NAG Toolbox for Matlab

s19aa

1 Purpose

s19aa returns a value for the Kelvin function \( \text{ber} \, x \) via the function name.

2 Syntax

\[
[\text{result}, \, \text{ifail}] = \\text{s19aa}(x)
\]

3 Description

s19aa evaluates an approximation to the Kelvin function \( \text{ber} \, x \).

Note: \( \text{ber}(-x) = \text{ber} \, x \), so the approximation need only consider \( x \geq 0.0 \).

The function is based on several Chebyshev expansions:

For \( 0 \leq x \leq 5 \),

\[
\text{ber} \, x = \sum_{r=0}^{\infty} a_r T_r(t), \quad \text{with} \ t = 2 \left( \frac{x}{5} \right)^4 - 1.
\]

For \( x > 5 \),

\[
\text{ber} \, x = \frac{e^{x/\sqrt{2}}}{\sqrt{2 \pi x}} \left[ \left( 1 + \frac{1}{x} a(t) \right) \cos \alpha + \frac{1}{x} b(t) \sin \alpha \right] + \frac{e^{-x/\sqrt{2}}}{\sqrt{2 \pi x}} \left[ \left( 1 + \frac{1}{x} c(t) \right) \sin \beta + \frac{1}{x} d(t) \cos \beta \right],
\]

where \( \alpha = \frac{x}{\sqrt{2}} - \frac{\pi}{8}, \beta = \frac{x}{\sqrt{2}} + \frac{\pi}{8} \).

and \( a(t), b(t), c(t), \) and \( d(t) \) are expansions in the variable \( t = \frac{10}{x} - 1 \).

When \( x \) is sufficiently close to zero, the result is set directly to \( \text{ber} \, 0 = 1.0 \).

For large \( x \), there is a danger of the result being totally inaccurate, as the error amplification factor grows in an essentially exponential manner; therefore the function must fail.

4 References


5 Parameters

5.1 Compulsory Input Parameters

1: \( x \) – double scalar
   The argument \( x \) of the function.

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
   ifail = 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, abs(x) is too large for an accurate result to be returned. On soft failure, the function
   returns zero.

7 Accuracy
Since the function is oscillatory, the absolute error rather than the relative error is important. Let E be the
absolute error in the result and \( \delta \) be the relative error in the argument. If \( \delta \) is somewhat larger than the
machine precision, then we have:

\[
E \approx \frac{x}{\sqrt{2}} \left| \text{ber}_1 x + \text{bei}_1 x \right| \delta
\]

(provided \( E \) is within machine bounds).
For small \( x \) the error amplification is insignificant and thus the absolute error is effectively bounded by the
machine precision.
For medium and large \( x \), the error behaviour is oscillatory and its amplitude grows like \( \sqrt{\frac{x e^x}{2\pi}} \).
Therefore it is not possible to calculate the function with any accuracy when \( \sqrt{\frac{x e^x}{2\pi}} > \frac{\sqrt{2\pi}}{\delta} \). Note that
this value of \( x \) is much smaller than the minimum value of \( x \) for which the function overflows.

8 Further Comments
None.

9 Example

```matlab
x = 0.1;
[result, ifail] = s19aa(x)
```

result =
1.0000
ifail =
0