NAG Toolbox for MATLAB

d01ah

1 Purpose

d01ah computes a definite integral over a finite range to a specified relative accuracy using a method described by Patterson.

2 Syntax

```
[result, npts, relerr, ifail] = d01ah(a, b, epsr, f, nlimit)
```

3 Description

d01ah computes a definite integral of the form

$$\int_a^b f(x)\,dx.$$

The method uses as its basis a family of interlacing high precision rules (see Patterson 1968b) using 1, 3, 7, 15, 31, 63, 127 and 255 nodes. Initially the family is applied in sequence to the integrand. When two successive rules differ relatively by less than the required relative accuracy, the last rule used is taken as the value of the integral and the operation is regarded as successful. If all rules in the family have been applied unsuccessfully, subdivision is invoked. The subdivision strategy is as follows. The interval under scrutiny is divided into two sub-intervals (not always equal). The basic family is then applied to the first sub-interval. If the required accuracy is not obtained, the interval is stored for future examination (see **ifail** $= 2$) and the second sub-interval is examined. Should the basic family again be unsuccessful, then the sub-interval is further subdivided and the whole process repeated. Successful integrations are accumulated as the partial value of the integral. When all possible successful integrations have been completed, those previously unsuccessful sub-intervals placed in store are examined.

A large number of refinements are incorporated to improve the performance. Some of these are:

(a) The rate of convergence of the basic family is monitored and used to make a decision to abort and subdivide before the full sequence has been applied.

(b) The ϵ-algorithm is applied to the basic results in an attempt to increase the convergence rate. See Wynn 1956.

(c) An attempt is made to detect sharp end point peaks and singularities in each sub-interval and to apply appropriate transformations to smooth the integrand. This consideration is also used to select interval sizes in the subdivision process.

(d) The relative accuracy sought in each sub-interval is adjusted in accordance with its likely contribution to the total integral.

(e) Random transformations of the integrand are applied to improve reliability in some instances.

4 References

Patterson T N L 1968b The Optimum addition of points to quadrature formulae *Math. Comput.* **22** 847–856

Wynn P 1956 On a device for computing the $e_m(S_n)$ transformation *Math. Tables Aids Comput.* **10** 91–96

5 Parameters

5.1 Compulsory Input Parameters

1: **a – double scalar**

a, the lower limit of integration.

2: **b – double scalar**

b, the upper limit of integration. It is not necessary that $a < b$.

3: **epsr – double scalar**

The relative accuracy required.

Constraint: **epsr** > 0.0.

4: **f – string containing name of m-file**

f must return the value of the integrand f at a given point.

Its specification is:

```
[result] = f(x)
```

Input Parameters

1: **x – double scalar**

The point at which the integrand f must be evaluated.

Output Parameters

1: **result – double scalar**

The result of the function.

5: **nlimit – int32 scalar**

A limit to the number of function evaluations. If **nlimit** ≤ 0, the function uses a default limit of 10000.

5.2 Optional Input Parameters

None.

5.3 Input Parameters Omitted from the MATLAB Interface

None.

5.4 Output Parameters

1: **result – double scalar**

The result of the function.

2: **npts – int32 scalar**

The number of function evaluations used in the calculation of the integral.

3: **relerr – double scalar**

A rough estimate of the relative error achieved.

4: **ifail – int32 scalar**

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

6 Error Indicators and Warnings

Note: d01ah may return useful information for one or more of the following detected errors or warnings.

ifail = 1

The integral has not converged to the accuracy requested. It may be worthwhile to try increasing **nlimit**.

ifail = 2

Too many unsuccessful levels of subdivision have been invoked.

ifail = 3

On entry, **epsr** ≤ 0.0.

When **ifail** = 1 or 2 a result may be obtained by continuing without further subdivision, but this is likely to be **inaccurate**.

7 Accuracy

The relative accuracy required is specified by you in the variable **epsr**. The function will terminate whenever the relative accuracy specified by **epsr** is judged to have been reached.

If on exit, **ifail** = 0, then it is most likely that the result is correct to the specified accuracy. If, on exit, **ifail** = 1 or 2, then it is likely that the specified accuracy has not been reached.

relerr is a rough estimate of the relative error achieved. It is a by-product of the computation and is not used to effect the termination of the function. The outcome of the integration must be judged by the value of **ifail**.

8 Further Comments

The time taken by d01ah depends on the complexity of the integrand and the accuracy required.

9 Example

```
d01ah_f.m

function [result] = d01ah_f(x)
 result=4.0/(1.0+x^2);
```

```
a = 0;
b = 1;
epsr = 1e-05;
nlimit = int32(0);
[result, npts, relerr, ifail] = d01ah(a, b, epsr, 'd01ah_f', nlimit)
```

```
result =
    3.1416
npts =
          15
relerr =
   5.8494e-09
ifail =
           0
```