NAG Library Routine Document

G05YMF

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

G05YMF generates a uniformly distributed low-discrepancy sequence as proposed by Sobol, Faure or Niederreiter. It must be preceded by a call to one of the initialization routines G05YLF or G05YNF.

2 Specification

```
SUBROUTINE G05YMF(N, RCORD, QUAS, LDQUAS, IREF, IFAIL)
INTEGER           N, RCORD, LDQUAS, IREF(liref), IFAIL
double precision  QUAS(LDQUAS,sdquas)
```

3 Description

Low discrepancy (quasi-random) sequences are used in numerical integration, simulation and optimization. Like pseudorandom numbers they are uniformly distributed but they are not statistically independent, rather they are designed to give more even distribution in multidimensional space (uniformity). Therefore they are often more efficient than pseudorandom numbers in multidimensional Monte Carlo methods.

G05YMF generates a set of points $x^1, x^2, \ldots, x^N$ with high uniformity in the S-dimensional unit cube $I^S = [0,1]^S$.

Let G be a subset of I^S and define the counting function $S_N(G)$ as the number of points $x^i \in G$. For each $x = (x_1, x_2, \ldots, x_S) \in I^S$, let G_x be the rectangular S-dimensional region

$$G_x = [0, x_1) \times [0, x_2) \times \cdots \times [0, x_S)$$

with volume $x_1, x_2, \ldots, x_S$. Then one measure of the uniformity of the points $x^1, x^2, \ldots, x^N$ is the discrepancy:

$$D_N^*\left(x^1, x^2, \ldots, x^N\right) = \sup_{x \in I^S} |S_N(G_x) - N x_1, x_2, \ldots, x_S|.$$

which has the form

$$D_N^*\left(x^1, x^2, \ldots, x^N\right) \leq C_S (\log N)^S + O\left((\log N)^{S-1}\right) \qquad \text{for all} \qquad N \geq 2.$$

The principal aim in the construction of low-discrepancy sequences is to find sequences of points in I^S with a bound of this form where the constant C_S is as small as possible.

The type of low-discrepancy sequence generated by G05YMF depends on the initialization routine called and can include those proposed by Sobol, Faure or Niederreiter. If the initialization routine G05YNF was used then the sequence will be scrambled (see Section 3 in G05YNF for details).

4 References

Brately P and Fox B L (1988) Algorithm 659: Implementing Sobol's Quasirandom Sequence Generator *ACM Trans. Math. Software* **14 (1)** 88–100

Fox B L (1986) Algorithm 647: Implementation and Relative Efficiency of Quasirandom Sequence Generators *ACM Trans. Math. Software* **12 (4)** 362–376

5 Parameters

Note: the following variables are used in the parameter descriptions:

$idim$ = IDIM, the number of dimensions required, see G05YLF or G05YNF

$liref$ = LIREF, the length of IREF as supplied to the initialization routine G05YLF or G05YNF

$sdquas$ = N if RCORD = 1; otherwise $sdquas = idim$

1: N – INTEGER *Input*

On entry: the number of quasi-random numbers required.

Constraint: N ≥ 0 and N + previous number of generated values $\leq 2^{30}$

2: RCORD – INTEGER *Input*

On entry: the order in which the generated values are returned.

Constraint: RCORD = 1 or 2.

3: QUAS(LDQUAS,$sdquas$) – ***double precision*** array *Output*

On exit: contains the N quasi-random numbers of dimension *idim*.

If RCORD = 1, QUAS(i,j) holds the jth value for the ith dimension.

If RCORD = 2, QUAS(i,j) holds the ith value for the jth dimension.

4: LDQUAS – INTEGER *Input*

On entry: the first dimension of the array QUAS as declared in the (sub)program from which G05YMF is called.

Constraints:

if RCORD = 1, LDQUAS $\geq idim$;
if RCORD = 2, LDQUAS $\geq$ N.

5: IREF($liref$) – INTEGER array *Communication Array*

On entry: contains information on the current state of the sequence.

On exit: contains updated information on the state of the sequence.

6: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction 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, if you are 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

If on entry IFAIL = 0 or -1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

On entry, either N < 0 or there have been too many calls to the generator.

IFAIL = 4

On entry, RCORD = 1 and LDQUAS < $idim$.

On entry, RCORD ≠ 1 and LDQUAS < N.

IFAIL = 5

Incorrect initialization. G05YLF must be called prior to G05YMF and IREF must remain unaltered after this call.

7 Accuracy

Not applicable.

8 Further Comments

None.

9 Example

This example calls G05YLF and G05YMF to estimate the value of the integral

$$\int_0^1 \cdots \int_0^1 \prod_{i=1}^{s} |4x_i - 2| dx_1, \, dx_2, \ldots, \, dx_s = 1.$$

In this example the number of dimensions S is set to 8.

9.1 Program Text

```
*     G05YMF Example Program Text
*     Mark 22 Release. NAG Copyright 2006.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
      INTEGER          MN, MIDIM, ORDRC
      PARAMETER        (MN=200,MIDIM=8,ORDRC=1)
      INTEGER          LDQUAS, TDQUAS
      PARAMETER        (LDQUAS=MIDIM,TDQUAS=MN)
      INTEGER          LIREF
      PARAMETER        (LIREF=32*MIDIM+7)
*     .. Local Scalars ..
      DOUBLE PRECISION SUM, TMP, VSBL
      INTEGER          D, GENID, I, IDIM, IFAIL, ISKIP, J, N
*     .. Local Arrays ..
      DOUBLE PRECISION QUAS(LDQUAS,TDQUAS)
      INTEGER          IREF(LIREF)
*     .. External Subroutines ..
      EXTERNAL         G05YLF, G05YMF
*     .. Intrinsic Functions ..
      INTRINSIC        ABS, DBLE
*     .. Executable Statements ..
      WRITE (NOUT,99999) 'G05YMF Example Program Results'
      IDIM = 8
      N = MN
*     Skip the first few variates in the sequence
      ISKIP = 1000
*     Initialize the Sobol generator
      GENID = 1
      IFAIL = 1
      CALL G05YLF(GENID,IDIM,IREF,LIREF,ISKIP,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99997) IFAIL
         GO TO 80
      END IF
*     Generate N quasi-random variates
```

```
      IFAIL = 1
      CALL G05YMF(N,ORDRC,QUAS,LDQUAS,IREF,IFAIL)
      IF (IFAIL.NE.0) THEN
         WRITE (NOUT,99996) IFAIL
         GO TO 80
      END IF
*     Evaluate the function, and sum
      SUM = 0.0D0
      DO 40 I = 1, N
         TMP = 1.0D0
         DO 20 D = 1, IDIM
            TMP = TMP*ABS(4.0D0*QUAS(D,I)-2.0D0)
   20    CONTINUE
         SUM = SUM + TMP
   40 CONTINUE
*     Convert sum to mean value
      VSBL = SUM/DBLE(N)
      WRITE (NOUT,99999)
      WRITE (NOUT,99999) 'Value of integral = ', VSBL
*     Dump the first 10 variates
      WRITE (NOUT,99999)
      WRITE (NOUT,99999) 'First 10 variates'
      DO 60 I = 1, 10
         WRITE (NOUT,99998) I, (QUAS(J,I),J=1,IDIM)
   60 CONTINUE
*
   80 CONTINUE
*
99999 FORMAT (1X,A,F8.4)
99998 FORMAT (1X,I3,20(1X,F8.4))
99997 FORMAT (1X,' ** G05YLF returned with IFAIL = ',I5)
99996 FORMAT (1X,' ** G05YMF returned with IFAIL = ',I5)
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05YMF Example Program Results

 Value of integral =   1.0410

 First 10 variates
   1   0.7197   0.5967   0.0186   0.1768   0.7803   0.4072   0.5459   0.3994
   2   0.9697   0.3467   0.7686   0.9268   0.5303   0.1572   0.2959   0.1494
   3   0.4697   0.8467   0.2686   0.4268   0.0303   0.6572   0.7959   0.6494
   4   0.3447   0.4717   0.1436   0.3018   0.1553   0.7822   0.4209   0.0244
   5   0.8447   0.9717   0.6436   0.8018   0.6553   0.2822   0.9209   0.5244
   6   0.5947   0.2217   0.3936   0.0518   0.9053   0.0322   0.1709   0.7744
   7   0.0947   0.7217   0.8936   0.5518   0.4053   0.5322   0.6709   0.2744
   8   0.0635   0.1904   0.0498   0.4580   0.6240   0.2510   0.9521   0.8057
   9   0.5635   0.6904   0.5498   0.9580   0.1240   0.7510   0.4521   0.3057
  10   0.8135   0.4404   0.2998   0.2080   0.3740   0.5010   0.7021   0.0557
```