

Wayne State University
DigitalCommons@WayneState

Theoretical and Behavioral Foundations of
Education Faculty Publications

Theoretical and Behavioral Foundations

5-1-2011

Bias in Monte Carlo Simulations Due To Pseudo-Random Number Generator Initial Seed Selection

Jack C. Hill

Beaumont Health System, hill3jc@cmich.edu

Shlomo S. Sawilowsky

Wayne State University, shlomo@wayne.edu

Recommended Citation

Hill, J. C., & Sawilowsky, S. S. (2011). Bias in Monte Carlo simulations due to pseudo-random number generator initial seed selection. *Journal of Modern Applied Statistical Methods*, 10(1).
Available at: http://digitalcommons.wayne.edu/coe_tbf/3

This Article is brought to you for free and open access by the Theoretical and Behavioral Foundations at DigitalCommons@WayneState. It has been accepted for inclusion in Theoretical and Behavioral Foundations of Education Faculty Publications by an authorized administrator of DigitalCommons@WayneState.

Bias in Monte Carlo Simulations Due To Pseudo-Random Number Generator Initial Seed Selection

Jack C. Hill
Beaumont Health System

Shlomo S. Sawilowsky
Wayne State University

Pseudo-random number generators can bias Monte Carlo simulations of the standard normal probability distribution function with initial seeds selection. Five generator designs were initial-seeded with values from 10000_{HEX} to $1\text{FFFF}_{\text{HEX}}$, estimates of the mean were calculated for each seed, the distribution of mean estimates was determined for each generator and simulation histories were graphed for selected seeds.

Key words: Pseudo-random generator, number sequences, initial seed, bias, Monte Carlo.

Introduction

It is possible, using a vetted pseudo-random number generator and a specific initial seed, to produce number sequences that may have very nonrandom-like characteristics; however, restarting the algorithm with a new seed will produce an excellent number sequence. Park and Miller (1988) asserted that coding errors in the algorithm might be responsible, although, in a vetted generator, coding errors are unlikely. (See Fishman (1995) for descriptions of other possible reasons for this phenomenon.)

Pseudo-Random Number Generator Design

Lehmer (1951) suggested a simple computer algorithm to generate a random number sequence whose period is limited only by the arithmetic registers of the digital computer and the parameters selected. This linear recurrence (congruential) generator is the basic and most widely used class of random number generators. Linear recurrence (congruential) generators are of the form

$$Z_i = \sum (A_s Z_{i-1} + B) \text{ mod } M$$
$$\text{for } 0 \leq i < 2^{31} - 1 \text{ and } 1 \leq s \leq X$$
(1.1)

where $0 < A_1, A_2, \dots, A_x < M$; A, B and M are integers, and X is the order of the generator (the number of non-zero A_s). The most widely used first-order, $X = 1$, form of the linear congruential generator is the multiplicative (linear) congruential generator:

$$Z_i = (AZ_{i-1} + B) \text{ mod } M$$
$$\text{for } 0 \leq i < 2^{31} - 1 \text{ and } Z_0 \text{ is the initial seed}$$
(1.2)

where $0 < A < M$; A and M are integers and $B = 0$.

After the terminology and symbols of L'Ecuyer (1993), the following short notation defines random number generators:

$$\text{XXX} (M, [A_{1,2,3\dots n}], B, Z_0),$$

where XXX is the abbreviation of the type of generator (e.g., MCG = Multiplicative Congruential Generator; MRG = Multiple Recursive Generator; and MWC = Multiply-With-Carry multiplicative congruential generator), M is the modulus, A is the array of multipliers or coefficients, B is the increment, and Z_0 is the initial seed.

Jack Hill is a BI Database Architect and adjunct faculty in statistics and quantitative analysis. Email him at: hill3jc@cmich.edu. Shlomo Sawilowsky is a professor of educational statistics, and editor of *JMASM*. Email him at: shlomo@wayne.edu.

PSEUDO-RANDOM NUMBER INITIAL SEED BIAS IN MONTE CARLO SIMULATIONS

MCG (2^{31} , 65539, 0, 1), also known as RANDU, is an example of a generator whose number sequences are recognized as poorly randomized and whose period is relatively short at 2^{29} (Park & Miller, 1988):

$$Z_i = 65539 * Z_{i-1} \text{ mod } 2^{31}$$

for $1 \leq i \leq 2^{31} - 1$ and Z_0 the initial seed

(1.3)

The set, S, of all numbers, period, $p = 2^{29}$, can be generated using an initial seed, $Z_0 = 1$ (or any other value, $0 < Z_0 \leq 2^{31}-1$). An initial seed (Z_0) determines the first number in the set S where values will be drawn, and each subsequent value is determined by recursively evaluating equation 1.3.

MCG ($2^{31}-1$, 16807, 0, 1), also known as Minimum Standard (Park & Miller, 1988), used in RANGEN (Sawilowsky & Blair, 1987; Fahome & Sawilowsky, 2001) is included as an example of a generator whose number sequences are generally recognized as good for a $2^{31}-1$ period generator. Park and Miller considered this an example of a good minimum standard generator and it has endured decades of study and testing:

$$Z_i = 16807 * Z_{i-1} \text{ mod } (2^{31} - 1)$$

for $1 \leq i \leq 2^{31} - 1$ and Z_0 the initial seed

(1.4)

where Z_i is the current value, Z_{i-1} is the prior value. When $i = 0$, Z_0 is the initial seed, the generator has been generally accepted as a good random number generator and has been used in numerous Monte Carlo studies. The set, S, of all numbers, period, $p = 2^{31}-1$, can be generated using an initial seed, $Z_0 = 1$ (or any other value, $0 < Z_0 \leq 2^{31}-1$). An initial seed, Z_0 , determines the first number in the set, S, where values will be drawn, and each subsequent value is determined by recursively evaluating equation 1.4.

MCG (231-1, 630360016, 0, 1) is also reported to be a good MCG (Entacher, 1998):

$$Z_i = 630360016 * Z_{i-1} \text{ mod } (2^{31} - 1)$$

for $1 \leq i \leq 2^{31} - 1$ and Z_0 the initial seed

(1.5)

The set, S, of all numbers, period, $p = 2^{31}-1$, can be generated using an initial seed, $Z_0 = 1$ (or any other value, $0 < Z_0 \leq 2^{31}-1$). An initial seed, Z_0 , determines the first number in the set, S, where values will be drawn, and each subsequent value is determined by recursively evaluating equation 1.5.

L'Ecuyer (1993) reported testing several higher-order linear congruential generators or multiple recursive generators having the form

$$Z_i = (A_1 Z_{i-1} + A_2 Z_{i-2} + \dots + A_x Z_{i-x}) \text{ mod } M$$

for $0 \leq i < 2^{31} - 1$ and $Z_{0,1,2,\dots,x-1}$ are the initial seeds

(1.6)

where $0 < A_1, A_2, \dots, A_x < M$; A and M are integers and X is the generator's order.

One of L'Ecuyer's best performing multiple recursive generators is MRG (231-1, [2001982722, 1412284257, 1155380217, 1668339922], 0, 1), also known as LECUYER, a 4th-order multiple recursive generator whose output sequence is reported to be good:

$$Z_i = (2001982722 * Z_{i-1} + 1412284257 * Z_{i-2} + 1155380217 * Z_{i-3} + 1668339922 * Z_{i-4}) \text{ mod } (2^{31} - 1)$$

for $1 \leq i \leq 2^{31}$ and $Z_{0,1,2,\dots,x-1}$ are the initial seeds

(1.7)

The set, S, of all numbers, period, $p \equiv 2^{158}$, can be generated using an initial seed, $Z_0 = 1$ (or any other value, $0 < Z_0 \leq 2^{31}-1$). An initial seed, Z_0 , determines the first number in the set, S, where values will be drawn, and each subsequent value is determined by recursively evaluating Equation 1.7.

Marsaglia (1994b) proposed a variation of the multiplicative congruential generator called multiply-with-carry of the form

$$Z_i = (A_1 Z_{i-4} + A_2 Z_{i-3} + A_3 Z_{i-2} + A_4 Z_{i-1} + C) \text{ mod } M$$

for $0 \leq i < 2^{31} - 1$ and $Z_{0,1,2,3}$ are initial seeds

(1.8)

where $0 < A_1, A_2, A_3, A_4 < M$; A and M are integers (Gentle, 2003). Marsaglia's design generates a 64-bit sum of four products and a carry; these 64 bits are separated into two 32-bit words. The upper 32-bits become the new carry, C, the lower 32-bits (modulo M) is the new Z_i and each subsequent value is determined by recursively evaluating equation 1.8.

MWC (232, [211111111111, 1492, 1776, 5115], C, 1) (Marsaglia, 1994a), is implemented in a subroutine library known as MOTHER. (The version considered here is due Miller, 1995, adapted and updated to Fortran 90 by Blair, 1999, in a subroutine library called BFRA. It is an implementation of Luescher's, 1994, algorithm and James', 1994, Fortran 77 coding of the "Luxury" generator.) It is included as an example of a popular form of the multiply-with-carry, random number generators:

$$Z_i = (211111111111 * Z_{i-1} + 1492 * Z_{i-2} + 1776 * Z_{i-3} + 5115 * Z_{i-4} + C) \text{ mod } 2^{32}$$

for $0 \leq i < 2^{31} - 1$ and $Z_{0,1,2,3}$ are initial seeds

(1.9)

The set, S, of all numbers, period, $p = 2^{158}$, can be generated using an initial seed, $Z_0 = 1$ (or any other value, $0 < Z_0 \leq 2^{31} - 1$). An initial seed, Z_0 , determines the first number in the set, S, where values will be drawn, and each subsequent value is determined by recursively evaluating equation 1.9.

Pseudo-Random Number Sequences

L'Ecuyer and Hellekalek (1997) suggested a mental-model of a huge roulette wheel to visualize the sequence of unscaled numbers from a generator such as RANGEN (equation 1.4) where the sequenced numbers appear once, in the range of 1 to $2^{31} - 1$. The order of the number sequence is determined by the RANGEN algorithm based on the previous number (seed). Spinning the roulette wheel would afford the researcher a random starting

point (initial seed) in the number sequence; however, because the number sequence is calculated based on the previous number, the sequence is truly deterministic in nature; it only has the appearance of randomness.

These number sequences can be easily scaled to the unit interval, $U[0,1]$, by dividing each number by P, the Period of the generator. RANGEN's period = $2^{31} - 1$. Resetting the algorithm to the same initial seed reproduces the identical random number-sequence. Individual numbers from these sequences are used as indices that determine the locations on a probability density function or in the data table where samples should be drawn.

It should be noted that, due to their deterministic nature, different initial seeds could generate overlapping number sequences. For example, a hypothetical random number generator using an initial seed S_0 calculated the following number sequence where $S_0 = N_0$:

$$N_0, N_1, N_2, N_3, N_4, N_5, N_6, N_7, N_8, N_9, N_{10} \dots N_n$$

If a different initial seed, $S_0 = N_3$, is used, the following number sequence would be calculated:

$$N_3, N_4, N_5, N_6, N_7, N_8, N_9, N_{10} \dots N_n$$

These number sequences clearly overlap beginning at N_3 ; however, the Monte Carlo simulations reported herein were not adjusted for possible overlapping number sequences.

Methodology

The purpose of this study was to determine the bias introduced into Monte Carlo simulation studies resulting from initial seed selection. The random number sequences of five random number generators were analyzed. Each random number generator was seeded with bit-patterns sequenced from an initial value of 10000_{Hex} (65,536) to a final value of 1FFFF_{Hex} (131,071).

The process was as follows: Use each random number generator's scaled number sequence from the unit interval ($U[0,1]$, Seed₀ = S_0). Select a sample, S, of sample size, $N = 10$, from the standard normal probability density function. Calculate the sample mean, $X_S = 1/N \sum S$, of these values. Continue sampling for T trials and calculate the overall mean, $X_T = 1/T \sum X_S$, of

PSEUDO-RANDOM NUMBER INITIAL SEED BIAS IN MONTE CARLO SIMULATIONS

T trials. For the standard normal probability density function, with a large number of trials, T, it is expected that the sample statistic, X_T , will converge on the population parameter mean, $\mu=0$, and the population standard deviation, $\sigma=1$.

Results

The population mean and standard deviation were estimated for each 1000_{HEX} (65,536) \leq Initial Seed $\leq 1\text{FFFF}_{\text{HEX}}$ (131,071) using five random number generators at Trials = 1,000, 10,000, 100,000 and 1,000,000 (Table 1.1). With the exception of MOTHER, the average mean estimate Z score = 0.000 and average standard deviation estimate = 1.000. The distribution of

mean estimate Z scores for each random number generator at each trial is normal-like, mean = 0, and standard deviation = 1.00 with minimum Z scores of -5.07 to -3.34 and maximum Z scores of 3.09 to 4.59 (Table 1.2).

RANGEN Pseudo-Random Number Generator

RANGEN's mean estimate distributions at Trials = 1,000, 10,000, 100,000 and 1,000,000 are typical of the five random number generators studied (Figures 1.1, 1.2, 1.3, 1.4). Each graph shows the normalized SPSS histogram of all mean estimates from Monte Carlo simulations using each initial seed with the standard normal distribution superimposed.

Table 1.1: Mean and Standard Deviation Estimates for Each Random Number Generator of Monte Carlo Simulations of the Standard Normal Probability Density Function Using All Initial Seeds from 10000Hex (65,536) through 1FFFFHex (131,072); Sample Size = 10, Trials = 1,000, 10,000, 100,000 and 1,000,000

Trials	1,000		10,000		100,000		1,000,000	
RNG	Mean	StdDev	Mean	StdDev	Mean	StdDev	Mean	StdDev
RANGEN	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
ENTACHER	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
LECUYER	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
MOTHER	-0.001	1.000	-0.005	1.000	-0.008	1.000	-0.005	1.000
RANDU	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000

Table 1.2: The Maximum and Maximum Mean Estimates For Each Random Number Generator of Monte Carlo Simulations of the Standard Normal Probability Density Function Using All Initial Seeds from 10000Hex (65,536) through 1FFFFHex (131,072); Sample Size = 10, Trials = 1,000, 10,000, 100,000, and 1,000,000

Trials	1,000		10,000		100,000		1,000,000	
RNG	Min	Max	Min	Max	Min	Max	Min	Max
RANGEN	-3.83	4.04	-4.46	4.18	-4.31	4.12	-3.34	3.09
ENTACHER	-4.35	4.05	-4.08	4.41	-4.14	4.15	-4.34	4.35
LECUYER	-4.01	4.01	-4.51	4.23	-4.13	4.32	-4.00	4.07
MOTHER	-4.02	4.10	-4.51	4.42	-5.07	4.58	-4.32	4.59
RANDU	-3.86	4.18	-4.21	4.04	-3.76	3.79	-3.86	3.84

Figure 1.1: RANGEN Random Number Generator
 A Normalized SPSS Histogram Graph of Monte Carlo Simulations of Mean Estimates of the Standard Normal Probability Density Function Using All Initial Seeds from 10000_{Hex} (65,536) through 1FFFF_{Hex} (131,072); Sample Size = 10, Trials = 1,000

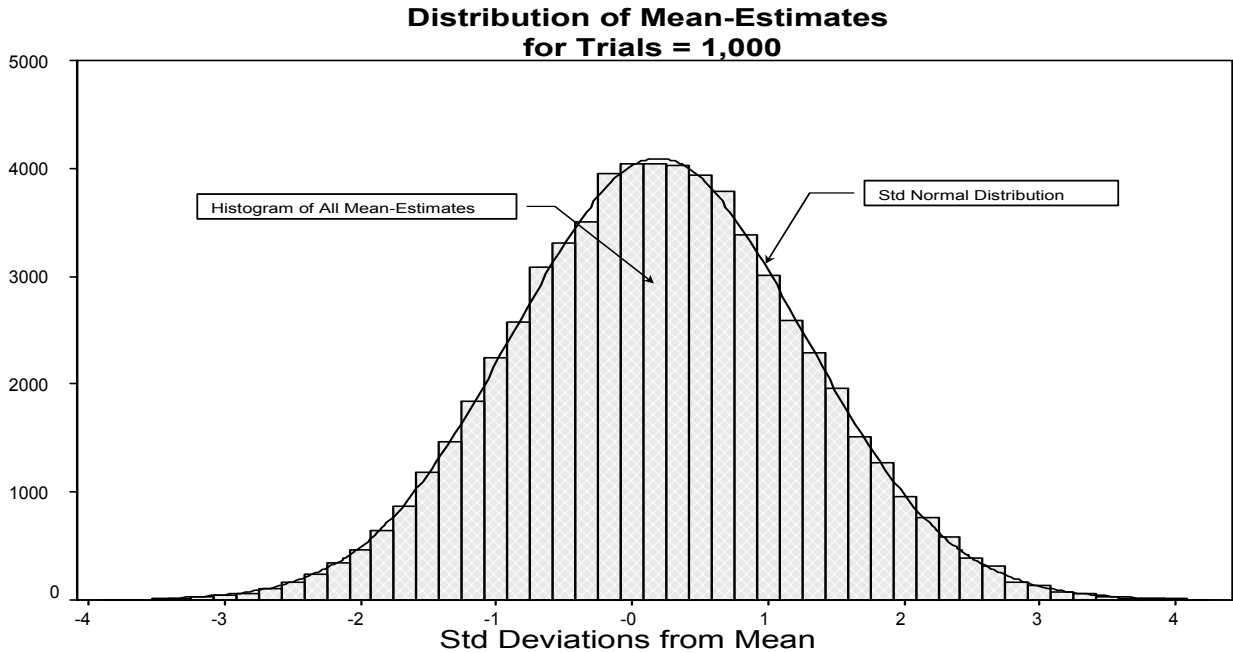


Figure 1.2: RANGEN Random Number Generator
 A Normalized SPSS Histogram Graph of Monte Carlo Simulations of Mean Estimates of the Standard Normal Probability Density Function Using All Initial Seeds from 10000_{Hex} (65,536) through 1FFFF_{Hex} (131,072); Sample Size = 10, Trials = 10,000

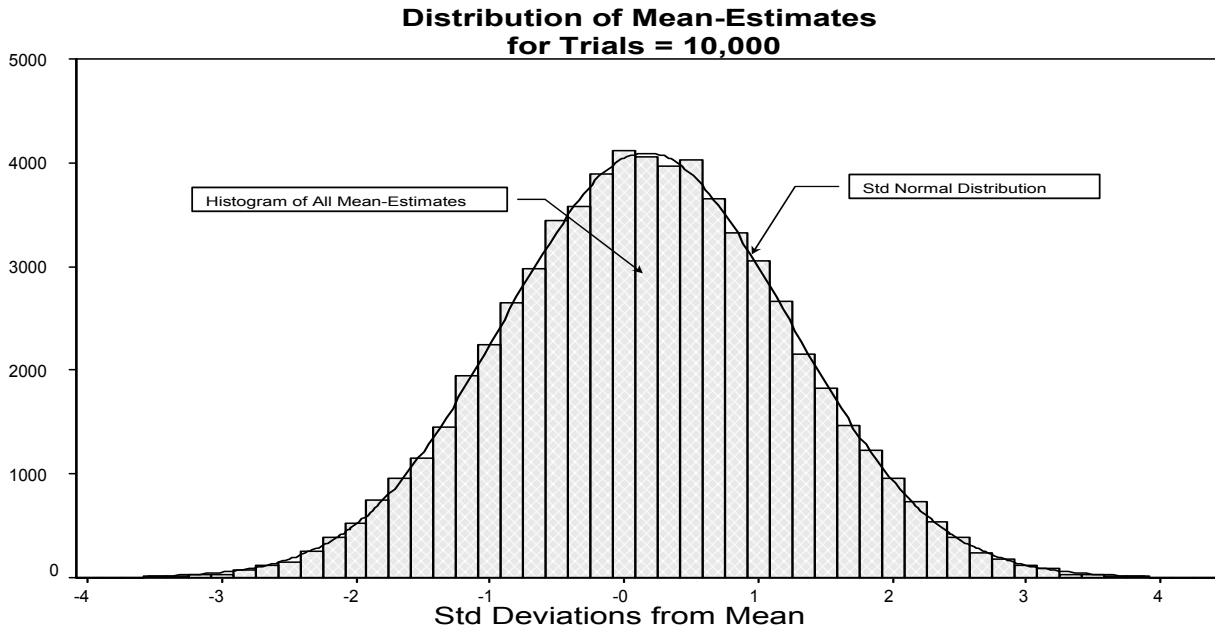


Figure 1.3: RANGEN Random Number Generator

A Normalized SPSS Histogram Graph of Monte Carlo Simulations of Mean Estimates of the Standard Normal Probability Density Function Using All Initial Seeds from 10000_{Hex} (65,536) through 1FFFF_{Hex} (131,072); Sample Size = 10, Trials = 100,000

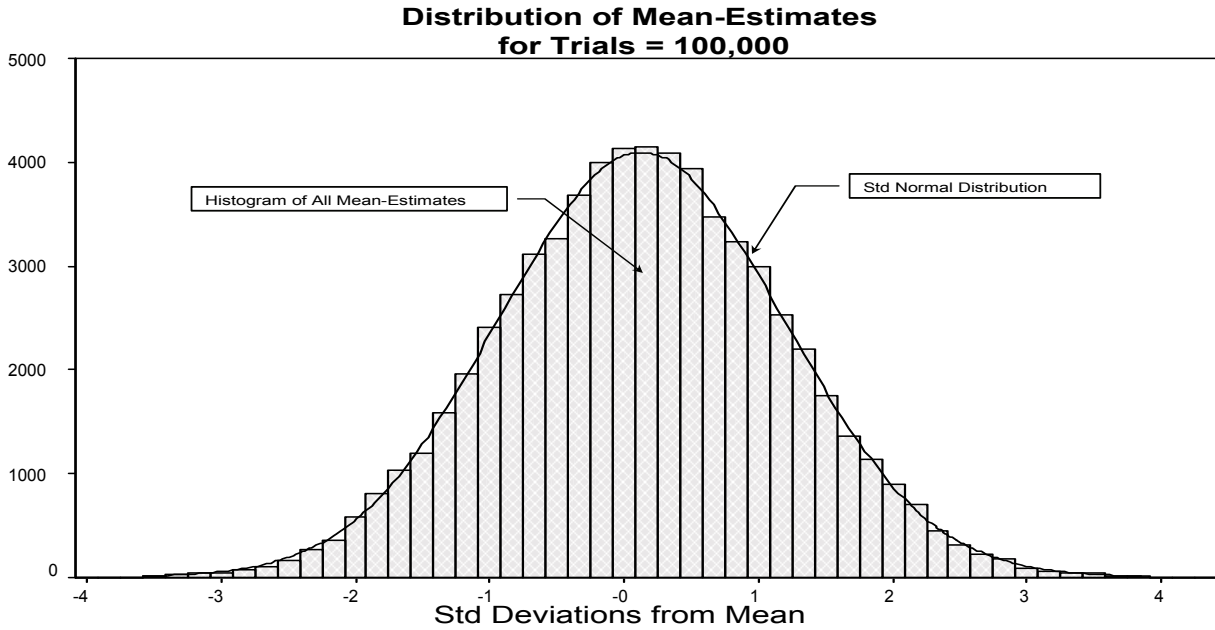


Figure 1.4: RANGEN Random Number Generator

A Normalized SPSS Histogram Graph of Monte Carlo Simulations of Mean Estimates of the Standard Normal Probability Density Function Using All Initial Seeds from 10000_{Hex} (65,536) through 1FFFF_{Hex} (131,072); Sample Size = 10, Trials = 1,000,000

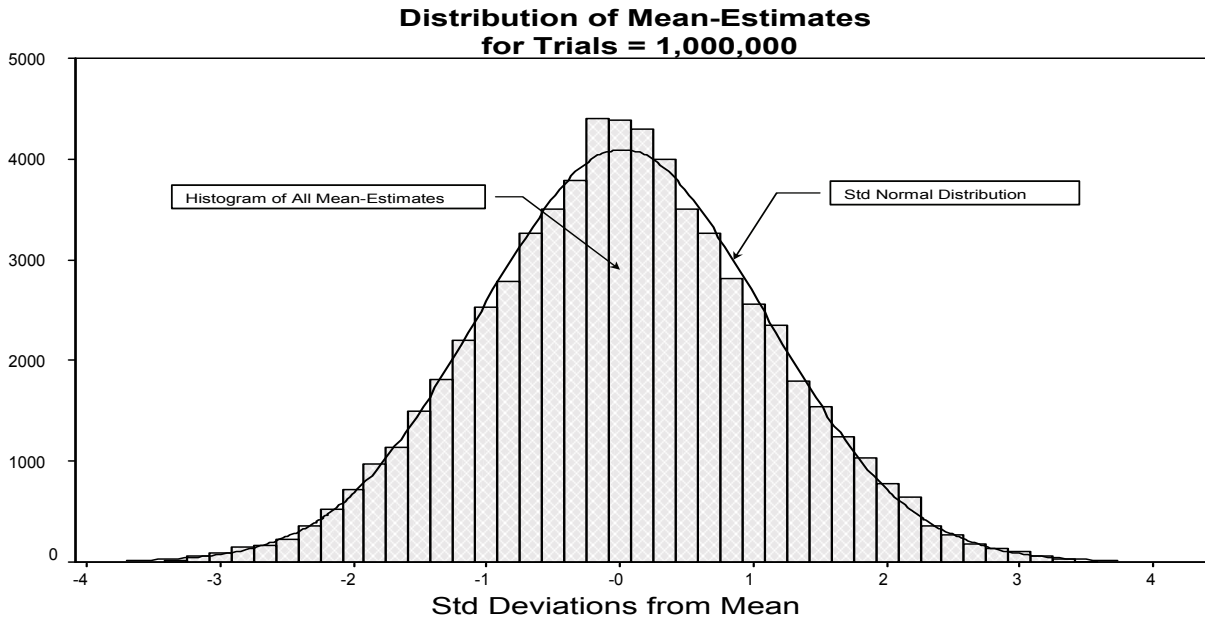


Table 1.3: RANGEN Random Number Generator

SPSS Extreme Values Table of Normalized Mean Estimates from Monte Carlo Simulations of the Standard Normal Probability Density Function Using All Initial Seeds, $S_i = 10000\text{Hex}$ through 1FFFFHex at Trials = 1,000, 10,000, 100,000, and 1,000,000

Trials	1,000		10,000		100,000		1,000,000	
Deviation	Seed	z-Score	Seed	z-Score	Seed	z-Score	Seed	z-Score
Above	66184	4.04	93552	4.18	92157	4.12	72114	3.09
	114634	3.80	68018	3.95	91955	4.11	70120	3.08
	87579	3.80	90064	3.84	91223	3.94	115292	3.04
	98710	3.73	85895	3.73	117600	3.94	118753	3.04
	118038	3.73	73471	3.73	116943	3.88	95913	3.03
Below	108848	-3.65	96630	-4.04	82111	-3.92	76920	-3.31
	79125	-3.69	114546	-4.14	121739	-4.16	68326	-3.31
	73787	-3.73	101156	-4.18	113374	-4.16	79462	-3.32
	87518	-3.74	105875	-4.34	65647	-4.17	77341	-3.32
	102745	-3.83	125434	-4.46	105966	-4.31	106675	-3.34

RANGEN Monte Carlo Simulation (MCS) History Graphs

For RANGEN’s distribution of mean estimates at Trial = 1,000,000, fifteen initial seeds were chosen, five from each extreme (the worst) and five from the center (the best). The mean estimates’ initial seeds were used to capture the cumulative (running) average at 10,000-trial intervals of a Monte Carlo simulation of the standard normal probability distribution function. These data were normalized and graphed, (Figure 1.5), as a graphical history of the Monte Carlo simulation’s mean estimate at 10,000-trial intervals. RANGEN’s cumulative mean-estimates undulated wildly ± 15 standard deviations for the first 100,000 trials and then converged on their respective region of the mean estimate distribution.

Individual RANGEN MCS history graphs for the two worst initial seeds (Table 1.3), 106675 and 72114, and closest to the true mean, 117656, were isolated. The RANGEN MCS history graph (Figure 1.6) for initial seed, $S_i = 106675$, demonstrated an immediate

negative departure from zero of -6 standard deviations at trials = 20,000; a recovery to -2 standard deviations at trials = 100,000; a sharp drop to -5 standard deviations at trials = 140,000; another recovery attempt at trials = 190,000; and a final convergence on -3 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The RANGEN MCS history graph (Figure 1.7) for initial seed, $S_i = 117656$, demonstrated an initial negative departure from zero, -9 standard deviations, at trials = 10,000; a recovery, -0.5 standard deviations, at trials = 80,000; and a final convergence on 0.0 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated some wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The RANGEN MCS history graph (Figure 1.8) for initial seed, $S_i = 72114$, demonstrated an initial negative departure from

PSEUDO-RANDOM NUMBER INITIAL SEED BIAS IN MONTE CARLO SIMULATIONS

zero, -13 standard deviations at trials = 10,000; a recovery, +9 standard deviations, at trials = 50,000; a sharp drop, +2 standard deviations, at trials = 100,000; and a final convergence on +3 standard deviations. Individual 10,000-trial intervals, designated as ♦ on the graph, demonstrated wildly fluctuating mean estimates of ±15 standard deviations throughout the simulation run.

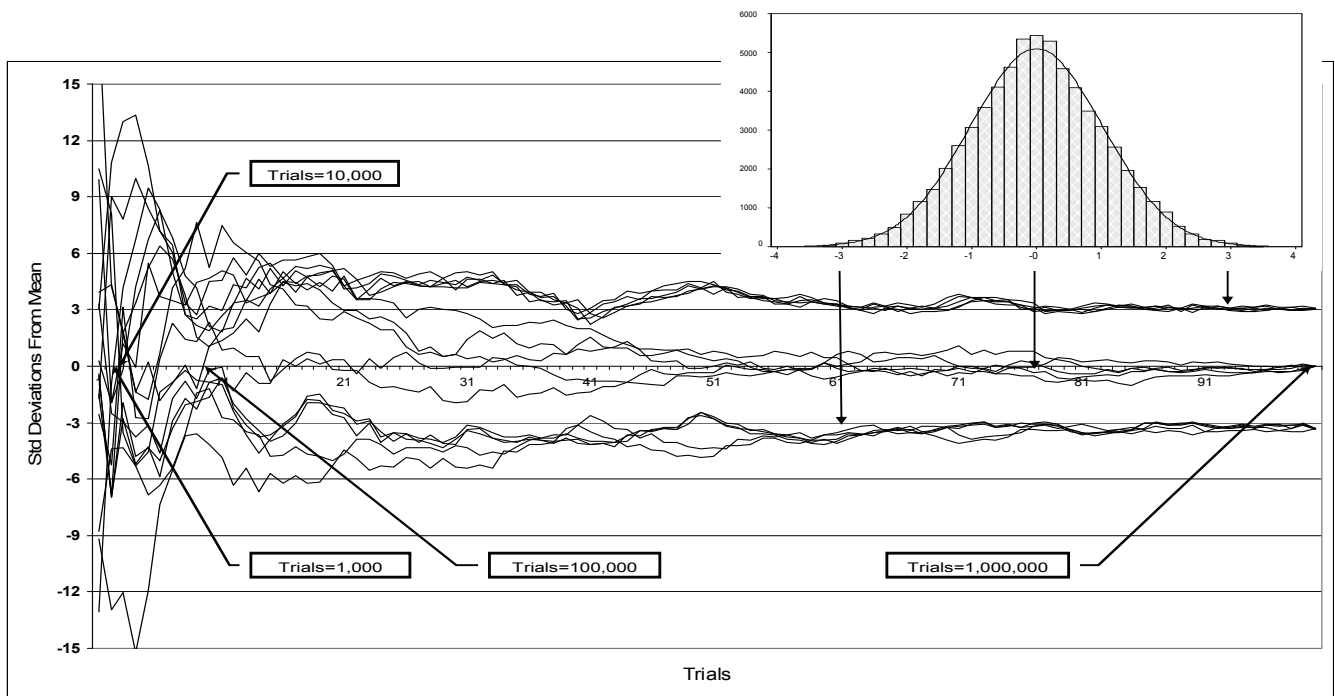
MOTHER Pseudo-Random Number Generator

For MOTHER's distribution of mean estimates at Trial = 1,000,000, fifteen initial seeds were chosen, five from each extreme, the worst, and five from the center, the best. The mean estimates' initial seeds were used to capture the cumulative (running) average at

10,000-trial intervals of a Monte Carlo simulation of the Standard Normal probability distribution function. These data were normalized and graphed, (Figure 2.1), as graphical history of the Monte Carlo simulation's mean-estimate at 10,000-trial intervals. MOTHER's cumulative mean estimates undulated wildly ±15 standard deviations for the first 100,000 trials and then converged on their respective region of the mean estimate distribution.

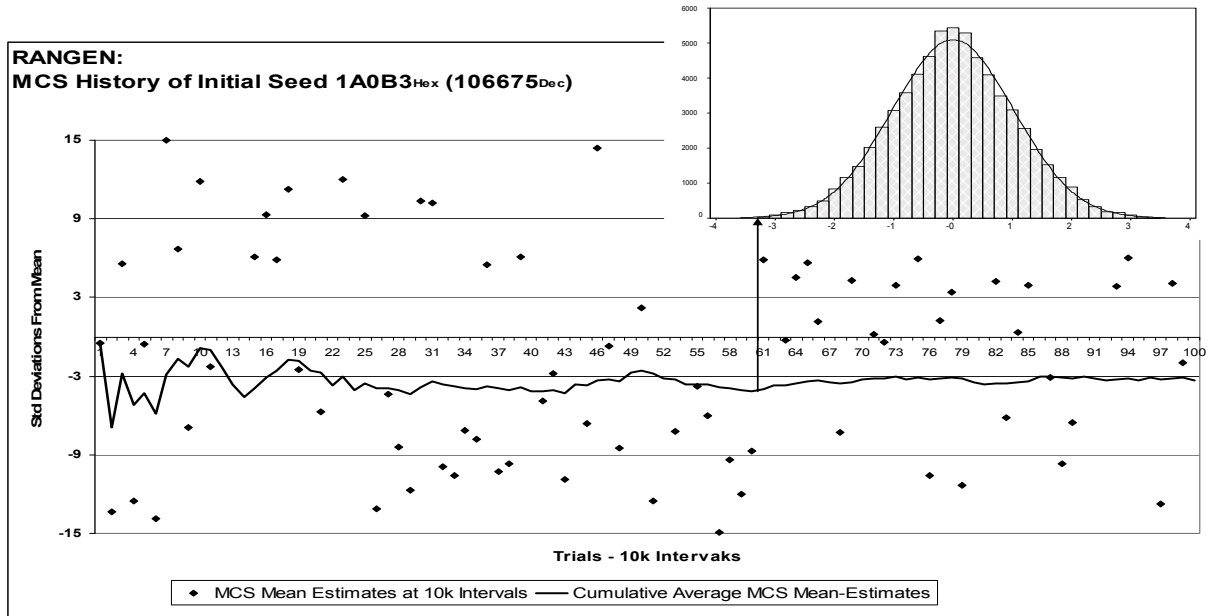
Individual MOTHER MCS history graphs for the two worst initial seeds, 83815 and 112145, and closest to the true mean, 126281, were isolated. The MOTHER MCS history graph (Figure 2.2) for initial seed, $S_i = 83815$, demonstrated an immediate negative

Figure 1.5: RANGEN Random Number Generator



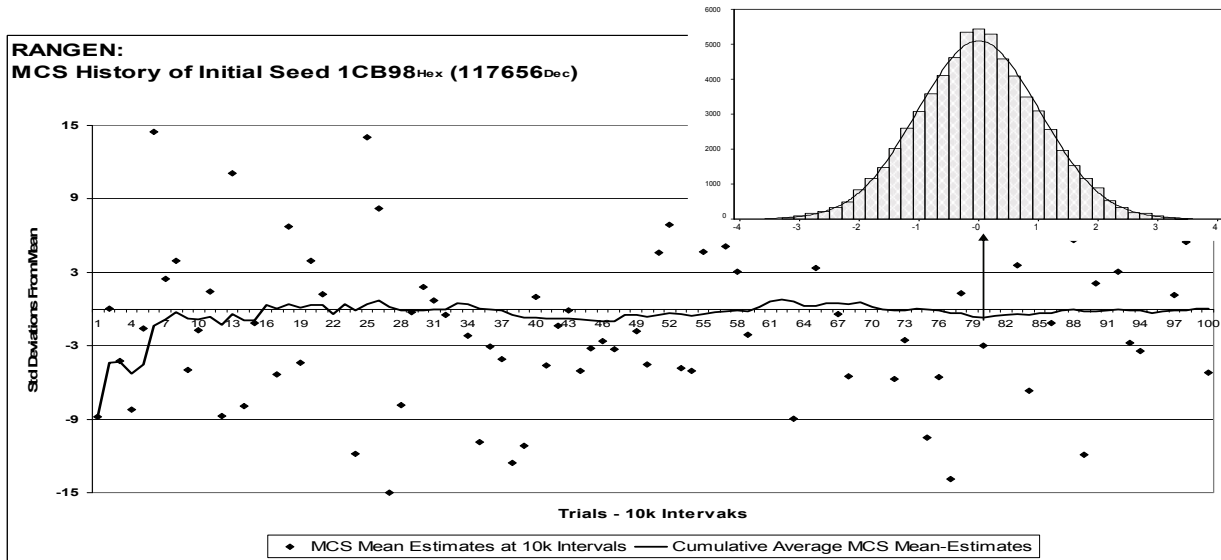
Each line in the graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Fifteen initial seeds (five each: high, middle and low mean estimates) are shown at 10k intervals, total trials = 1,000,000. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Down arrows correlate the -3σ , 0 , and $+3\sigma$ points of the standard normal probability density function to the horizontal cumulative averages and their deviations from the mean.

Figure 1.6: RANGEN Random Number Generator



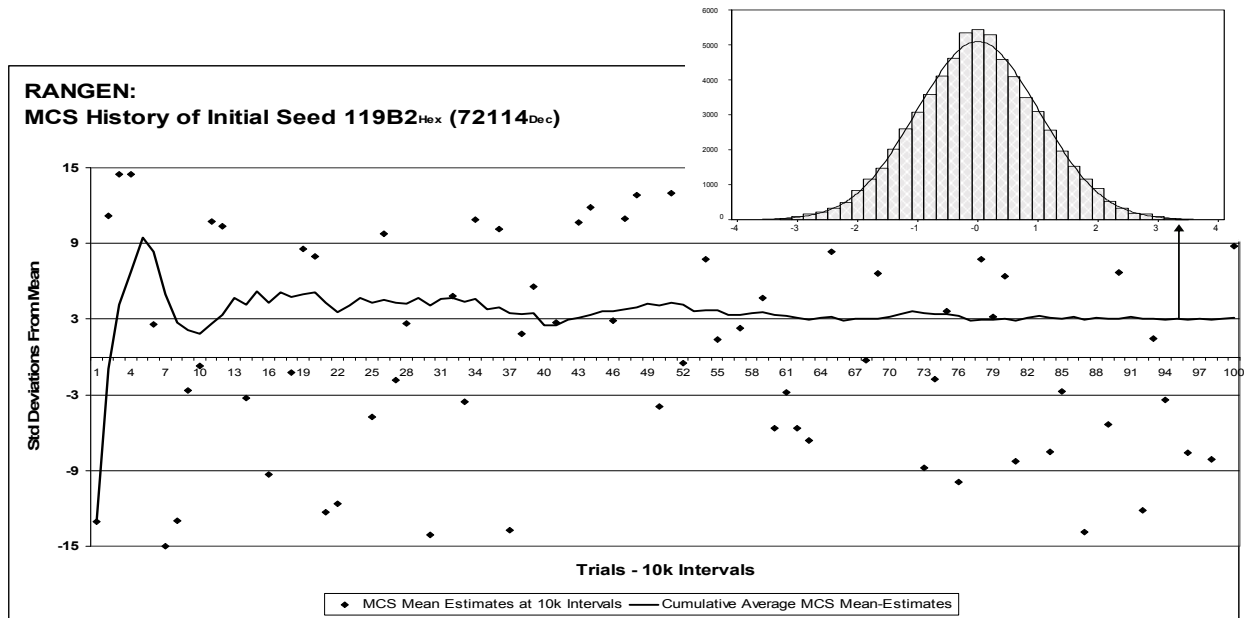
This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 106675, is the worst mean estimate below the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 106675, on the standard normal probability density function.

Figure 1.7 RANGEN Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 117656, is the best mean estimate. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 117656, on the standard normal probability density function.

Figure 1.8: RANGEN Random Number Generator



This Graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 72114, is the worst mean estimate above the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 72114, on the standard normal probability density function.

departure from zero, -6 standard deviations, at trials = 10,000; a recovery to -3 standard deviations at trials = 50,000; and a final gradual convergence on -4 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The MOTHER MCS history graph (Figure 2.3) for initial seed, $S_i = 126281$, demonstrated an initial negative departure from zero, -15 standard deviations, at trials = 10,000 and a final convergence on 0.0 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated some wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

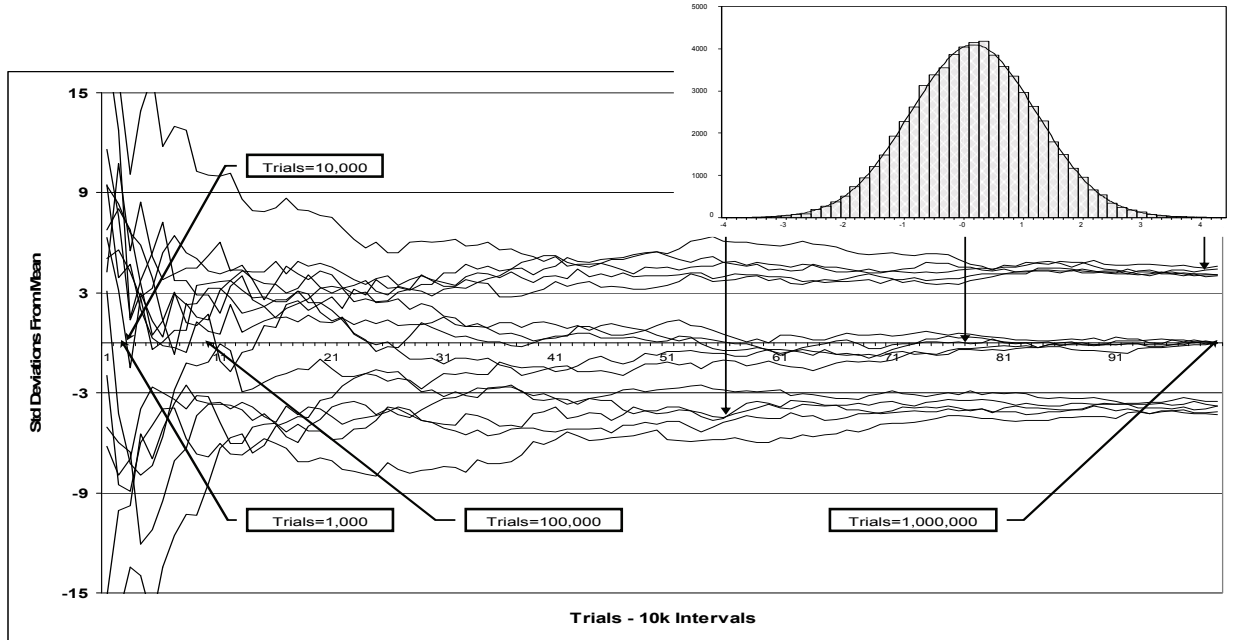
The MOTHER MCS history graph (Figure 2.4) for initial seed, $S_i = 112145$, demonstrated an initial departure from zero, 15 standard deviations, at trials = 10,000 and a final convergence on +4 standard deviations.

Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

ENTACHER Pseudo-Random Number Generator

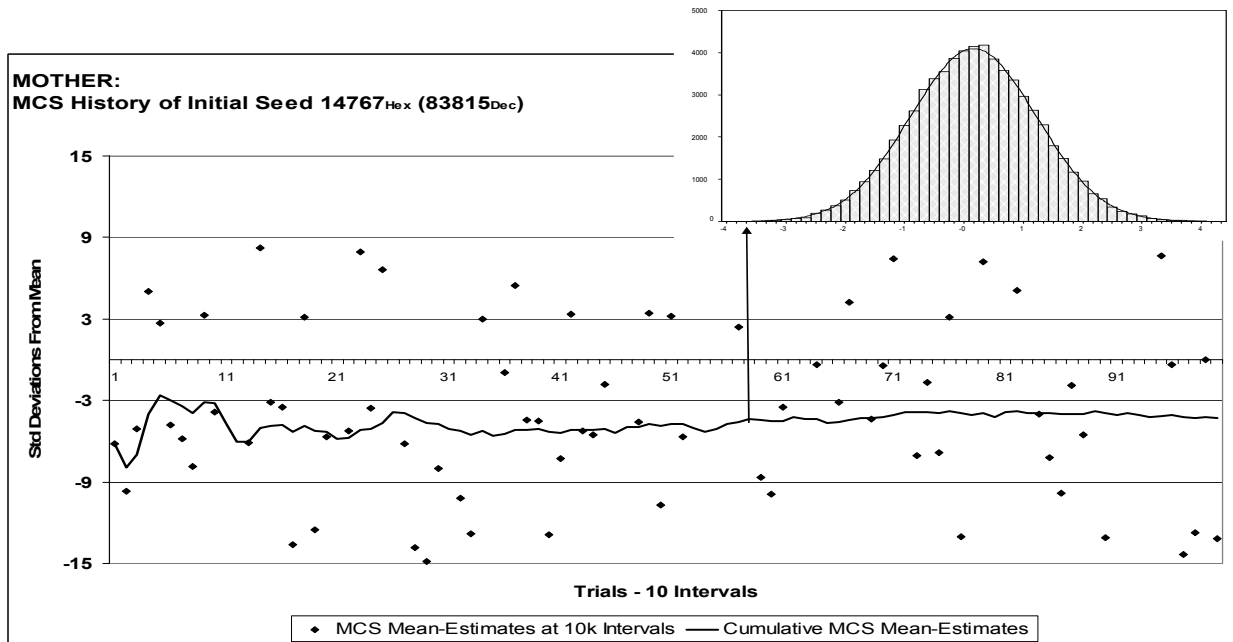
For ENTACHER's distribution of mean estimates at Trial = 1,000,000, fifteen initial seeds were chosen, five from each extreme (the worst) and five from the center (the best). The mean estimates' initial seeds were used to capture the cumulative (running) average at 10,000-trial intervals of a Monte Carlo simulation of the Standard Normal probability distribution function. These data were normalized and graphed, (Figure 3.1), as graphical history of the Monte Carlo simulation's mean-estimate at 10,000-trial intervals. ENTACHER's cumulative mean estimates undulated wildly (± 15 standard deviations) for the first 100,000 trials and then converged on their respective region of the mean estimate distribution.

Figure 2.1: MOTHER Random Number Generator



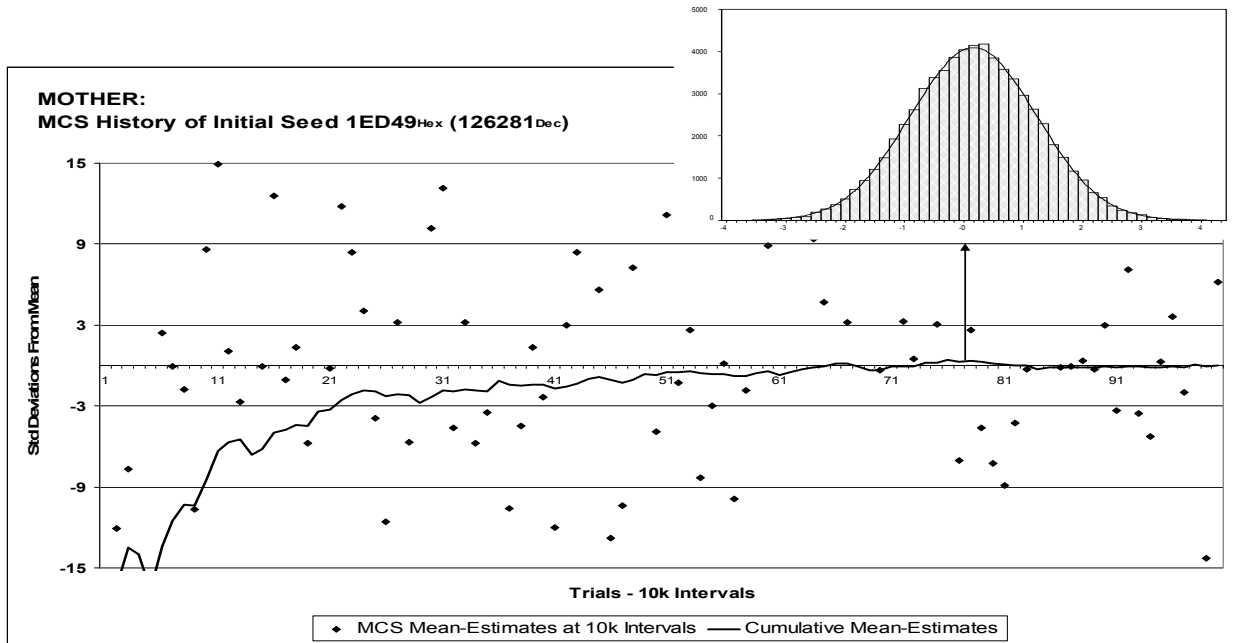
Each line in the graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Fifteen initial seeds (five each: high, middle and low mean estimates) are shown at 10k intervals, total trials = 1,000,000. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Down arrows correlate the 3σ , 0, and $+3\sigma$ points of the standard normal probability density function to the horizontal cumulative averages and their deviations from the mean.

Figure 2.2: MOTHER Random Number Generator



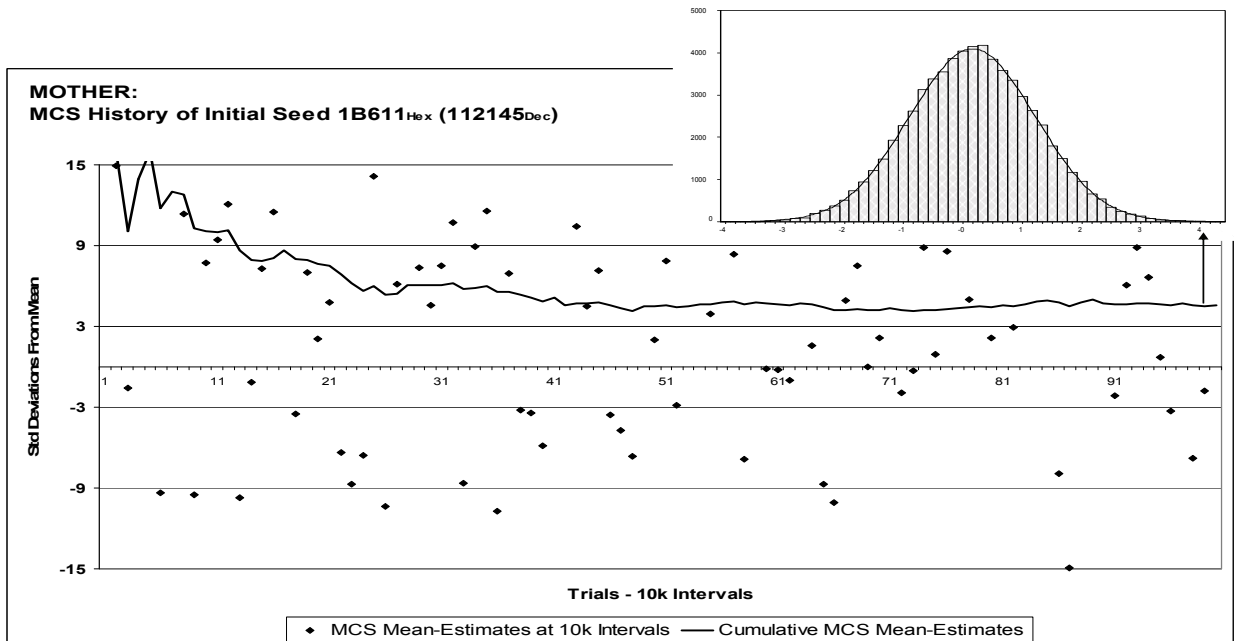
This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 83815, is the worst mean estimate below the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 83815, on the standard normal probability density function.

Figure 2.3: MOTHER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 126281, is the best mean estimate. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 126281, on the standard normal probability density function.

Figure 2.4 MOTHER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 112145, is the worst mean estimate above the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 112145, on the standard normal probability density function.

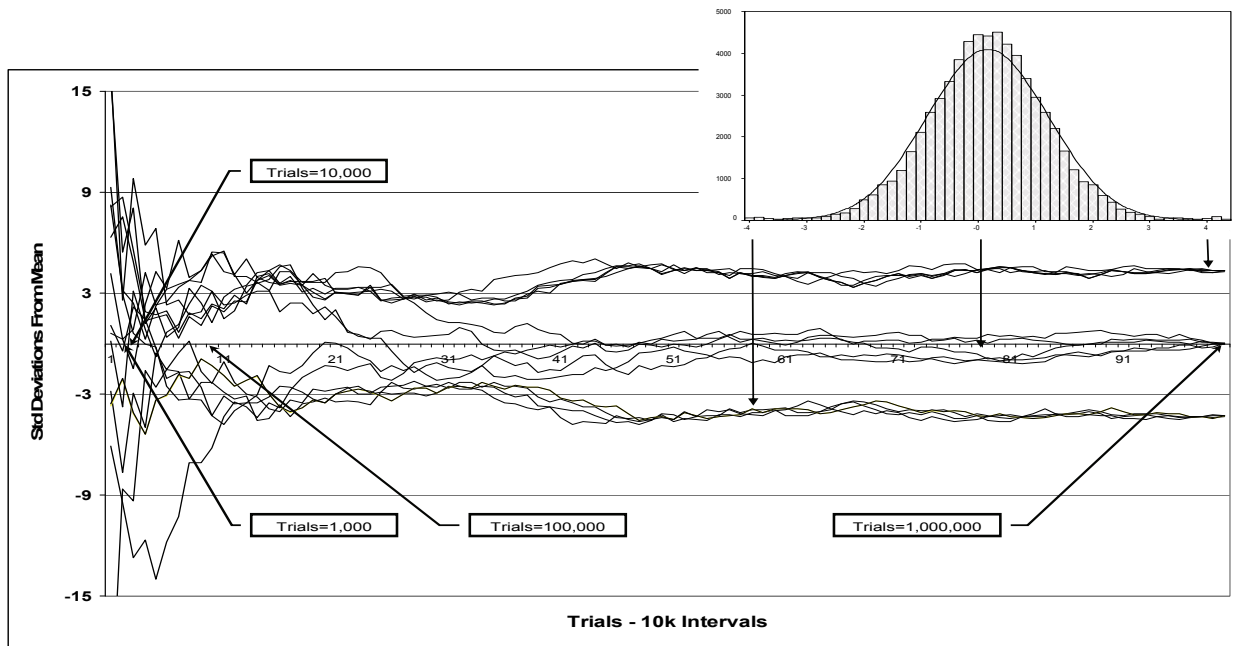
Individual ENTACHER MCS history graphs for the two worst initial seeds, 82315 and 73489, and closest to the true mean, 120803, were isolated. The ENTACHER MCS history graph (Figure 3.2) for initial seed, $S_i = 82315$, demonstrated an immediate departure from zero, -15 standard deviations, at trials = 10,000; a recovery to -2 standard deviations at trials = 30,000; and a final convergence on -4 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The ENTACHER MCS history graph (Figure 3.3) for initial seed, $S_i = 120803$, demonstrated an immediate departure from zero, +8 standard deviations, at trials = 10,000; a recovery to +0.5 standard deviations at trials =

30,000; a sharp jump to +5 standard deviations at trials = 80,000; and a final convergence on 0.0 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

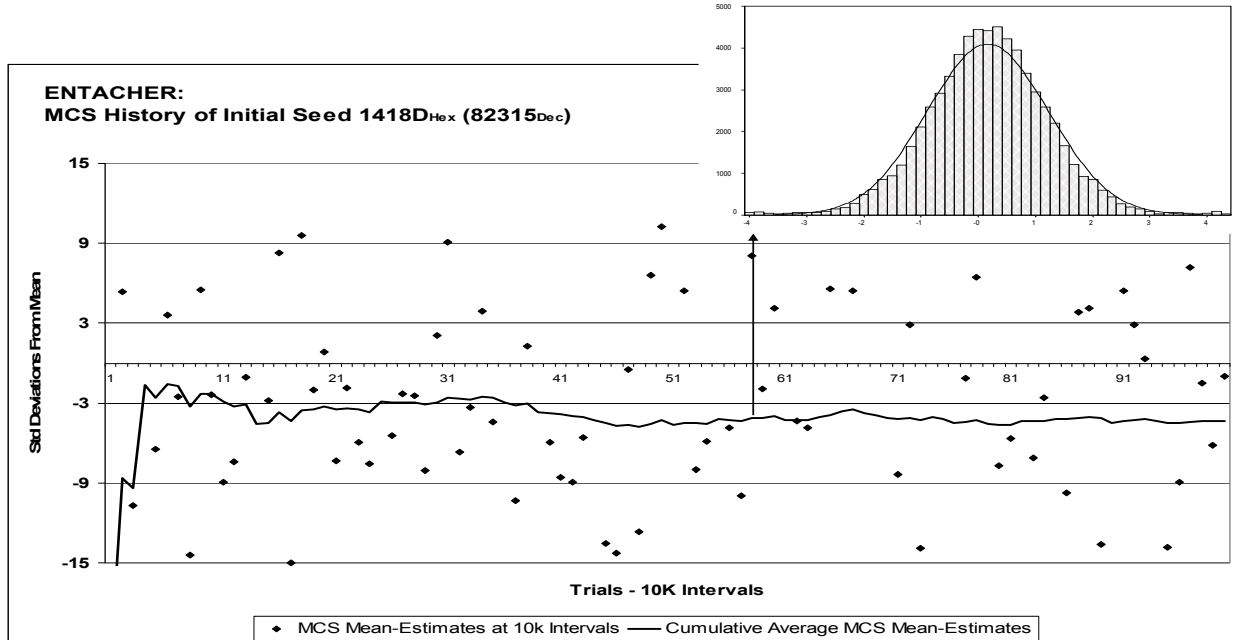
The ENTACHER MCS history graph (Figure 3.4) for initial seed, $S_i = 73489$, demonstrated an initial departure from zero, +15 standard deviations, at trials = 10,000; a recovery, +4 standard deviations, at trials = 200,000; and a final convergence on +4 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

Figure 3.1: ENTACHER Random Number Generator



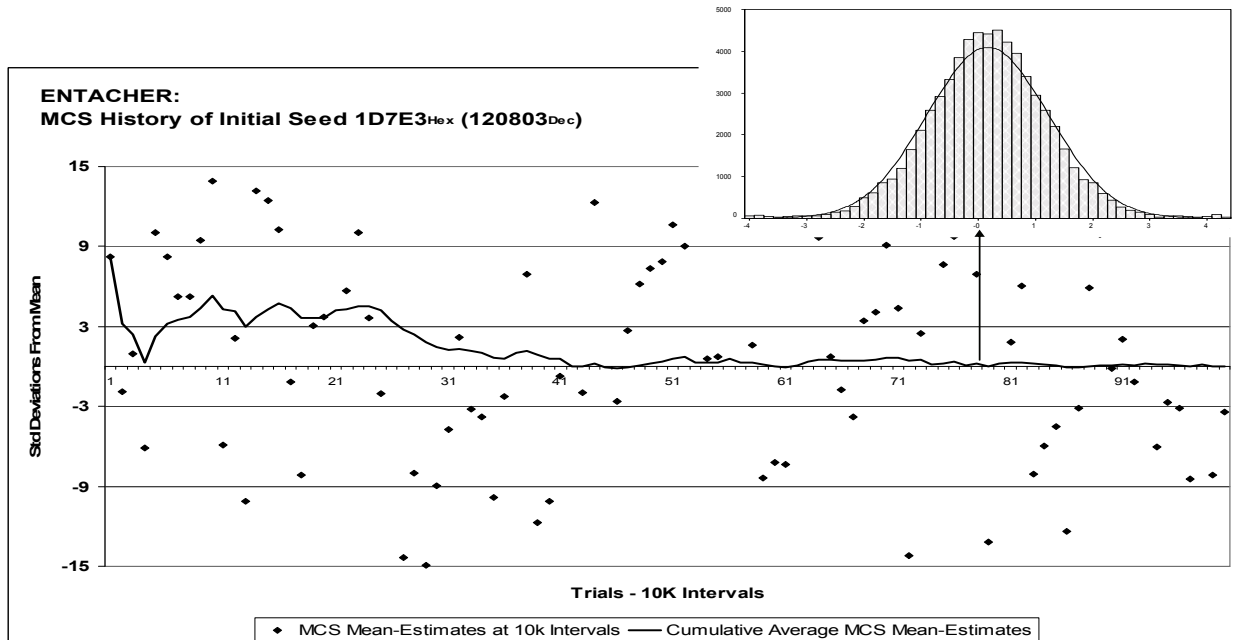
Each line in the graph follows the cumulative mean estimate of a Monte Carlo Simulation of the standard normal probability density function. Fifteen initial seeds (five each: high, middle and low mean estimates) are shown at 10k intervals, total trials = 1,000,000. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Down arrows correlate the 3σ , 0, and $+3\sigma$ points of the standard normal probability density function to the horizontal cumulative averages and their deviations from the mean.

Figure 3.2: ENTACHER Random Number Generator



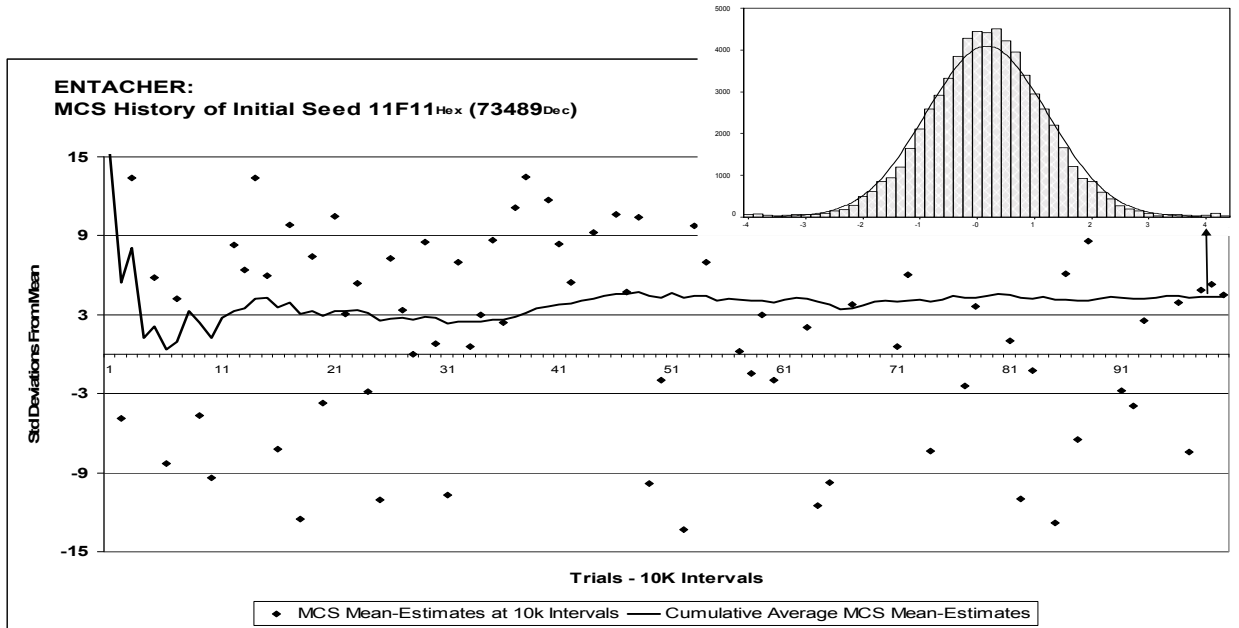
This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 82315, is the worst mean estimate below the mean. Each ♦ is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 82315, on the standard normal probability density function.

Figure 3.3: ENTACHER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 120803, is the best mean estimate. Each ♦ is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 120803, on the standard normal probability density function.

Figure 3.4: ENTACHER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 73489, is the worst mean estimate above the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 73489, on the standard normal probability density function.

LECUYER Pseudo-Random Number Generator

For LECUYER's distribution of mean estimates at Trial = 1,000,000, fifteen initial seeds were chosen, five from each extreme (the worst) and five from the center (the best). The mean estimates' initial seeds were used to capture the cumulative (running) average at 10,000-trial intervals of a Monte Carlo simulation of the Standard Normal probability distribution function. These data were normalized and graphed, (Figure 4.1), as graphical history of the Monte Carlo simulation's mean estimate at 10,000-trial intervals. LECUYER's cumulative mean-estimates undulated wildly (± 15 standard deviations) for the first 100,000 trials and then converged on their respective region of the mean estimate distribution.

Individual LECUYER MCS history graphs for the two worst initial seeds, 92765 and 102702, and closest to the True Mean, 105488, were isolated. The LECUYER MCS history graph (Figure 4.2) for initial seed, $S_i = 92765$,

demonstrated an immediate departure from zero, -15 standard deviations, at trials = 10,000; a recovery to -3 standard deviations at trials = 170,000; and a final convergence on -4 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The LECUYER MCS history graph (Figure 4.3) for initial seed, $S_i = 105488$, demonstrated an initial departure from zero, -1 standard deviations, at trials = 10,000; a spike to +6 standard deviations, at trials = 30,000; and a final convergence on 0.0 standard deviations. Individual 10,000-trial intervals, designated as \blacklozenge on the graph, demonstrated some wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

The LECUYER MCS history graph (Figure 4.4) for initial seed, $S_i = 102702$, demonstrated an initial departure from zero, +2 standard deviations, at trials = 10,000; a spike to

PSEUDO-RANDOM NUMBER INITIAL SEED BIAS IN MONTE CARLO SIMULATIONS

+11 standard deviations at trials = 20,000; a sharp drop to +2 standard deviations, at trials = 30,000; and a final convergence on +4 standard deviations. Individual 10,000-trial intervals, designated as ♦ on the graph, demonstrated wildly fluctuating mean estimates of ±15 standard deviations throughout the simulation run.

RANDU Pseudo-Random Number Generator

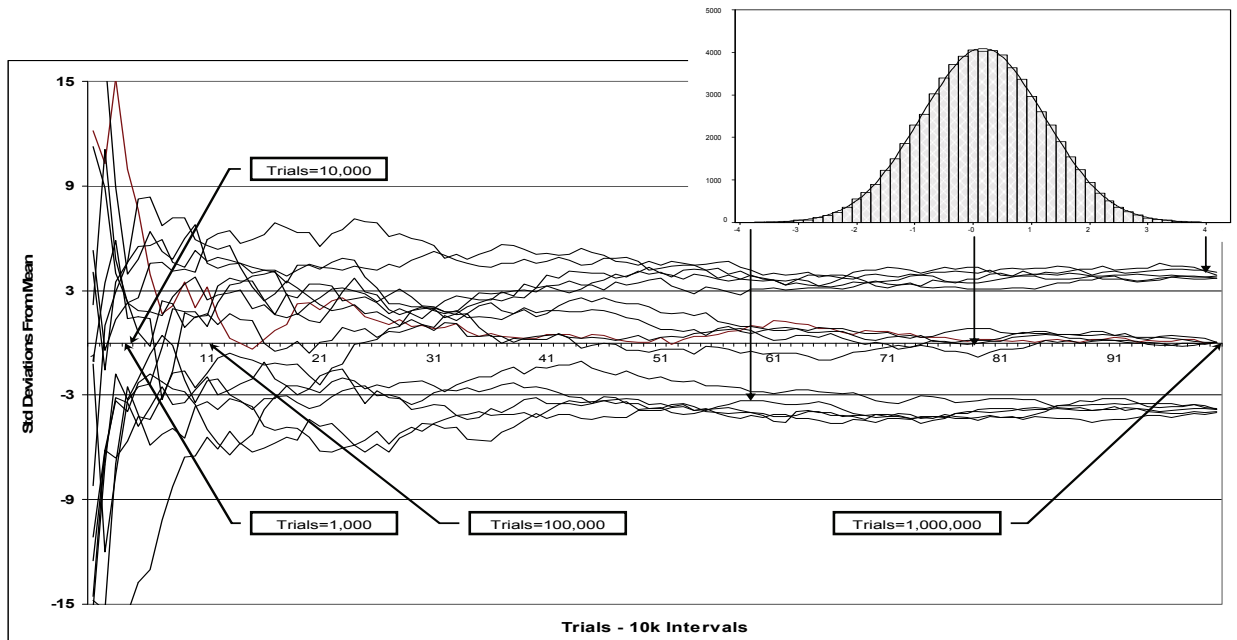
For RANDU's distribution of mean estimates at Trial = 1,000,000, fifteen initial seeds were chosen, five from each extreme, the worst, and five from the center, the best. The mean estimates' initial seeds were used to capture the cumulative (running) average at 10,000-trial intervals of a Monte Carlo simulation of the Standard Normal probability distribution function. These data were normalized and graphed, (Figure 5.1), as graphical history of the Monte Carlo simulation's mean estimate at 10,000-trial

intervals. RANDU's cumulative mean estimates undulated wildly (±15 standard deviations) for the first 100,000 trials and then converged on their respective region of the mean estimate distribution.

Individual RANDU MCS history graphs for the two worst initial seeds, 128395 and 93665, and closest to the True Mean, 81514, were isolated. The RANDU MCS history graph (Figure 5.2) for initial seed, $S_i = 128395$, demonstrated an immediate departure from zero, -15 standard deviations and a gradual convergence on -4 standard deviations. Individual 10,000-trial intervals, designated as ♦ on the graph, demonstrated wildly fluctuating mean estimates of ±15 standard deviations throughout the simulation run.

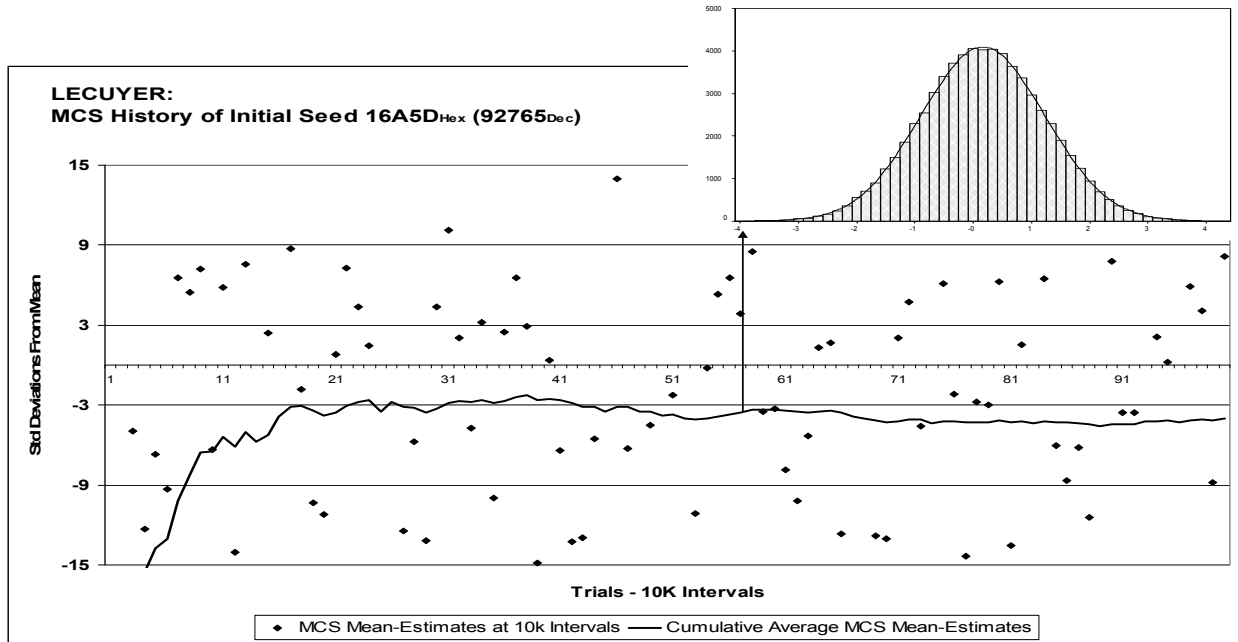
The RANDU MCS history graph (Figure 5.3) for initial seed, $S_i = 81514$, demonstrated an initial departure from zero, +12 standard deviations, at trials = 10,000; a recovery, 0.5 standard deviations, at trials =

Figure 4.1 LECUYER Random Number Generator



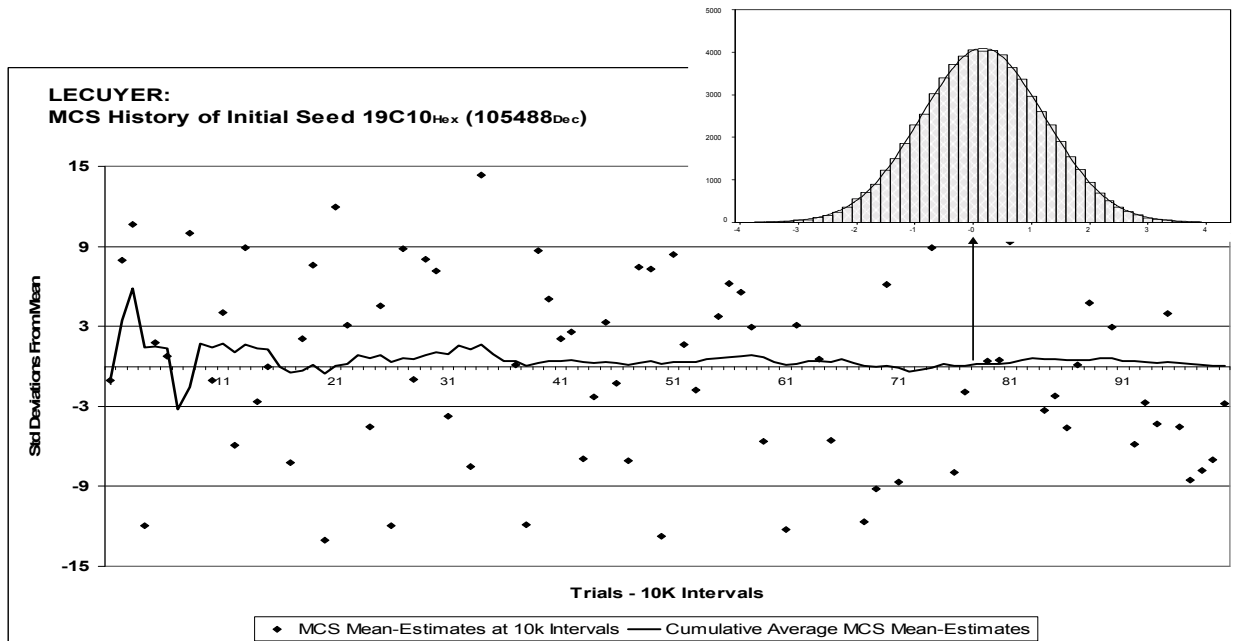
Each line in the graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Fifteen initial seeds (five each: high, middle and low mean estimates) are shown at 10k intervals, total trials = 1,000,000. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Down arrows correlate the 3σ , 0, and $+3\sigma$ points of the standard normal probability density function to the horizontal cumulative averages and their deviations from the mean.

Figure 4.2: LECUYER Random Number Generator



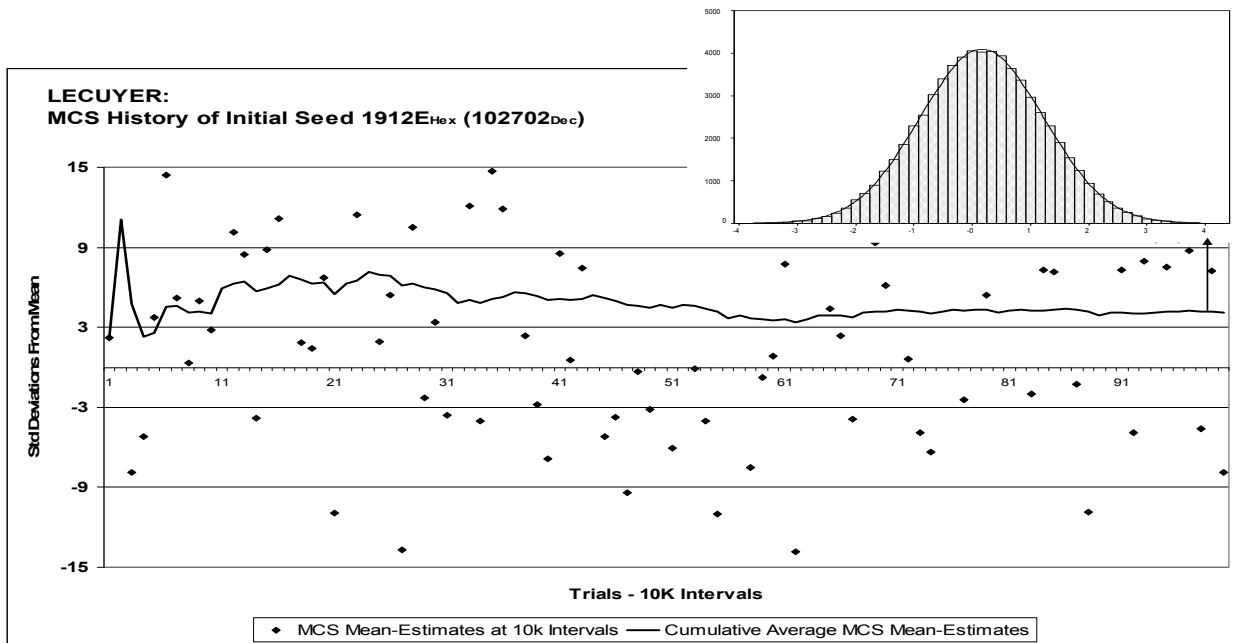
This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 92765, is the worst mean estimate below the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 92765, on the standard normal probability density function.

Figure 4.3: LECUYER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 105488, is the best mean estimate. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 105488, on the standard normal probability density function.

Figure 4.4: LECUYER Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 102702, is the worst mean estimate above the mean. Each \diamond is the average mean estimate of its 10k trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 102702, on the standard normal probability density function.

50,000; and a final convergence on 0.0 standard deviations. Individual 10,000-trial intervals, designated as \diamond on the graph, demonstrated some wildly fluctuating mean estimates of ± 15 standard deviations but many were ± 15 standard deviations throughout the simulation run.

The RANDU MCS history graph (Figure 5.4) for initial seed, $S_i = 93665$, demonstrated an initial negative departure from zero, -13 standard deviations, at trials = 10,000; a recovery, +9 standard deviations, at trials = 50,000; a sharp drop, +2 standard deviations, at trials = 100,000; and a final convergence on +3 standard deviations. Individual 10,000-trial intervals, designated as \diamond on the graph, demonstrated wildly fluctuating mean estimates of ± 15 standard deviations throughout the simulation run.

Conclusion

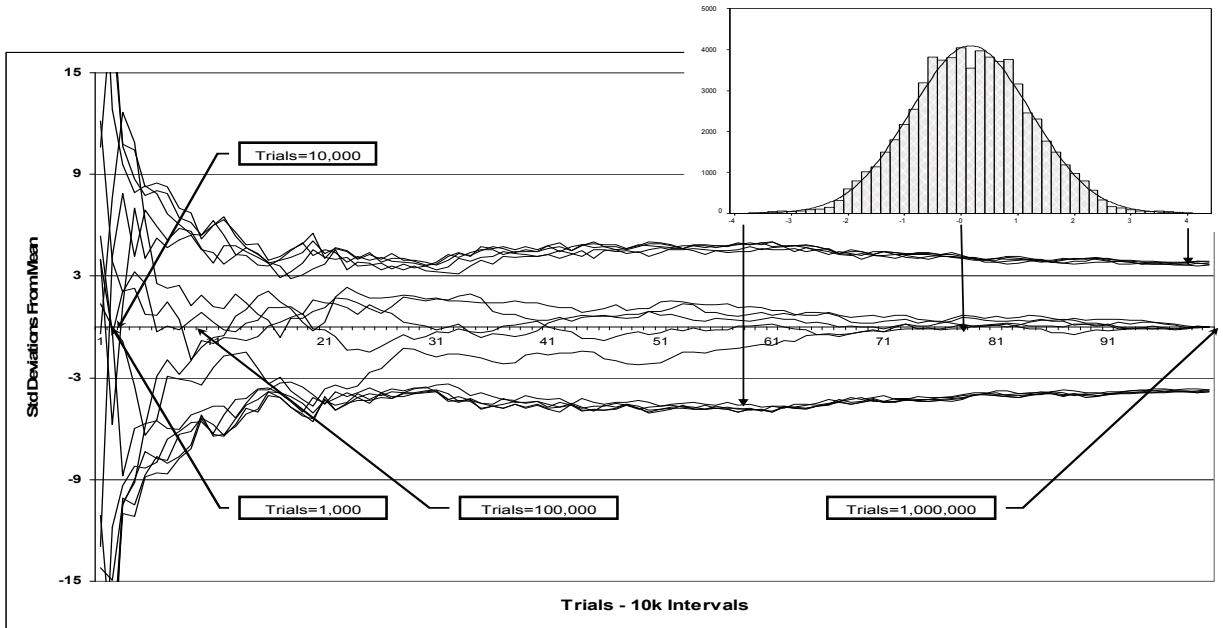
Monte Carlo simulations of the standard normal probability density function using all initial seed between 10000_{Hex} (65,536) through $1FFFF_{Hex}$

(131,071) did not always yield accurate estimates of the True Mean, $\mu = 0$. It was observed that the distribution of all mean estimates by Monte Carlo simulation is normal-like where about 67% of the mean estimates are within $\pm 1\sigma$ and about 95% of the mean estimates are within $\pm 2\sigma$. Conversely, about 5% of the mean estimates exceed 2σ , and the maximum mean estimates were $\pm 5\sigma$.

Most researchers have recognized that Monte Carlo simulations should be run using several initial seeds and this study supported that practice. Most researchers have also recognized that Monte Carlo simulations should be run for Trials = 1,000,000 or more. This study demonstrated that most mean estimates converged on their final value by trial = 500,000 and did not change very much with additional trials. Each random number generator exhibited similarly distributed mean estimates.

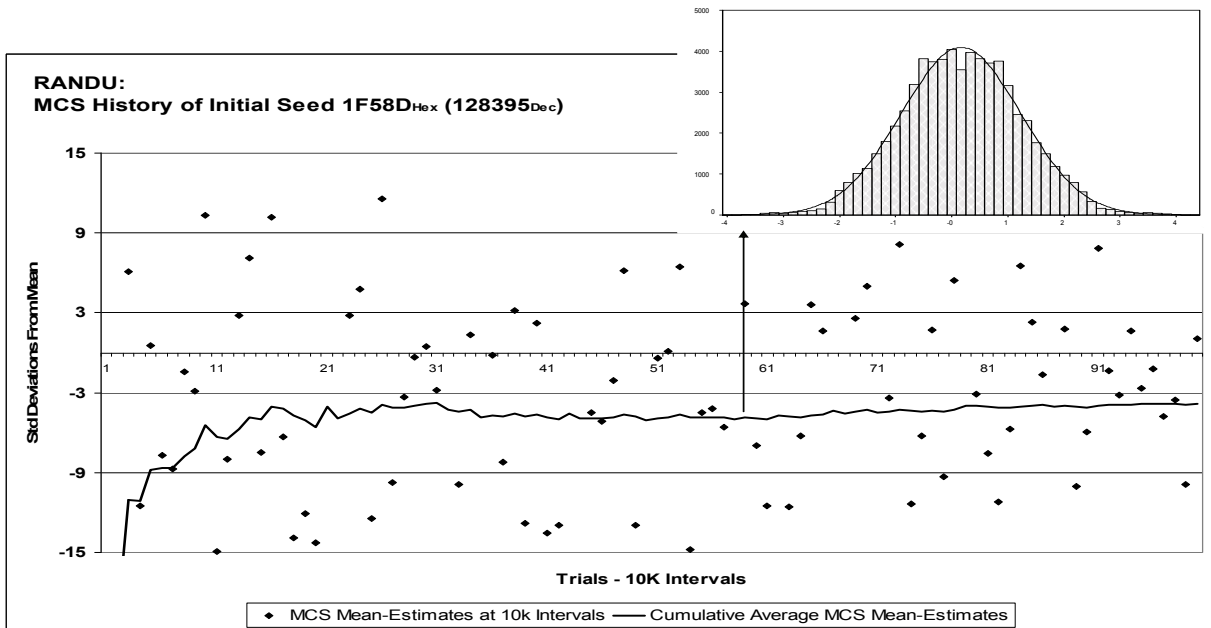
Good and bad initial seed numbers were not identified, but it was shown that errors of 2σ or more could occur in Monte Carlo simulations

Figure 5.1: RANDU Random Number Generator



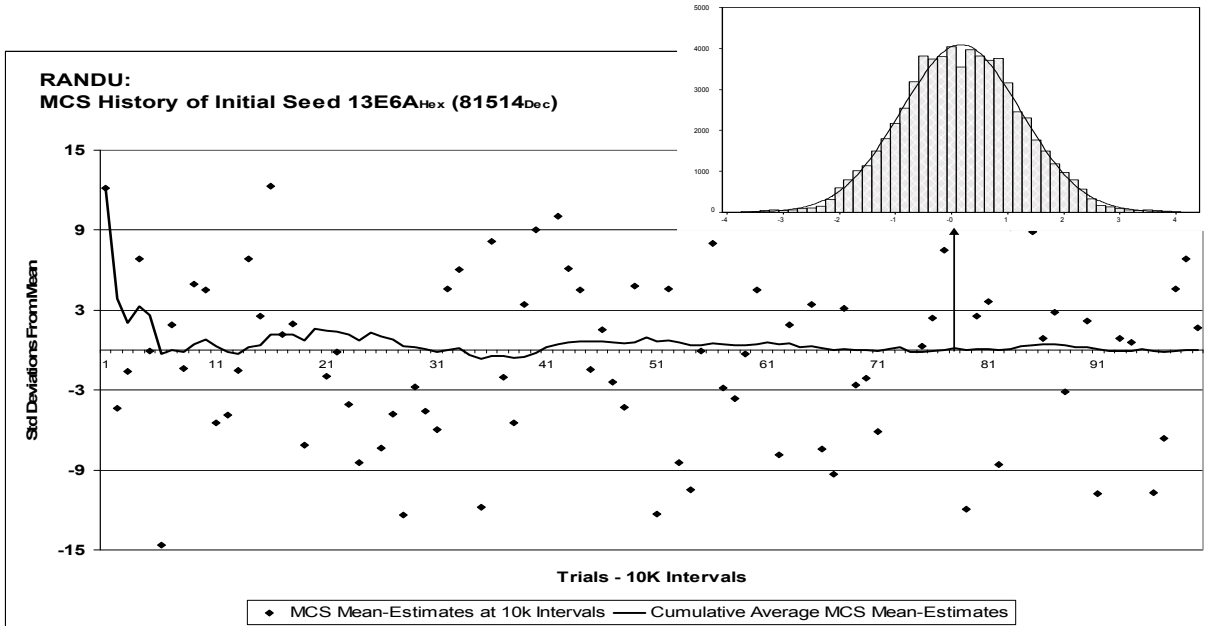
Each line in the graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Fifteen initial seeds (five each: high, middle and low mean estimates) are shown at 10k intervals, total trials = 1,000,000. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Down arrows correlate the 3σ , 0, and $+3\sigma$ points of the standard normal probability density function to the horizontal cumulative averages and their deviations from the mean.

Figure 5.2: RANDU Random Number Generator



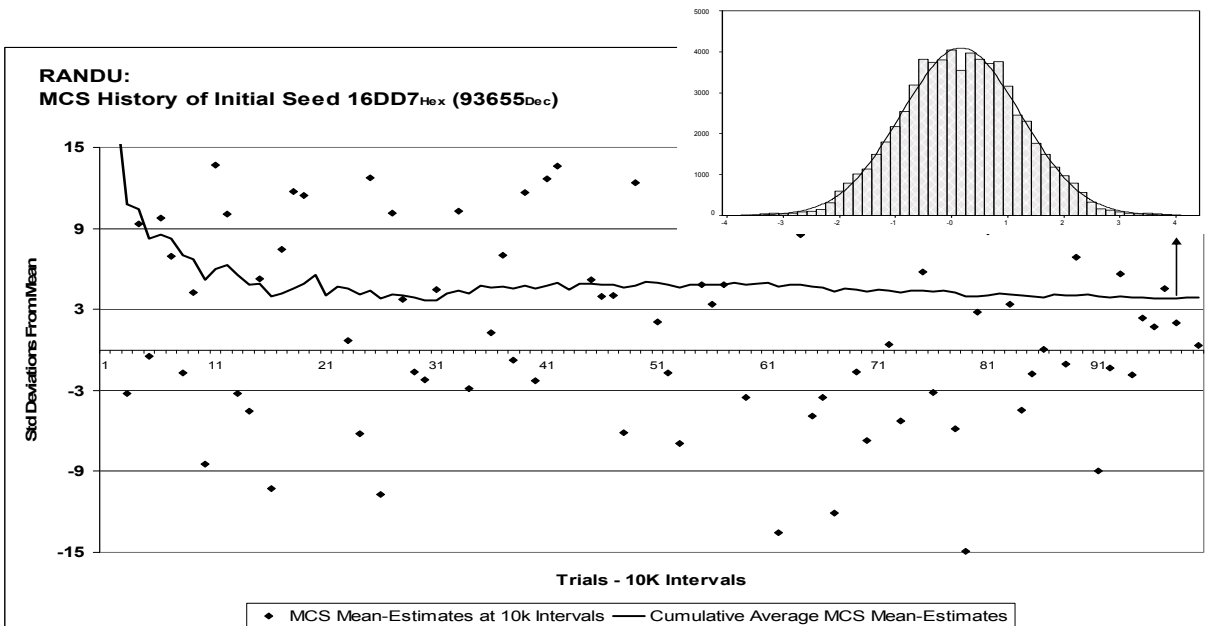
This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 128395, is the worst mean estimate below the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 128395, on the standard normal probability density function.

Figure 5.3 RANDU Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 81514, is the best mean estimate. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 81514, on the standard normal probability density function.

Figure 5.4: RANDU Random Number Generator



This graph follows the cumulative mean estimate of a Monte Carlo simulation of the standard normal probability density function. Initial seed, 93655, is the worst mean estimate above the mean. Each \blacklozenge is the average mean estimate of its 10k-trial interval. Inset graph is the distribution of all 65,536 initial seeds' mean estimates at 1,000,000 trials with the standard normal probability density function superimposed. Up arrow correlates the location of initial seed, 93655, on the standard normal probability density function.

based on initial seed selection. This study did not intend to identify a list of bad initial seeds that introduced significant error in Monte Carlo simulations of the standard normal probability density function, and then, assume they would also introduce significant error in other distributions or data sets; additional study is required to make this assumption.

Because the mean estimates randomly fluctuated throughout the simulation, any strategy that runs the simulation for T trials, discards these samples, and starts sampling at $Trials = T+1$, will repeat the ± 15 standard deviation fluctuations and gain nothing. In this study, every Monte Carlo simulation of the standard normal probability density function exhibited this phenomenon: randomly fluctuating mean estimates until the average of all the simulations stabilized. It is well known that averaging multiple samples will decrease the effects of random variations, thus, as samples accumulate, the average mean estimate stabilizes. However, discarding samples or resetting the average = 0 will also discard the accumulated diminishing effect of random variations and the mean estimates will wildly fluctuate, again.

The overall mean estimate of Monte Carlo simulations of the standard normal probability density function using all initial seed between 10000_{Hex} (65,536) through $1FFFF_{Hex}$ (131,071) was 0.0 at $Trial = 1,000$, 0.0 at $Trial = 10,000$, 0.0 at $Trial = 100,000$, and 0.0 at $Trial = 1,000,000$ for RANGEN, ENTACHER, LECUYER and RANDU pseudo-random number generators. The standard deviation estimates were 1.00 at all trials for each of these pseudo-random number generators.

Questions For Additional Research

Did these findings suggest relatively short simulation runs, trials = 1,000 or less, using tens of thousands initial seeds would yield more accurate mean estimates than typical Monte Carlo long simulation runs, trials = 500,000 or more, using a few initial seeds?

The total number of samples drawn from the population need not be increased. For example, typical Monte Carlo simulations requiring 100,000,000 samples (10 seeds, sample size 10, and 1,000,000 trials) cannot

guarantee an accurate mean-estimate. However, Monte Carlo simulations of 10,000 seeds, sample size 10, and 1,000 trials also require 100,000,000 samples and may yield estimates the mean with greater accuracy.

This study has shown that good pseudo-random number generators like RANGEN, ENTACHER, LECUYER and MOTHER yielded accurate mean estimates by Monte Carlo simulation of the standard normal probability density function when tens of thousands initial seeds were used with short simulation runs of trials = 1,000. It was also shown that a bad pseudo-random number generator, such as RANDU, could yield accurate mean estimates, which is contrary to its reported poor performance in Monte Carlo simulations.

This study raised many unanswered questions: Are these findings limited to Monte Carlo simulation of the standard normal probability density function, or will other data distributions exhibit similar findings? Will other pseudo-random number generators exhibit similar findings? Will Monte Carlo simulation runs of $10 \leq Trials \leq 1,000$ using tens of thousands initial seeds continue to yield accurate mean estimates of the standard normal probability density function?

References

- Blair, R. C. (1999). *BFRA ver. 2.1 users' manual*. College of Public Health, University of South Florida.
- Entacher, K. (1998). Bad subsequences of well-known linear congruential pseudorandom number generators. *ACM Transactions on Modeling and Computer Simulations*, 8(1), 61-70.
- Fahome, G., & Sawilowsky, S. S. (2001). *Rangen 2.0 for Fortran 90*. College of Education, Wayne State University.
- Fishman, G. S. (1995). *Monte Carlo: concepts, algorithms, and applications*. New York: Springer-Verlag.
- James, F. (1994). RANLUX: A Fortran implementation of the high-quality pseudo-random number generator of Lüscher. *Computer Physics Communications*, 79, 111-114.

L'Ecuyer, P. (1993). Two approaches for estimating the gradient in functional form. *Proceedings of the 1993 Winter Simulation Conference*.338-346.

L'Ecuyer, P., &Hellegalek, P. (1997). Random number generators: selection criteria and testing. *Random and Quasi-Random Point Sets, Lectures Notes In Statistics (138)*, Springer, 223-266.

Lehmer, D. H. (1951).Mathematical methods in large-scale computing units.^{2nd} *Symposium on Large-Scale Digital Calculating Machines, Harvard University Press, 26(141)*, 141-146.

Lüscher, M. (1994). A portable high-quality random number generator for lattice field theory calculations. *Computer Physics Communications, 79*, 100-110.

Marsaglia, G. (1994a). *The mother of all random generators*.Posted by Bob Wheeler to sci.stat.consult and sci.math.num-analysis on behalf of George Marsaglia on October 28, 1994.

Marsaglia, G. (1994b). Multiply-with-carry generators. Diehard CD-ROM, available online at ftp.csis.hku.hk.

Miller, A. (1995). MODULE luxury. <http://jblevins.org/mirror/amiller/luxury.f90>

Park, S. K., & Miller, K. W. (1988). Random number generators: Goods ones are hard to find. *Communications of the ACM, 31(10)*, 1192-1201.

Sawilowsky, S. S., & Blair, R. C. (1987). *RANGENPC 1.0 for Fortran 77*. College of Education, Wayne State University.