THE GPSS/360 RANDOM NUMBER. GENERATOR

Harry Felder*
Department of Mathematics
Princeton University
Princeton, New Jersey

Introduction

Computers are increasingly being used in the simulation of processes which can be considered to be systems of interacting probabilistic elements. Because of the complicated interrelations between these probabilistic elements, analytic solutions are usually impossible, leaving as the only tractable procedure Monte Carlo techniques.

Since such methods require a plentiful supply of random numbers, a number of studies of various uniform random number generators (URNG) have been published in the past few years. (1), (2), (3), (5), (7), (8). Such studies are complicated by the fact that virtually all URNG use a definite scheme which is able to be repeated, giving the same sequence of "random numbers". Obviously, the numbers generated are not truly random. If, however, the scheme's pattern of generation does not affect the desired statistical properties of. the sequence, the appearance of randomness is satisfactory. Since each problem using random numbers assumes an almost unique set of statistical properties to be satisfied by the random number sequence, the problem of designing a suitable testing procedure for a URNG is difficult at best.

This study of yet another URNG has been designated with a particular user of URNG in mind: the General Purpose Simulation System/360 user. By eliminating from consideration such uses of the tested generator as large Monte Carlo calculations on nuclear energy problems, and considering only the typical uses of URNG by GPSS/360 problems, a more suitable testing procedure was devised.

Appropriately enough, the URNG studied was the one included in GPSS/360.** (Actually, there are eight URNG in GPSS/360, but they use identical generation schemes.) Each of the eight GPSS/360 URNG has one parameter, a "multiplier", which the user may alter, causing a new sequence of random numbers to be generated. The goal of this study was to locate multipliers whose sequences would meet a number of criteria desirable if the sequences are to be used in GPSS/360 simulations.

Properties of a URNG which are Important in Simulation Studies

Two almost obvious restrictions on a random number generation scheme are that it be deterministic and have a reasonably short generation time. A deterministic generator is one that follows a definite scheme in generating pseudo-random numbers; the scheme must be able to be repeated, giving the same sequence of numbers each time. This is desirable so that changes in a model can be made, and the new model tested against identical "random fluctuations". The second restriction results from the total number of times the URNG may be called upon in a single run of a simulation model.

Since it is the initial part of a sequence of random numbers that is used in debugging of a model, bizarre behavior of, say, the first 5000 elements of a sequence can cause time-consuming searches for the cause of the unexpectedly biased output statistics. Poor local behavior (behavior of less than 2000 consecutive numbers) of a sequence can lengthen the time it takes before a model's statistics stabilize. Because of these factors, bizarre local or initial behavior is undesirable in a URNG used in simulation, even though this behavior may be statistically insignificant when 100,000 to 1,000,000 numbers of a sequence are employed. Another important property is that the sequence actually used in a run of a model, whether 250 or 50,000 numbers long, should be a 'suitable approximation' to a sequence of uniform random numbers. Statistics on the sequence should not drift to extreme values as the sequence is extended from 250 numbers to 50,000.

Testing Procedure Followed (A Working Definition of 'Suitable Approximation')

It was decided to test no more than the first 50,000 numbers of a sequence generated by a given multiplier because it was felt that this was a reasonable upper limit to the number of times a single generator is called upon in a single simulation run. This decision to test only the first 50,000 numbers is in contrast to the larger numbers--100,000 to 1,000,000--tested in previous studies of URNG. To check local behavior, vital statistics on the sequence of 50,000 numbers were computed on 25 trials of 2,000 numbers and 50 trials of 1,000 numbers. To check initial behavior of the sequence, the first 5,000 numbers of the sequence were further tested in 20 trials of 250. To check on whether the sequence would present a good approximation to a uniform random sequence, no matter how many elements a user

---

* This report was written while at IBM during the summer of 1968.

**The random number generator tested was from GPSS/360, Version 1, Modification Level 2.

would call on, statistics were computed on the first 250, 1000, 2000, 5000, 10000, 15000, 25000, 35000, and 50000 numbers.

The following statistical tests were used, since they were thought to test the properties of the URNG which a typical GPSS user would intuitively assume the generator possessed:

1. A χ^2 test for uniformity over the range of the URNG. The number of intervals the range was divided into was decided by applying the Mann-Wald criteria (6) to maximize the power of the test at the 95% level of confidence. At this level of confidence, the number of intervals used in the test would be $4\left(\frac{2(n-1)^2}{2.706}\right)^{1/5}$, where n is the number of elements of the sequence of random numbers used in the computation of the statistic.

2. A χ^2 test for uniformity of pairs of consecutive numbers. The range was divided into ten equal intervals, and the number of pairs falling into each element of the resulting 10 x 10 grid was computed. This procedure checked only the high order decimal digit of the random numbers for pairwise uniformity.

3. The Autocorrelations of the sequence for lags 0, 1, 2, 3, 4 and 5. If the sequence of random numbers is denoted by (x_i), i=1,...,n then for lag h, the autocorrelation of the sequence is defined as

$$A_h = \frac{1}{n-h} \sum_{i=1}^{n-h} X_i X_{i+h}$$

   It follows that A_0 is the second moment of the sequence. For a uniform distribution in the interval (0,1) with h=0, A_h should be approximately normally distributed with mean = .25 and standard deviation = $\sqrt{13n-19h}/12(n-h)$. For h=0, A_h should be approximately normally distributed with mean = 1/3 and standard deviation = $.29814/\sqrt{n}$.

4. A χ^2 test for runs above and below the mean. A run below the mean of length j is defined to be when j consecutive numbers of the sequence are below the mean, and this subsequence of length j is bracketed by numbers above the mean. Runs above the mean are defined similarly. For example, if the mean is .5, then the sequence
   .1 .2 .6 .4 .3 .7 .6 .2 .1 .4
   has a run below the mean of length 2, followed by a run above the mean length 1, below of length 2, above of length 2, and below of length 3.

   If the total length of the sequence is n, the total number of runs expected is (n+1)/2, while the expected number of runs above (below) the mean of length j is $(n-j+3)/2^{j+2}$.

   The χ^2 statistic was computed by comparing the actual distribution of runs to the expected distribution.

5. Comparison of the mean of the batch with its expected distribution. A mean should be normally distributed with standard deviation = $\sigma_u/\sqrt{n}$ where σ_u is the standard deviation of the uniform distribution and n is the number of random numbers used to compute the mean.

A Description of the GPSS/360 URNG

The following material, substantially from the IBM GPSS/360 User's Manual (4), describes the random number generators of GPSS/360. For each generator there is one parameter available for each user--the initial value of the appropriate element of the multiplier array. The default value is 1. At present the user is not permitted to choose an even multiplier or a multiplier larger than 99999. With each cycle of the generator, a random integer and a random decimal fraction is generated. If the decimal fraction random number is .XYZABC, the integer random number is ABC.

GPSS/360 has eight 'pseudo-random' number generators, specified as RN1 through RN8. Each produces the same sequence of numbers, unless the RMULT card is used to produce unique sequence. Furthermore, the sequence will be duplicated exactly from run to run if a JOB card is used--hence the designation 'pseudo-random'. The availability of eight independent sources of random numbers permits reproduction of exact sequences for certain variables from run to run; for example, the pattern of input transactions may be kept unchanged when comparing alternative system designs in a series of runs. Also, the ability to alter the sequences of the random number generators provides up to eight unique sequences of random numbers. The following describes the mechanism by which random numbers are created.

The GPSS/360 program maintains 8 x 1 arrays: (1) The base number array contains eight words, each different. The first of these words is called the seed. (2) The multiplier array contains eight words, one for each random number generator. Each multiplier is initially one, unless an alternate initial multiplier is supplied by the analyst (by means of the RMULT card). (3) The index array contains eight words, one for each random number generator; each is initially zero.

When a random number is called for, the following procedure is used:

1. The appropriate word in the index array (depending upon which RN# is used) points to one of the eight numbers in the base number array. Since the index array

words are initially zero, the first base number used will be the seed. All eight random number generators use this common seed.

2. The appropriate number in the multiplier array (again depending upon which RN# is used) is multiplied by the base number chosen in step 1.

3. The low order 32 bits of this product are then sometimes subjected to another transformation. If the highest order bit of these 32 bits is 1, the two's complement of these 32 bits replace the original 32 low order bits of the 64 bit product.

4. The low order 31 bits of this possibly partially complemented product are stored in the appropriate word of the multiplier array, to be used the next time a random number is called for.

5. Three bits of the high-order 16 bits of the product after step 3 are stored in the appropriate word of the index array, for future use. This number (0-7) points to one of eight words of the base number to be used the next time a random number is called for.

6. (a) If the random number required is a fraction, the middle 32 bits of the product after step 3 are divided by 10^6, and the remainder becomes the six-digit fractional random number.

   (b) If the random number required is an integer, the middle 32 bits of the product after step 3 are divided by 10^3, and the remainder becomes the three digit random number.

Unlike the power residue method, the GPSS/360 URNG does not use the previously generated random number as the new multiplier, but rather uses a separate, although overlapping, segment of the 64 bit product. Also no indexing is used in a power residue method of generating random numbers. This feature of the GPSS/360 URNG, a second "randomizing factor" in the scheme, does not make the generator appreciably slower. Another difference between this generator and power residue methods is that the new scheme does not lend itself to analytic methods of analysis. In fact, even the period of the generator for different multipliers is not known.

Results of the Tests

A number of multipliers were tested, including the default multiplier, 1. No pattern in the location of satisfactory multipliers was found. The following eight multipliers appear satisfactory for use by GPSS/360 users:

31
37
743
6352
751860533
1088707833
1366595713
1626487679

Of these eight initial multipliers only 31, 37 and 743 can be used in the present version of GPSS/360. The multiplier 6352 violates the condition that the multipliers must be odd, and the remaining four are over five decimal digits in length.

Tables 1-8 summarize the number of times a multiplier's two sequences (decimal fraction and integer) "failed" the various tests. A failure was considered to be when a statistic was outside the appropriate 95% confidence limits. 5% of the statistics calculated should be failures. None of the above multipliers show unusual or systematic deviations from the 5% figure.

There can be no assurances that these multipliers will not fail other statistical tests. For some purposes, other multipliers may be better suited. Hopefully, the testing procedure used will satisfy the needs of most GPSS/360 users; it probably will not satisfy all.

Table 9 summarizes the behavior of the sequence generated by the default multiplier, 1. Because of its repeated failure on the χ^2 test of the decimal random numbers for uniformity and its systematic failures on the mean test and various autocorrelation tests when more than 15000 numbers in the integer sequence were considered, cautious use of it should be made.

Conclusion

It was felt that a simulation user rarely makes use of n-tuples of random numbers, $n > 2$. For this reason, no tests on the n-tuple behavior of the sequences were done. This is a departure from previous studies. Before the generator could be adopted by non-simulation users, it would be desirable to check its n-tuple behavior.

Throughout this study it was assumed that a user would use a sequence for only one purpose. For example, he would not use every third number generated for a single type of decision and the remaining numbers for other purposes. Since no tests were performed on a sequence consisting of every third random number generated, there is no guarantee that such a sequence would be suitably random. For critical applications of the URNG, a user should use one of the eight generators for each type of decision if he wishes to make use of the random number sequences under the conditions they were studied.

References

1) Gorenstein, S., Testing a Random Number Generator, C.ACM 10(1967), pp 111-118.

2) Hull, T.E., and Dobell, A. R., Random Number Generators, SIAM Review 4 (1962), pp 230-254.

3) Hull, T. E., and Dobell, A. R., Mixed Congruential Random Number Generators for Binary Machines, J. ACM 11 (1964), pp 31-40.

4) International Business Machines Corporation, General Purpose Simulation System/360 User's Manual, H20-0326-2, 1968.

5) MacLaren, M. Donald, and Marsaglia, George, Uniform Random Number Generators, J. ACM 12 (1965), pp 83-89.

6) Mann, H. B., and Wald, A., On the Choice of the Number of Class Intervals in the Application of the Chi Square Test, Annals of Mathematical Statistics 13 (1942), pp 302-317.

7) VanGender, A., Some New Results in Pseudo-Random Number Generation, J. ACM 14 (1967), pp 785-792.

8) Westlake, W. J., A Uniform Random Number Generator Based on the Combination of Two Congruential Generators, J. ACM 14 (1967), pp 337-340.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05			1																	
10000	1	10000	.05																				
15000	1	15000	.05												1								
25000	1	25000	.05																				
35000	1	35000	.05											1									
50000	1	50000	.05					1						1									
250	20	5000	1	1	1	3	2	1	2	1	2	1	1	3.		1	1	1	1	1	1	1	1
1000	50	50000	2.5	3	4	4	2	4	5	4	4	3	2	4	5	3	2	4	4	5	3	3	5
2000	25	50000	1.25	2	1	1	2				1				3	2	1	2	2	2	2	2	2

Table 1 : Summary of Testing of the Multiplier 31.
The entries under each test indicate the number of failures of that test under each procedure.
The auto correlation tests (Tests #5-10) are not independent of each other.

Test 1 One Dimensional Uniformity
2 - Uniformity of Pairs
3 - Mean
4 - Runs
5 - Auto correlation, Lag 0

Test 6 - Auto correlation, Lag 1
7 - Auto correlation, Lag 2
8 - Auto correlation, Lag 3
9 - Auto correlation, Lag 4
10 - Auto correlation, Lag 5

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05																				
10000	1	10000	.05																				
15000	1	15000	.05																				
25000	1	25000	.05																				
35000	1	35000	.05																				
50000	1	50000	.05																				
250	20	5000	1			2	1	1	2	1	2	2	2	3	1		1	1	1	1	1	1	1
1000	50	50000	2.5	2	2		6		1	1				1	2		1	1		2	3	1	1
2000	25	50000	1.25	1	3	1			1	2		1	1	3	3	3		3	3	2	2	3	3

Table 2 : Summary of Testing of the Multiplier 37.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05											1									
10000	1	10000	.05	1																			
15000	1	15000	.05																				
25000	1	25000	.05																			1	
35000	1	35000	.05																				
50000	1	50000	.05																				
250	20	5000	1	2	1									2	1	2	1	1	2	2	2	2	2
1000	50	50000	2.5	4	2	4	6	1	2	1	3	1	1	2		1	2	2	2	1	3	3	2
2000	25	50000	1.25	2	1	1	2							1	1	1	1	1	1	1	2	1	1

Table 3 : Summary of Testing of the Multiplier 743.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250		250	.05																				
1000		1000	.05																				
2000		2000	.05																				
5000		5000	.05																				
10000		10000	.05																				
15000		15000	.05																				
25000		25000	.05																				
35000		35000	.05					1		1	1		1										
50000		50000	.05	1	1									1	1								
250		5000	1			2	1	1	2	1	2	2	2	3	1		1	1	1	1	1	1	1
1000		50000	2.5	3	2	2	3	2	1	1	2	2	2	5	3	1	1	2		1	2	1	1
2000		50000	1.25	1	2							1		3	2		2						

Table 4 · Summary of Testing of the Multiplier 6352.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05																				
10000	1	10000	.05																				
15000	1	15000	.05																				
25000	1	25000	.05																				
35000	1	35000	.05																				
50000	1	50000	.05																				
250	20	5000	1	1	2	1	1	1		1		1	1			2	1	1	1	1	1	4	3
1000	50	50000	2.5	5	1		4				1	1		4	1	2	3	2	2	2	3	2	3
2000	25	50000	1.25	5	1		1								1	1	2	1	1		1		0

Table 5 : Summary of Testing of the Multiplier 751860533.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05																				
10000	1	10000	.05																				
15000	1	15000	.05																				
25000	1	25000	.05																				
35000	1	35000	.05																				
50000	1	50000	.05											1	1								
250	20	5000	1	2	1		1									1	1	1	1		2	1	1
1000	50	50000	2.5	3	6	4	3	2	1	5	3	1	1	1	4		5	1		1		1	2
2000	25	50000	1.25		3		1			2				2	2	1	1	1			2		3

Table 6 : Summary of Testing of the Multiplier 1088707833.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05																				
5000	1	5000	.05																				
10000	1	10000	.05	1																			
15000	1	15000	.05																				
25000	1	25000	.05																				
35000	1	35000	.05																				
50000	1	50000	.05											1									
250	20	5000	1		1										2	1	1			1		1	1
1000	50	50000	2.5		4	5		3	3	4	4	2	3	4	5	3	2	4	4	5	3	3	5
2000	25	50000	1.25	2	2	2		2	2	1	2	1	1		3	2	1	2	2	2	2	2	2

Table 7 : Summary of Testing of the Multiplier 1366595713.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05											1									
2000	1	2000	.05		1																		
5000	1	5000	.05		1																		
10000	1	10000	.05																				
15000	1	15000	.05																				
25000	1	25000	.05																				
35000	1	35000	.05																				
50000	1	50000	.05		1																		
250	20	5000	1	1	1		2	1	1		1				1	1	2	1	2	2	1	1	1
1000	50	50000	2.5	1	3	2	2	1	2		1		2	4	1	1	3	2		3	1	1	4
2000	25	50000	1.25		3		1			1				1	1	1	1	1	1	1	1	1	2

Table 8 : Summary of Testing of the Multiplier 1626487679.

Size of Batch	Number of Trials	Length of Tested Sequence	Failures Expected per Test	Tests of Integers Sequence										Tests of Decimal Fraction Sequence									
				1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
250	1	250	.05																				
1000	1	1000	.05																				
2000	1	2000	.05											1						1			
5000	1	5000	.05																				
10000	1	10000	.05											1									
15000	1	15000	.05											1									
25000	1	25000	.05			1			1		1												
35000	1	35000	.05			1		1	1		1	1	1										
50000	1	50000	.05			1		1			1												
250	20	5000	1		1	2		1	2	2	1	1	1	2			1	1		1		1	1
1000	50	50000	2.5		4	6	3	5	4	5	5	4	5	2	4	3	4	2	1	1	2	2	4
2000	25	50000	1.25	1		4	1	2	2	2	2	2	3	3		2	2	1	2	2	3	3	2

Table 9 : Summary of Testing of the Multiplier 1.