NAG Toolbox

nag_roots_contfn_brent (c05ay)

1 Purpose

nag_roots_contfn_brent (c05ay) locates a simple zero of a continuous function in a given interval using Brent's method, which is a combination of nonlinear interpolation, linear extrapolation and bisection.

2 Syntax

```
[x, user, ifail] = nag_roots_contfn_brent(a, b, eps, eta, f, 'user', user)
[x, user, ifail] = c05ay(a, b, eps, eta, f, 'user', user)
```

3 Description

nag_roots_contfn_brent (c05ay) attempts to obtain an approximation to a simple zero of the function $f(x)$ given an initial interval $[a, b]$ such that $f(a) \times f(b) \leq 0$. The same core algorithm is used by nag_roots_contfn_brent_rcomm (c05az) whose specification should be consulted for details of the method used.

The approximation x to the zero α is determined so that at least one of the following criteria is satisfied:

(i) $|x - \alpha| \leq$ **eps**,

(ii) $|f(x)| \leq$ **eta**.

4 References

Brent R P (1973) *Algorithms for Minimization Without Derivatives* Prentice–Hall

5 Parameters

5.1 Compulsory Input Parameters

1: **a** – REAL (KIND=nag_wp)

a, the lower bound of the interval.

2: **b** – REAL (KIND=nag_wp)

b, the upper bound of the interval.

Constraint: **b** $\neq$ **a**.

3: **eps** – REAL (KIND=nag_wp)

The termination tolerance on x (see Section 3).

Constraint: **eps** > 0.0.

4: **eta** – REAL (KIND=nag_wp)

A value such that if $|f(x)| \leq$ **eta**, x is accepted as the zero. **eta** may be specified as 0.0 (see Section 7).

5: **f** – REAL (KIND=nag_wp) FUNCTION, supplied by the user.

f must evaluate the function f whose zero is to be determined.

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

Input Parameters

1: **x** – REAL (KIND=nag_wp)

The point at which the function must be evaluated.

2: **user** – INTEGER array

f is called from nag_roots_contfn_brent (c05ay) with the object supplied to nag_roots_contfn_brent (c05ay).

Output Parameters

1: **result**

The value of f evaluated at **x**.

2: **user** – INTEGER array

5.2 Optional Input Parameters

1: **user** – INTEGER array

user is not used by nag_roots_contfn_brent (c05ay), but is passed to **f**. Note that for large objects it may be more efficient to use a global variable which is accessible from the m-files than to use **user**.

5.3 Output Parameters

1: **x** – REAL (KIND=nag_wp)

If **ifail** $= 0$ or 2, **x** is the final approximation to the zero. If **ifail** $= 3$, **x** is likely to be a pole of $f(x)$. Otherwise, **x** contains no useful information.

2: **user** – INTEGER array

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: **a** $\neq$ **b**.

Constraint: **eps** > 0.0.

On entry, $\mathbf{f}(\mathbf{a})$ and $\mathbf{f}(\mathbf{b})$ have the same sign with neither equalling 0.0.

ifail $= 2$ (*warning*)

No further improvement in the solution is possible.

ifail $= 3$ (*warning*)

The function values in the interval $[\mathbf{a}, \mathbf{b}]$ might contain a pole rather than a zero. Reducing **eps** may help in distinguishing between a pole and a zero.

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

The levels of accuracy depend on the values of **eps** and **eta**. If full machine accuracy is required, they may be set very small, resulting in an exit with **ifail** $= 2$, although this may involve many more iterations than a lesser accuracy. You are recommended to set **eta** $= 0.0$ and to use **eps** to control the accuracy, unless you have considerable knowledge of the size of $f(x)$ for values of x near the zero.

8 Further Comments

The time taken by nag_roots_contfn_brent (c05ay) depends primarily on the time spent evaluating **f** (see Section 5).

If it is important to determine an interval of relative length less than $2 \times$ **eps** containing the zero, or if **f** is expensive to evaluate and the number of calls to **f** is to be restricted, then use of nag_roots_contfn_brent_rcomm (c05az) is recommended. Use of nag_roots_contfn_brent_rcomm (c05az) is also recommended when the structure of the problem to be solved does not permit a simple **f** to be written: the reverse communication facilities of nag_roots_contfn_brent_rcomm (c05az) are more flexible than the direct communication of **f** required by nag_roots_contfn_brent (c05ay).

9 Example

This example calculates an approximation to the zero of $e^{-x} - x$ within the interval $[0, 1]$ using a tolerance of **eps** $= 1.0\text{e}{-5}$.

9.1 Program Text

```
function c05ay_example

fprintf('c05ay example results\n\n');

a = 0;
b = 1;
eps = 1e-5;
eta = 0;
fprintf('\n');
[x, user, ifail] = c05ay(a, b, eps, eta, @f);
switch ifail
  case {0}
    fprintf('With eps = %10.2e, root = %14.5f\n', eps, x);
  case {2, 3}
    fprintf('With eps = %10.2e, final value = %14.5f\n', eps, x);
end

function [result, user] = f(x, user)
  result = x - exp(-x);
```

9.2 Program Results

```
     c05ay example results

With eps =   1.00e-05, root =        0.56714
```