

NAG Toolbox

nag_rand_init_skipahead_power2 (g05kk)

1 Purpose

nag_rand_init_skipahead_power2 (g05kk) allows for the generation of multiple, independent, sequences of pseudorandom numbers using the skip-ahead method. The base pseudorandom number sequence defined by **state** is advanced 2^n places.

2 Syntax

```
[state, ifail] = nag_rand_init_skipahead_power2(n, state)
[state, ifail] = g05kk(n, state)
```

3 Description

nag_rand_init_skipahead_power2 (g05kk) adjusts a base generator to allow multiple, independent, sequences of pseudorandom numbers to be generated via the skip-ahead method (see the G05 Chapter Introduction for details).

If, prior to calling nag_rand_init_skipahead_power2 (g05kk) the base generator defined by **state** would produce random numbers x_1, x_2, x_3, \dots , then after calling nag_rand_init_skipahead_power2 (g05kk) the generator will produce random numbers $x_{2^n+1}, x_{2^n+2}, x_{2^n+3}, \dots$

One of the initialization functions nag_rand_init_repeat (g05kf) (for a repeatable sequence if computed sequentially) or nag_rand_init_nonrepeat (g05kg) (for a non-repeatable sequence) must be called prior to the first call to nag_rand_init_skipahead_power2 (g05kk).

The skip-ahead algorithm can be used in conjunction with any of the six base generators discussed in the G05 Chapter Introduction.

4 References

Haramoto H, Matsumoto M, Nishimura T, Panneton F and L'Ecuyer P (2008) Efficient jump ahead for F2-linear random number generators *INFORMS J. on Computing* **20(3)** 385–390

Knuth D E (1981) *The Art of Computer Programming (Volume 2)* (2nd Edition) Addison–Wesley

5 Parameters

5.1 Compulsory Input Parameters

1: **n** – INTEGER

n , where the number of places to skip-ahead is defined as 2^n .

Constraint: $n \geq 0$.

2: **state**(:) – INTEGER array

Note: the actual argument supplied **must** be the array **state** supplied to the initialization routines nag_rand_init_repeat (g05kf) or nag_rand_init_nonrepeat (g05kg).

Contains information on the selected base generator and its current state.

5.2 Optional Input Parameters

None.

5.3 Output Parameters

- 1: **state**(:) – INTEGER array
Contains updated information on the state of the generator.
- 2: **ifail** – INTEGER
ifail = 0 unless the function detects an error (see Section 5).

6 Error Indicators and Warnings

Errors or warnings detected by the function:

ifail = 1

Constraint: $n \geq 0$.

ifail = 2

On entry, **state** vector has been corrupted or not initialized.

ifail = 3

On entry, cannot use skip-ahead with the base generator defined by **state**.

ifail = 4

On entry, the **state** vector defined on initialization is not large enough to perform a skip-ahead (applies to Mersenne Twister base generator). See the initialization function `nag_rand_init_repeat` (g05kf) or `nag_rand_init_nonrepeat` (g05kg).

ifail = -99

An unexpected error has been triggered by this routine. Please contact NAG.

ifail = -399

Your licence key may have expired or may not have been installed correctly.

ifail = -999

Dynamic memory allocation failed.

7 Accuracy

Not applicable.

8 Further Comments

Calling `nag_rand_init_skipahead_power2` (g05kk) and then generating a series of uniform values using `nag_rand_dist_uniform01` (g05sa) is equivalent to, but more efficient than, calling `nag_rand_dist_uniform01` (g05sa) and discarding the first 2^n values. This may not be the case for distributions other than the uniform, as some distributional generators require more than one uniform variate to generate a single draw from the required distribution.

9 Example

This example initializes a base generator using `nag_rand_init_repeat` (g05kf) and then uses `nag_rand_init_skipahead_power2` (g05kk) to advance the sequence 2^{17} places before generating five variates from a uniform distribution using `nag_rand_dist_uniform01` (g05sa).

9.1 Program Text

```
function g05kk_example

fprintf('g05kk example results\n\n');

genid = nag_int(1);
subid = nag_int(1);
seed = [nag_int(1762543)];

% Initialise the generator to a repeatable sequence
[state, ifail] = g05kf( ...
    genid, subid, seed);

% Advance the sequence 2**n places
n = nag_int(17);
[state, ifail] = g05kk( ...
    n, state);

% Generate nv variates from a uniform distribution
nv = nag_int(5);
[state, x, ifail] = g05sa( ...
    nv, state);

% Display the variates
disp(x);
```

9.2 Program Results

```
g05kk example results

0.7357
0.3521
0.4188
0.0046
0.0365
```
