

NAG Toolbox for MATLAB

s21bc

1 Purpose

s21bc returns a value of the symmetrised elliptic integral of the second kind, via the function name.

2 Syntax

```
[result, ifail] = s21bc(x, y, z)
```

3 Description

s21bc calculates an approximate value for the integral

$$R_D(x, y, z) = \frac{3}{2} \int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)^3}}$$

where $x, y \geq 0$, at most one of x and y is zero, and $z > 0$.

The basic algorithm, which is due to Carlson 1979 and Carlson 1988, is to reduce the arguments recursively towards their mean by the rule:

$$\begin{aligned} x_0 &= x, y_0 = y, z_0 = z \\ \mu_n &= (x_n + y_n + 3z_n)/5 \\ X_n &= (1 - x_n)/\mu_n \\ Y_n &= (1 - y_n)/\mu_n \\ Z_n &= (1 - z_n)/\mu_n \\ \lambda_n &= \sqrt{x_n y_n} + \sqrt{y_n z_n} + \sqrt{z_n x_n} \\ x_{n+1} &= (x_n + \lambda_n)/4 \\ y_{n+1} &= (y_n + \lambda_n)/4 \\ z_{n+1} &= (z_n + \lambda_n)/4 \end{aligned}$$

For n sufficiently large,

$$\epsilon_n = \max(|X_n|, |Y_n|, |Z_n|) \sim \left(\frac{1}{4}\right)^n$$

and the function may be approximated adequately by a fifth order power series

$$\begin{aligned} R_D(x, y, z) = & 3 \sum_{m=0}^{n-1} \frac{4^{-m}}{(z_m + \lambda_n) \sqrt{z_m}} \\ & + \frac{4^{-n}}{\sqrt{\mu_n^3}} \left[1 + \frac{3}{7} S_n^{(2)} + \frac{1}{3} S_n^{(3)} + \frac{3}{22} (S_n^{(2)})^2 + \frac{3}{11} S_n^{(4)} + \frac{3}{13} S_n^{(2)} S_n^{(3)} + \frac{3}{13} S_n^{(5)} \right] \end{aligned}$$

where $S_n^{(m)} = (X_n^m + Y_n^m + 3Z_n^m)/2m$. The truncation error in this expansion is bounded by $\frac{3\epsilon_n^6}{\sqrt{(1 - \epsilon_n)^3}}$

and the recursive process is terminated when this quantity is negligible compared with the **machine precision**.

The function may fail either because it has been called with arguments outside the domain of definition, or with arguments so extreme that there is an unavoidable danger of setting underflow or overflow.

Note: $R_D(x, x, x) = x^{-3/2}$, so there exists a region of extreme arguments for which the function value is not representable.

4 References

Abramowitz M and Stegun I A 1972 *Handbook of Mathematical Functions* (3rd Edition) Dover Publications

Carlson B C 1979 Computing elliptic integrals by duplication *Numerische Mathematik* **33** 1–16

Carlson B C 1988 A table of elliptic integrals of the third kind *Math. Comput.* **51** 267–280

5 Parameters

5.1 Compulsory Input Parameters

- 1: **x** – double scalar
- 2: **y** – double scalar
- 3: **z** – double scalar

The arguments x , y and z of the function.

Constraint: $x, y \geq 0.0$, $z > 0.0$ and only one of x and y may be zero.

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: **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, either x or y is negative, or both x and y are zero; the function is undefined.

ifail = 2

On entry, $z \leq 0.0$; the function is undefined.

ifail = 3

On entry, either z is too close to zero or both x and y are too close to zero: there is a danger of setting overflow.

ifail = 4

On entry, at least one of x , y and z is too large: there is a danger of setting underflow. On soft failure the function returns zero.

7 Accuracy

In principle the function is capable of producing full *machine precision*. However round-off errors in internal arithmetic will result in slight loss of accuracy. This loss should never be excessive as the algorithm does not involve any significant amplification of round-off error. It is reasonable to assume that the result is accurate to within a small multiple of the *machine precision*.

8 Further Comments

You should consult the S Chapter Introduction which shows the relationship of this function to the classical definitions of the elliptic integrals.

9 Example

```
x = 0.5;
y = 0.5;
z = 1;
[result, ifail] = s21bc(x, y, z)

result =
    1.4787
ifail =
    0
```
