program Results

```
     d02uw example results

Interpolated function values
      x          F
   -1.0000    -0.7163
   -0.8750    -1.2060
   -0.7500    -1.5706
   -0.6250    -1.6249
   -0.5000    -1.3011
   -0.3750    -0.6745
   -0.2500     0.0653
   -0.1250     0.6860
    0.0000     1.0000
    0.1250     0.9360
    0.2500     0.5653
    0.3750     0.0755
    0.5000    -0.3011
    0.6250    -0.3749
    0.7500    -0.0706
    0.8750     0.5440
    1.0000     1.2837
```