NAG Fortran Library Routine Document

G01MUF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of ***bold italicised*** terms and other implementation-dependent details.

1 Purpose

G01MUF returns the value of the Vavilov density function $\phi_V(\lambda;\kappa,\beta^2)$, via the routine name.

It is intended to be used after a call to G01ZUF.

2 Specification

double precision `FUNCTION G01MUF (X, WORK, IFAIL)`

```
INTEGER                          IFAIL
double precision                 X, WORK(322)
```

3 Description

G01MUF evaluates an approximation to the Vavilov density function $\phi_V(\lambda;\kappa,\beta^2)$ given by

$$\phi_V(\lambda;\kappa,\beta^2) = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{\lambda s} f(s;\kappa,\beta^2)\,ds,$$

where $\kappa > 0$ and $0 \leq \beta^2 \leq 1$, c is an arbitrary real constant and

$$f(s;\kappa,\beta^2) = C(\kappa,\beta^2)\exp\left\{s\ln\kappa + (s+\kappa\beta^2)\left[\ln\left(\frac{s}{\kappa}\right) + E_1\left(\frac{s}{\kappa}\right)\right] - \kappa\exp\left(-\frac{s}{\kappa}\right)\right\}.$$

$E_1(x) = \displaystyle\int_0^x t^{-1}(1-e^{-t})\,dt$ is the exponential integral, $C(\kappa,\beta^2) = \exp\{\kappa(1+\gamma\beta^2)\}$ and γ is Euler's constant.

The method used is based on Fourier expansions. Further details can be found in Schorr (1974).

For values of $\kappa \leq 0.01$, the Vavilov distribution can be replaced by the Landau distribution since $\lambda_V = (\lambda_L - \ln\kappa)/\kappa$. For values of $\kappa \geq 10$, the Vavilov distribution can be replaced by a Gaussian distribution with mean $\mu = \gamma - 1 - \beta^2 - \ln\kappa$ and variance $\sigma^2 = (2-\beta^2)/2\kappa$.

4 References

Schorr B (1974) Programs for the Landau and the Vavilov distributions and the corresponding random numbers *Comp. Phys. Comm.* **7** 215–224

5 Parameters

1: X – ***double precision*** *Input*

On entry: the argument λ of the function.

2: WORK(322) – ***double precision*** array *Communication Array*

On entry: this **must** be the same parameter WORK as returned by a previous call to G01ZUF.

3: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, −1 or 1. Users who are unfamiliar with this parameter should refer to Chapter P01 for details.

On exit: IFAIL $= 0$ unless the routine detects an error (see Section 6).

For environments where it might be inappropriate to halt program execution when an error is detected, the value -1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, for users not familiar with this parameter the recommended value is 0. **When the value -1 or 1 is used it is essential to test the value of IFAIL on exit.**

6 Error Indicators and Warnings

IFAIL $= 1$

Either the initialization routine has not been called prior to the first call of this routine or a communication array has become corrupted.

7 Accuracy

At least 5 significant digits are usually correct.

8 Further Comments

G01MUF can be called repeatedly with different values of λ provided that the values of κ and β^2 remain unchanged between calls. Otherwise, G01ZUF must be called again. This is illustrated in Section 9.

9 Example

The example program evaluates $\phi_V\left(\lambda;\kappa,\beta^2\right)$ at $\lambda = 2.5$, $\kappa = 0.4$ and $\beta^2 = 0.1$, and prints the results.

9.1 Program Text

Note: the listing of the example program presented below uses ***bold italicised*** terms to denote precision-dependent details. Please read the Users' Note for your implementation to check the interpretation of these terms. As explained in the Essential Introduction to this manual, the results produced may not be identical for all implementations.

```
*     G01MUF Example Program Text
*     Mark 21 Release. NAG Copyright 2004.
*     .. Parameters ..
      INTEGER          NIN, NOUT
      PARAMETER        (NIN=5,NOUT=6)
      INTEGER          MODE
      PARAMETER        (MODE=0)
*     .. Local Scalars ..
      DOUBLE PRECISION BETA2, C1, C2, RKAPPA, X, XL, XU, Y
      INTEGER          IFAIL
*     .. External Functions ..
      DOUBLE PRECISION G01MUF, X02ALF
      EXTERNAL         G01MUF, X02ALF
*     .. Local Arrays ..
      DOUBLE PRECISION WORK(322)
*     .. External Subroutines ..
      EXTERNAL         G01ZUF
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G01MUF Example Program Results'
*     Skip heading in data file
      READ (NIN,*)
      C1 = -X02ALF()
      C2 = -X02ALF()
      WRITE (NOUT,*)
      WRITE (NOUT,*) '  X     RKAPPA    BETA2        Y        IFAIL'
      WRITE (NOUT,*)
   20 READ (NIN,*,END=40) X, RKAPPA, BETA2
      IF ((RKAPPA.NE.C1) .OR. (BETA2.NE.C2)) THEN
*
*        Initialise array WORK before the the first call to G01MUF and
```

```
*        on subsequent calls when RKAPPA or BETA2 has changed.
*
         IFAIL = 0
*
         CALL G01ZUF(RKAPPA,BETA2,MODE,XL,XU,WORK,IFAIL)
*
      END IF
*
*     Compute the value of the Vavilov density function
*
      IFAIL = 0
*
      Y = G01MUF(X,WORK,IFAIL)
*
      WRITE (NOUT,99999) X, RKAPPA, BETA2, Y, IFAIL
      C1 = RKAPPA
      C2 = BETA2
      GO TO 20
   40 STOP
*
99999 FORMAT (1X,F4.1,5X,F4.1,5X,F4.1,3X,1P,D12.4,I6)
      END
```

9.2 Program Data

```
G01MUF Example Program Data
 2.5  0.4  0.1 : Values of X, RKAPPA and BETA2
```

9.3 Program Results

```
 G01MUF Example Program Results

   X     RKAPPA    BETA2        Y        IFAIL

  2.5      0.4      0.1     8.3675D-02     0
```