NAG Toolbox

nag_specfun_ellipint_symm_3 (s21bd)

1 Purpose

nag_specfun_ellipint_symm_3 (s21bd) returns a value of the symmetrised elliptic integral of the third kind, via the function name.

2 Syntax

```
[result, ifail] = nag_specfun_ellipint_symm_3(x, y, z, r)
[result, ifail] = s21bd(x, y, z, r)
```

3 Description

nag_specfun_ellipint_symm_3 (s21bd) calculates an approximation to the integral

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

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

If $\rho<0$, the result computed is the Cauchy principal value of the integral.

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

$$\begin{aligned}
x_0 &= x, y_0=y, z_0=z, \rho_0=\rho \\
\mu_n &= (x_n+y_n+z_n+2\rho_n)/5 \\
X_n &= 1-x_n/\mu_n \\
Y_n &= 1-y_n/\mu_n \\
Z_n &= 1-z_n/\mu_n \\
P_n &= 1-\rho_n/\mu_n \\
\lambda_n &= \sqrt{x_ny_n}+\sqrt{y_nz_n}+\sqrt{z_nx_n} \\
x_{n+1} &= (x_n+\lambda_n)/4 \\
y_{n+1} &= (y_n+\lambda_n)/4 \\
z_{n+1} &= (z_n+\lambda_n)/4 \\
\rho_{n+1} &= (\rho_n+\lambda_n)/4 \\
\alpha_n &= \left[\rho_n\left(\sqrt{x_n},+\sqrt{y_n},+\sqrt{z_n}\right)+\sqrt{x_ny_nz_n}\right]^2 \\
\beta_n &= \rho_n(\rho_n+\lambda_n)^2
\end{aligned}$$

For n sufficiently large,

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

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

$$R_J(x,y,z,\rho)=\quad 3\sum_{m=0}^{n-1}4^{-m}R_C(\alpha_m,\beta_m)$$

$$+\frac{4^{-n}}{\sqrt{\mu_n^3}}\left[1+\frac{3}{7}S_n^{(2)}+\frac{1}{3}S_n^{(3)}+\frac{3}{22}\left(S_n^{(2)}\right)^2+\frac{3}{11}S_n^{(4)}+\frac{3}{13}S_n^{(2)}S_n^{(3)}+\frac{3}{13}S_n^{(5)}\right]$$

where $S_n^{(m)}=\left(X_n^m+Y_n^m+Z_n^m+2P_n^m\right)/2m$.

The truncation error in this expansion is bounded by $3\epsilon_n^6/\sqrt{(1-\epsilon_n)^3}$ and the recursion 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: $R_J(x,x,x,x) = x^{-\frac{3}{2}}$, 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** – REAL (KIND=nag_wp)
2: **y** – REAL (KIND=nag_wp)
3: **z** – REAL (KIND=nag_wp)
4: **r** – REAL (KIND=nag_wp)

The arguments x, y, z and ρ of the function.

Constraint: **x**, **y**, **z** ≥ 0.0, **r** $\neq 0.0$ and at most one of **x**, **y** and **z** may be zero.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

1: **result**

The result of the function.

2: **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 $= 1$

On entry, at least one of **x**, **y** and **z** is negative, or at least two of them are zero; the function is undefined.

ifail $= 2$

r $= 0.0$; the function is undefined.

ifail $= 3$

On entry, either **r** is too close to zero, or any two of **x**, **y** and **z** are too close to zero; there is a danger of setting overflow.

ifail $= 4$

On entry, at least one of **x**, **y**, **z** and **r** is too large; there is a danger of setting underflow.

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

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.

If the argument **r** is equal to any of the other arguments, the function reduces to the integral R_D, computed by nag_specfun_ellipint_symm_2 (s21bc).

9 Example

This example simply generates a small set of nonextreme arguments which are used with the function to produce the table of low accuracy results.

9.1 Program Text

```
function s21bd_example

fprintf('s21bd example results\n\n');

x = [0.5   0.5   0.5   0.5   0.5   0.5   1    1    1     1.5];
y = [0.5   0.5   0.5   1     1     1.5   1    1    1.5   1.5];
z = [0.5   1     1.5   1     1.5   1.5   1    1.5  1.5   1.5];
r = [2     2     2     2     2     2     2    2    2     2  ];
result = x;

for j=1:numel(x)
  [result(j), ifail] = s21bd(x(j), y(j), z(j), r(j));
end

fprintf('    x      y      z      r      R_J(x,y,z)\n');
fprintf('%7.2f%7.2f%7.2f%7.2f%12.4f\n',[x; y; z; r; result]);
```

9.2 Program Results

```
s21bd example results

    x      y      z      r      R_J(x,y,z)
   0.50   0.50   0.50   2.00      1.1184
   0.50   0.50   1.00   2.00      0.9221
   0.50   0.50   1.50   2.00      0.8115
   0.50   1.00   1.00   2.00      0.7671
   0.50   1.00   1.50   2.00      0.6784
```

```
  0.50   1.50   1.50   2.00       0.6017
  1.00   1.00   1.00   2.00       0.6438
  1.00   1.00   1.50   2.00       0.5722
  1.00   1.50   1.50   2.00       0.5101
  1.50   1.50   1.50   2.00       0.4561
```