

NAG Library Function Document

nag_dawson (s15afc)

1 Purpose

nag_dawson (s15afc) returns a value for Dawson's Integral, $F(x)$.

2 Specification

```
#include <nag.h>
#include <nags.h>
double nag_dawson (double x)
```

3 Description

nag_dawson (s15afc) evaluates an approximation for Dawson's Integral

$$F(x) = e^{-x^2} \int_0^x e^{t^2} dt.$$

The function is based on two Chebyshev expansions:

For $0 < |x| \leq 4$,

$$F(x) = x \sum_{r=0}^l a_r T_r(t), \quad \text{where } t = 2 \left(\frac{x}{4} \right)^2 - 1.$$

For $|x| > 4$,

$$F(x) = \frac{1}{x} \sum_{r=0}^l b_r T_r(t), \quad \text{where } t = 2 \left(\frac{4}{x} \right)^2 - 1.$$

For $|x|$ near zero, $F(x) \simeq x$, and for $|x|$ large, $F(x) \simeq \frac{1}{2x}$. These approximations are used for those values of x for which the result is correct to *machine precision*.

4 References

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

5 Arguments

1: x – double

Input

On entry: the argument x of the function.

6 Error Indicators and Warnings

7 Accuracy

Let δ and ϵ be the relative errors in the argument and result respectively.

If δ is considerably greater than the *machine precision* (i.e., if δ is due to data errors etc.), then ϵ and δ are approximately related by:

$$\epsilon \simeq \left| \frac{x(1 - 2xF(x))}{F(x)} \right| \delta.$$

The following graph shows the behaviour of the error amplification factor $\left| \frac{x(1 - 2xF(x))}{F(x)} \right|$:

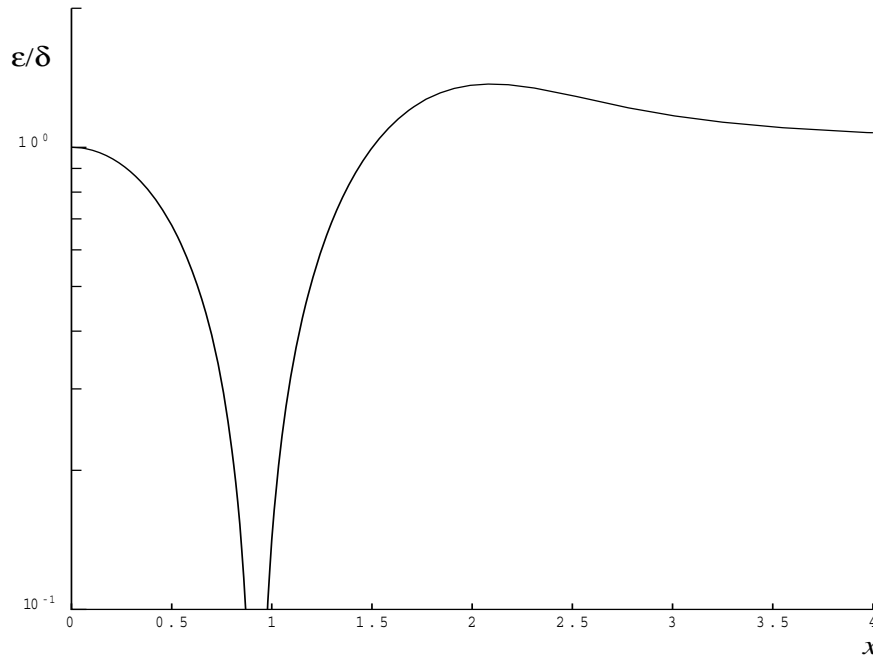


Figure 1

However if δ is of the same order as *machine precision*, then rounding errors could make ϵ somewhat larger than the above relation indicates. In fact ϵ will be largely independent of x or δ , but will be of the order of a few times the *machine precision*.

8 Parallelism and Performance

nag_dawson (s15afc) is not threaded in any implementation.

9 Further Comments

None.

10 Example

This example reads values of the argument x from a file, evaluates the function at each value of x and prints the results.

10.1 Program Text

```
/* nag_dawson (s15afc) Example Program.
 *
 * NAGPRODCODE Version.
 *
 * Copyright 2016 Numerical Algorithms Group.
 *
 * Mark 26, 2016.
 */

#include <nag.h>
#include <stdio.h>
```

```

#include <nag_stdlib.h>
#include <nags.h>

int main(void)
{
    double x, y;
    Integer exit_status = 0;

    /* Skip heading in data file */
#ifdef _WIN32
    scanf_s("%*[\n]");
#else
    scanf("%*[\n]");
#endif
    printf("nag_dawson (s15afc) Example Program Results\n");
    printf("      x              y\n");
#ifdef _WIN32
    while (scanf_s("%lf", &x) != EOF)
#else
    while (scanf("%lf", &x) != EOF)
#endif
    {
        /* nag_dawson (s15afc).
         * Dawson's integral
         */
        y = nag_dawson(x);
        printf("%12.3e %12.3e\n", x, y);
    }

    return exit_status;
}

```

10.2 Program Data

```

nag_dawson (s15afc) Example Program Data
-2.0
-0.5
 1.0
 1.5
 2.0
 5.0
10.0

```

10.3 Program Results

```

nag_dawson (s15afc) Example Program Results
      x              y
-2.000e+00  -3.013e-01
-5.000e-01  -4.244e-01
 1.000e+00   5.381e-01
 1.500e+00   4.282e-01
 2.000e+00   3.013e-01
 5.000e+00   1.021e-01
1.000e+01   5.025e-02

```
