NAG Library Routine Document

G05YAF

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

To generate multi-dimensional quasi-random sequences with a uniform probability distribution.

2 Specification

```
SUBROUTINE G05YAF(FCALL, SEQ, ISKIP, IDIM, QUASI, IREF, IFAIL)
INTEGER           ISKIP, IDIM, IREF(2000), IFAIL
double precision  QUASI(IDIM)
LOGICAL           FCALL
CHARACTER*1       SEQ
```

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.

G05YAF generates a set of points $x^1, x^2, \ldots, x^N$ with high uniformity in the S-dimensional unit cube $I^S = [0,1]^S$. One measure of the uniformity is the discrepancy which is defined as follows:

> Given a set of points $x^1, x^2, \ldots, x^N \in I^S$ and a subset $G \subset I^S$, 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 the discrepancy of the points $x^1, x^2, \ldots, x^N$ is
>
> $$D_N^*\left(x^1, x^2, \ldots, x^N\right) = \sup_{x \in I^S} |S_N(G_x) - N x_1, x_2, \ldots, x_S|.$$
>
> The discrepancy of the first N terms of such a sequence 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.

G05YAF generates the low-discrepancy sequences proposed by Sobol, Faure and Niederreiter.

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

1: FCALL – LOGICAL *Input*

On entry: the type of operation to perform.

FCALL = .TRUE.
The first call for initialization, and there is no output via array QUASI.

FCALL = .FALSE.
The sequence has been initialized by a prior call to G05YAF with FCALL = .TRUE.. Random numbers are output via array QUASI.

2: SEQ – CHARACTER*1 *Input*

On entry: the type of sequence to generate.

SEQ = 'S'
A Sobol sequence.

SEQ = 'N'
A Niederreiter sequence.

SEQ = 'F'
A Faure sequence.

Constraint: SEQ = 'S', 'N' or 'F'.

3: ISKIP – INTEGER *Input*

On entry: the number of terms in the sequence to skip on initialization.

ISKIP = 0
All the terms of the sequence are generated.

ISKIP = k
The first k terms of the sequence are ignored and the first term of the sequence now corresponds to the kth term of the sequence when ISKIP = 0.

If SEQ = 'F', ISKIP is not referenced.

Constraint: if SEQ = 'N' or 'S' and FCALL = .TRUE., ISKIP ≥ 0.

4: IDIM – INTEGER *Input*

On entry: the number of dimensions required.

Constraint: $1 \leq$ IDIM ≤ 40.

5: QUASI(IDIM) – ***double precision*** array *Output*

On exit: the random numbers.

If FCALL = .FALSE., QUASI(k) contains the random number for the kth dimension.

6: IREF(2000) – INTEGER array *Communication Array*

On entry/exit: workspace used to contain information between calls to the routine. The contents of this array should not be changed.

7: 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, because for this routine the values of the output parameters may be useful even if IFAIL $\neq 0$ on exit, the recommended value is -1. **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).

Note: G05YAF may return useful information for one or more of the following detected errors or warnings.

Errors or warnings detected by the routine:

IFAIL $= 1$

On entry, ISKIP < 0, if SEQ $=$ 'N' or 'S' and FCALL $=$ .TRUE.,
or IDIM < 1,
or IDIM > 40,
or SEQ $\neq$ 'F', 'N' or 'S'.

IFAIL $= 2$

On entry, the array IREF has not been correctly initialized.

IFAIL $= 3$

The value of ISKIP is too large.

IFAIL $= 4$

There have been too many calls in the sequence.

IFAIL $= 5$

An internal error has occurred within the routine. Please contact NAG.

7 Accuracy

Not applicable.

8 Further Comments

The maximum length of the generated sequences is $2^{29} - 1$, this should be adequate for practical purposes. In the case of the Niederreiter generator G05YAF jumps to the appropriate starting point, while for the Sobol generator it simply steps through the sequence. In consequence the Sobol generator with large values of ISKIP will take a significant amount of time.

9 Example

This example approximates the integral

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

where s is the number of dimensions.

9.1 Program Text

```
*     G05YAF Example Program Text
*     Mark 20 Release. NAG Copyright 2001.
*     .. Parameters ..
      INTEGER          NOUT
      PARAMETER        (NOUT=6)
*     .. Local Scalars ..
      DOUBLE PRECISION SUM, VSBL
      INTEGER          I, IDIM, IFAIL, ISKIP, NTIMES
      LOGICAL          FCALL
      CHARACTER        SEQ
*     .. Local Arrays ..
      DOUBLE PRECISION QUASI(15)
      INTEGER          IREF(2000)
*     .. External Functions ..
      DOUBLE PRECISION FUN
      EXTERNAL         FUN
*     .. External Subroutines ..
      EXTERNAL         G05YAF
*     .. Intrinsic Functions ..
      INTRINSIC        DBLE
*     .. Executable Statements ..
      WRITE (NOUT,*) 'G05YAF Example Program Results'
      IDIM = 15
      NTIMES = 10000
      SEQ = 'S'
      IF (SEQ.EQ.'N' .OR. SEQ.EQ.'n') THEN
         ISKIP = 1000
      ELSE
         ISKIP = 0
      END IF
      FCALL = .TRUE.
      IFAIL = 1
*
      CALL G05YAF(FCALL,SEQ,ISKIP,IDIM,QUASI,IREF,IFAIL)
*
      IF (IFAIL.EQ.0) THEN
         SUM = 0.0D0
         FCALL = .FALSE.
         IFAIL = 0
         DO 20 I = 1, NTIMES
*
            CALL G05YAF(FCALL,SEQ,ISKIP,IDIM,QUASI,IREF,IFAIL)
*
            SUM = SUM + FUN(IDIM,QUASI)
   20    CONTINUE
         VSBL = SUM/DBLE(NTIMES)
         WRITE (NOUT,*)
         WRITE (NOUT,99999) 'Value of integral = ', VSBL
      ELSE
         WRITE (NOUT,99998) IFAIL
      END IF
*
99999 FORMAT (1X,A,F8.3)
99998 FORMAT (1X,' ** G05YAF returned with IFAIL = ',I5)
      END
*
      DOUBLE PRECISION FUNCTION FUN(IDIM,X)
*     .. Scalar Arguments ..
      INTEGER          IDIM
*     .. Array Arguments ..
      DOUBLE PRECISION X(IDIM)
*     .. Local Scalars ..
      DOUBLE PRECISION TMP
      INTEGER          J
*     .. Intrinsic Functions ..
      INTRINSIC        ABS
*     .. Executable Statements ..
      TMP = 1.0D0
      DO 20 J = 1, IDIM
```

```
         TMP = TMP*ABS(4.0D0*X(J)-2.0D0)
   20 CONTINUE
      FUN = TMP
      RETURN
      END
```

9.2 Program Data

None.

9.3 Program Results

```
 G05YAF Example Program Results

 Value of integral =    1.020
```