***Jñānābha*, Vol. 50(2) (2020), 273-282**
(Dedicated to Honor Dr. R. C. Singh Chandel on His 75th Birth Anniversary Celebrations)

A STUDY ON THE EFFICIENCY OF PSEUDO RANDOM NUMBERS BY ANALYZING THE ERROR PROPAGATION WITH REFERENCE TO THE MONTE CARLO METHOD FOR NUMERICAL INTEGRATION

By
Saurabh Saxena[1], Ashish Arora[2] and Shweta Choudhary[3]
Department of Applied Science and Humanities, ABES Engineering College
Campus -1, 19th KM Stone, NH-24, Ghaziabad-201009, Uttar Pradesh, India
Email:saxena22saurabh@gmail.com (Corresponding Author), ashish.arora@abes.ac.in, shweta.chaudhary@abes.ac.in
(*Received : July* 11, 2020 *; Revised: December* 29, 2020)

Abstract

Monte Carlo method is a powerful method for computing the value of complex integrals using probabilistic techniques and estimates the integrals or other quantities that can be expressed as an expectation by averaging the results of a high number of statistical trials.Its convergence rate$O(\sqrt{N})$, is independent of dimension and hence it is preferred for a wide range of high dimensional problems. In this article efficiency of random numbers is being analysed by comparing the error in the evaluation of bi-variate integral corresponding to different size of random numbers, and equi-spaced points by Monte Carlo method. Also analysed and discussed the propagation of error in every case.
2010 Mathematics Subject Classifications: 65C05, 65D30.
Keywords and phrases: Monte Carlo Method, Numerical Integration, Random Numbers, efficiency of random numbers.

1 Introduction

1.1 Need for study

The Monte Carlo method for numerical integration is believed to rely absolutely on the pseudo random numbers in the context of their randomness. So far great analysis has been done to establish the fact that by assuring the randomness of pseudo random numbers we may have more accurate results i.e. the computational error may be minimized but does this error also depends upon the size of random numbers. Such analysis takes an attention towards the size of random numbers and will always play a key role to get the value of the integral more accurately.

1.2 Pseudo random numbers

Randomness generated by any deterministic pattern with the help of the system is called Pseudo- randomness and the numbers attaining such type of randomness are known as pseudo random numbers. A process that appears to be random but is not is said to be pseudorandom process. Statistical randomness is a typical exhibition of pseudorandom sequences while it is generated by an entirely deterministic process. Most computer programming languages include functions or library routines that purport to be random number generators. They are often designed to provide a random byte or word, or a floating point number uniformly distributed between 0 and 1. Such library functions often have poor statistical properties and some will repeat patterns after only tens of thousands of trials. They all fall in the category of pseudo random numbers.

1.3 Monte Carlo method for numerical integration

The Monte Carlo method is a method for solving problems using random variables. This method was first introduced by Stanislaw Ulam for simulations in physics and other fields that require solutions for problems that are impractical or impossible to solve by traditional analytical or numerical methods. Monte Carlo method has become very popular in recent years, especially in those cases where the number of factors included in the problem are large in numbers and an analytical solution is impossible (for example numerical integration for higher orders).

The central idea behind the Monte Carlo method is either to construct a stochastic model which is in agreement with the actual problem analytically or to simulate the whole problem directly. In both the cases the element of randomness has to be introduced according to well defined rules. After that a large number of trials are performed and the results are observed and finally a statistical analysis is undertaken in the usual way. The advantages of the method are, above everything is that even very difficult problems can often be treated quite easily and desired modifications can be applied without too much trouble.

For a bi-variate integral like

(1.1) $I = \int_c^d \int_a^b f(x, y)dxdy.$

The Monte Carlo method consists of the following steps.

1.3.1 (Using Random Numbers)

***Step* 1.** Pick up n randomly generated points $(x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots, (x_n, y_n)$ in the rectangle $[a, b] \times [c, d]$.

***Step* 2.** Determine the average value of the function which is given by

(1.2) $\hat{f} = \dfrac{1}{n} \sum_{i=1}^{n} f(x_i, y_i).$

***Step* 3.** Compute the approximation to the integral

(1.3) $\int_c^d \int_a^b f(x, y)dxdy \approx (b - a) \times (d - c)\hat{f}.$

The error for this estimation is given by

(1.4) Error $= (b - a) \times (d - c)\sqrt{\dfrac{f^2 - (\hat{f})^2}{n}},$

where

(1.5) $\hat{f^2} = \dfrac{1}{n} \sum_{i=1}^{n} f^2(x_i, y_i).$

1.3.2 (Using Equispaced Points)

***Step* 1.** Divide the x-range and y-range in n equal parts to get
$x_i = a + ih$ where $i = 1, 2, 3, \ldots, n$
$y_j = c + jk$ where $j = 1, 2, 3, \ldots, n$
where $h = \frac{1}{n}(b - a)$ and $k = \frac{1}{n}(d - c)$.
Our equispaced nodes are $(x_i, y_i)\ \forall i, j$ in the rectangle $[a, b] \times [c, d]$.

***Step* 2.** Determine the average value of the function which is given by

(1.6) $\hat{f} = \dfrac{1}{n \times n} \sum_{i=1}^{n} \sum_{j=1}^{n} f(x_i, y_j).$

***Step* 3.** Compute the approximation to the integral

(1.7) $\int_c^d \int_a^b f(x, y)dxdy \approx (b - a) \times (d - c)\hat{f}.$

The error for this estimation is given by

(1.8) Error $= (b - a) \times (d - c)\sqrt{\dfrac{f^2 - (\hat{f})^2}{n \times n}},$

where

(1.9) $\hat{f^2} = \dfrac{1}{n \times n} \sum_{i=1,j=1}^{n} f^2(x_i, y_j).$

2 Objective

The focal point of this investigation is primarily to analyze the error obtained in the evaluation of a bi-variate integral by Monte Carlo method using different size of random numbers and equi-spaced points. The basic objective of this investigation is to explore and analyze the error propagation and the efficiency of random numbers for Monte Carlo integration with reference to the error produced as the size of numbers is increased.

3 Review of literature

The detailed study regarding the beginning of Monte Carlo method and its origination is discussed in a scholarly article by Metropolis[11] then after Eckhardt[4] in his scholarly article mentioned the problem, as a solution of which this technique was discovered. In this article the letters showing the personal conversation of Stan Ulam and John Von Neumann regarding Monte Carlo method were also published. A comprehensive review of literature concerning Monte Carlo method may be found in the book by Kalos et al. [10]. A chapter in encyclopedia of Biostatistics by Smyth[14] throws light on all the techniques available to solve one or two dimensional integration including Monte Carlo technique. In 2001 Gould, Tobochnik and Christian[8] in their book studied all the techniques of numerical integration and mentioned their error analysis. In 2002 Yang[15] discussed all other methods and techniques of numerical integration and stated the benefits of Monte Carlo integration over the other methods available in an extended essay in mathematics during his IB diploma program. In parallel to the same we can also find the practical limitations of numerical integration discussed by Evans[6].

Since Monte Carlo Integration was based on random numbers therefore equal attention of scientists were attracted towards the random numbers. In early times a great work has accomplished regarding random numbers, randomness and their use in which a great role was played by Chaitin[3] to discuss the same. A detailed study of random numbers and various techniques (Random Number Generators) to produce random numbers may be found in the book by Gentle[7]. The types of random numbers and randomness are clearly defined and illustrated in an article by Eddelbuettel[5]. In a research paper by Hellekalek[9] he mentioned that what should be the properties of good random number generators.

Early studies on Monte Carlo integration were mainly concerned with the problem of improving the randomness of numbers used. The non-parametric tests to check the randomness of numbers may be referred from Bhar[1]. Just to avoid the inherent errors of random numbers Park and Miller[12] suggested to follow the minimal standards for random number generators. To get rid of this situation of ambiguity that whether the numbers in use are true random or not, concept of quasi random numbers was coined and discussed by Caflisch[2]. The question on the reliability of random numbers with respect to one dimensional Monte Carlo Integration was raised by Saxena and Saxena[13].

4 Methodology

For the proposed objective first random numbers were collected and tested for their randomness in the sense of their independence and uniformity. Two bi-variate integrals are evaluated by Monte Carlo Method using the different size of these random numbers and equi-spaced points. Error in every case was recorded and analyzed to state the findings and conclusion.

The random number's data files of different size numbers of 1000, 2000, 3000, 4000, 5000through two different sources, online and computer generated, are collected and saved with following nomenclature

***Table* 4.1:** Online generated random numbers

File Name	Online Source	Size
olrr1.dat	Research Randomizer	1000
olrr1.dat	Research Randomizer	2000
olrr1.dat	Research Randomizer	3000
olrr1.dat	Research Randomizer	4000
olrr1.dat	Research Randomizer	5000
olrorg1.dat	Random.Org	1000
olrorg2.dat	Random.Org	2000
olrorg3.dat	Random.Org	3000
olrorg4.dat	Random.Org	4000
olrorg5.dat	Random.Org	5000
olgp1.dat	Graph Pad	1000
olgp2.dat	Graph Pad	2000
olgp3.dat	Graph Pad	3000
olgp4.dat	Graph Pad	4000
olgp5.dat	Graph Pad	5000

***Table* 4.2:** Computer generated random numbers

File Name	Source	Size
Int_1_1	Computer generated(RNG)	1000
Int_1_2	Computer generated (RNG)	2000
Int_1_3	Computer generated (RNG)	3000
Int_1_4	Computer generated (RNG)	4000
Int_1_5	Computer generated (RNG)	5000
Int_2_1	Computer generated (RNG)	1000
Int_2_2	Computer generated (RNG)	2000
Int_2_3	Computer generated (RNG)	3000
Int_2_4	Computer generated (RNG)	4000
Int_2_5	Computer generated (RNG)	5000
Int_3_1	Computer generated (RNG)	1000
Int_3_2	Computer generated (RNG)	2000
Int_3_3	Computer generated (RNG)	3000
Int_3_4	Computer generated (RNG)	4000
Int_3_5	Computer generated (RNG)	5000

Then to assure the randomness of these numbers with reference to their independence and uniformity these numbers have been tested by four of the important statistical tests namely Poker Test, Run Test, Frequency Test and Frequency Monobit Test. The detailed study of the tests applied may be obtained from the web address*:
https://drive.google.com/file/d/1ja22gYjVOxliXDI3vmaqiGFsyeC6wJ4A/view?usp=sharing

4.1 Error Evaluation

Although many bi-variate integrals were evaluated but in the present work, only two bi-variate integral is considered to justify the efficiency of random numbers and to discuss the error analysis, generated by Monte Carlo method using random points (computer generated & online generated) as well as equispaced points.

4.1.1 First Integral

The first integral under investigation is following of which the exact value is 1.71828

$$I_1 = \int_0^1 \int_0^{\frac{\pi}{2}} (e^y \cos x) dxdy.$$

4.1.1.1 (Using Random Nodes)

Out of the six data files for 1000 data size there can be 15 different combinations and 15 more combinations when data files for x-series and y-series are interchanged. These 30 pairs of codes for 30 file combinations are

(1,2), (1,3), (1,4), (1,5), (1,6), (2,3), (2,4), (2,5), (2,6), (3,4), (3,5), (3,6) (4,5), (4,6), (5,6), (2,1), (3,1), (4,1), (5,1), (6,1), (3,2), (4,2), (5,2), (6,2), (4,3), (5,3), (6,3), (5,4), (6,4), (6,5)

The same number of combinations may be obtained for 2000, 3000,4000 and 5000 data size.

We shall now use the following program to get the error in the evaluation of the above integral corresponding to all the 30 combinations in each case of different data size of 1000, 2000, 3000, 4000 and 5000.

```
10   CLS:KEY OFF:DIM F$(30)
20   LOCATE 10,5: INPUT "Give drive letter of data files";D$:CLS
30   FOR I = 1 TO 30 :READ F$(I) :NEXT I
40   DATA "INT_1_1.DAT", "INT_2_1.DAT","INT_3_1.DAT","olrr1.DAT", "olrorg1.DAT","olgp1.DAT"
50   DATA "INT_1_2.DAT", "INT_2_2.DAT","INT_3_2.DAT","olrr2.DAT", "olrorg2.DAT", "olgp2.DAT"
60   DATA "INT_1_3.DAT", "INT_2_3.DAT","INT_3_3.DAT","olrr3.DAT", "olrorg3.DAT", "olgp3.DAT"
70   DATA "INT_1_4.DAT", "INT_2_4.DAT","INT_3_4.DAT","olrr4.DAT", "olrorg4.DAT","olgp4.DAT"
80   DATA "INT_1_5.DAT", "INT_2_5.DAT","INT_3_5.DAT", "olrr5.DAT","olrorg5.DAT","olgp5.DAT"
90   FOR I = 1 TO 30: F$(I)= D$ +":"+ F$(I): NEXT I
100  LOCATE 10,2:PRINT "Choose your DATA FILE  size": LOCATE 16,1
110  FOR I=1 TO 5: PRINT I;".  "; I*1000,:NEXT I
120  LOCATE 18,2: INPUT "Select choice Number";C :CLS
130  IF C = 1 THEN LOI = 1:IF C =1 THEN UPI = 6
140  IF C= 2 THEN LOI =7:IF C = 2 THEN UPI =12
150  IF C= 3 THEN LOI =13:IF C=3 THEN UPI =18
160  IF C= 4 THEN LOI =19:IF C=4 THEN UPI =24
170  IF C= 5 THEN LOI =25:IF C=5 THEN UPI =30
180  DS = C*1000
190  LOCATE 10,5: PRINT "Select the Data file":LOCATE 12,1
200  FOR I= LOI TO UPI : PRINT K+1;". "+F$(I),:K=K+1:NEXT I
210  LOCATE 15,1: INPUT "For x-range type your File Number";C1
220  IF C1 =< 0 OR C1 > 6 THEN 230 ELSE 240
230  LOCATE 15,1:PRINT"   " : GOTO 210
240  LOCATE 17,1: INPUT "For y-range type your File Number";C2
250  IF C2 =< 0 OR C2 > 6 THEN 260 ELSE 270
260  LOCATE 17,1:PRINT "   " : GOTO 240
270  IF C1 = C2 THEN 280 ELSE 330
280  LOCATE 18,1: PRINT"choose distinct files"
290  ANS$=INKEY$:IF ANS$="" THEN 290 ELSE 300
300  LOCATE 18,1:PRINT "   " : GOTO 210
310  IF C=1 THEN C1=C1 :IF C=1 THEN C2 = C2
320  IF C=2 THEN C1=C1+6 :IF C=2 THEN C2=C2+6
330  IF C=3 THEN C1=C1+12 :IF C=3 THEN C2=C2+12
340  IF C=4 THEN C1=C1+18 : IF C=4 THEN C2=C2+18
350  IF C=5 THEN C1=C1+24 : IF C=5 THEN C2=C2+24
360  F1$=F$(C1):F2$=F$(C2):CLS
370  DEF FNI(A,B)=  EXP(B)*(COS(A))
380  XLO = 0: XUP = 1.57: YLO = 0: YUP = 1:CLS
390  H=(XUP-XLO)/DS : K=(YUP-YLO)/DS
400  OPEN F1$ FOR INPUT AS #1
410  OPEN F2$ FOR INPUT AS #2
420  INPUT #1,X : X=X*(XUP-XLO)
430  INPUT #2,Y : Y=Y*(YUP-YLO)
440  Y = YLO +Y:X = XLO+X:SUM = SUM + FNI(X,Y)
450  IF NOT EOF(1) THEN 470
460  SUM=SUM/DS :SUM=SUM*(XUP-XLO)*(YUP-YLO)
470  LOCATE 8,10:PRINT "By Monte-Carlo Integration :-"
480  LOCATE 10,10 :  PRINT "Using  Random data files: ";F1$; " and ";F2$
490  LOCATE 12,10:PRINT"Value of integral ";SUM
500   LOSE:END
```

By repeated execution of the program to evaluate the integral for the above 30 combinations of files for *x* and *y* series for different data size, each time we get the different values of the integral given in the following ***Table*** **4.3**

Table **4.3:** Error in the evaluation of the first integral corresponding to different data size of random numbers

S.No.	File Combination	No of Random Points				
		1000	**2000**	**3000**	**4000**	**5000**
1	(1,2)	0.067408	0.048285	0.017989	0.003063	0.00491
2	(1,3)	0.041154	-0.062	-0.00728	0.00838	-0.00991
3	(1,4)	0.032333	0.034495	-0.00917	-0.00541	-0.0055
4	(1,5)	0.039267	0.021702	0.01348	-0.00322	-0.00505
5	(1,6)	0.057623	0.034826	0.002838	0.01335	-0.00996
6	(2,1)	-0.05087	-0.05682	-0.02459	0.000674	0.00351
7	(2,3)	-0.05254	-0.03479	-0.02618	0.001821	-0.0156
8	(2,4)	-0.04025	-0.03009	-0.02576	-0.00819	-0.00431
9	(2,5)	-0.01698	-0.04166	-0.00776	0.001848	-0.007
10	(2,6)	-0.01698	-0.03773	-0.01538	0.01783	-0.01571
11	(3,1)	-0.00833	-0.02658	0.012672	0.002917	0.020358
12	(3,2)	0.018618	0.005832	0.033779	-0.00075	0.015812
13	(3,4)	-0.01177	0.003089	0.011078	-0.00438	0.012958
14	(3,5)	0.00941	-0.01308	0.019395	-0.00127	0.025447
15	(3,6)	0.037444	-0.00405	0.023443	0.004379	0.011069
16	(4,1)	-0.0157	-0.04233	-0.00815	0.014344	0.00533
17	(4,2)	0.032217	-0.00112	0.017967	0.013179	0.00734
18	(4,3)	-0.01004	-0.01062	-0.00607	0.020466	-0.00116
19	(4,5)	0.012551	-0.02935	0.015483	0.017253	0.004658
20	(4,6)	0.030893	-0.01538	0.015429	0.019	-0.00493
21	(5,1)	-0.04	-0.01764	-0.00935	-0.00618	-0.00655
22	(5,2)	0.004684	0.022436	0.010774	0.000235	-0.00731
23	(5,3)	-0.0211	0.010683	-0.02219	0.00123	-0.0064
24	(5,4)	-0.0203	0.008032	-0.00927	-0.00472	-0.00734
25	(5,6)	0.000663	0.011038	0.001451	0.020623	-0.01201
26	(6,1)	-0.04784	-0.0254	-0.01884	-0.01361	0.011057
27	(6,2)	-0.00151	0.006768	0.005076	-0.00695	0.007871
28	(6,3)	-0.02007	-0.00191	-0.01608	-0.01724	0.002574
29	(6,4)	-0.02801	0.001943	-0.00855	-0.0275	0.006288
30	(6,5)	-0.02269	-0.01021	0.003333	-0.00292	0.009838

The following ***Figure*** **4.1** displays the values of our first integral corresponding to the same combination using different size of random numbers.

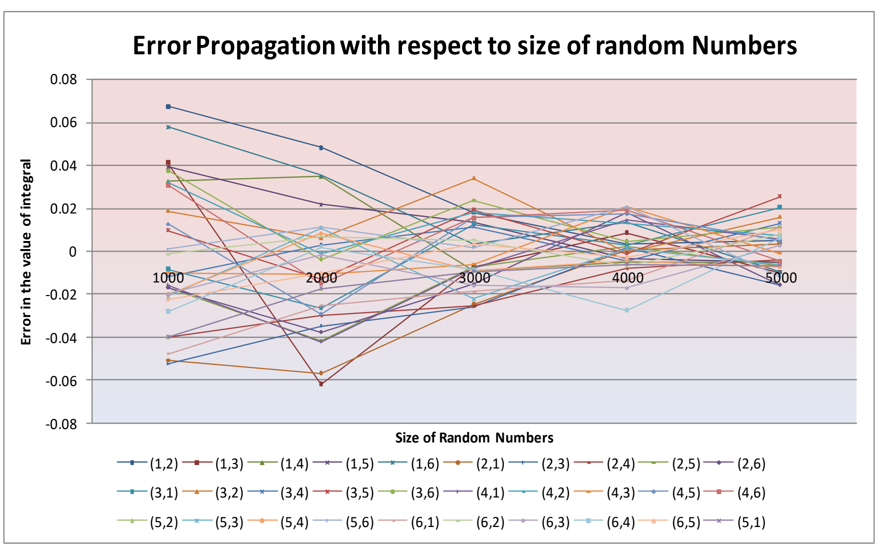

Figure **4.1:** Value of first Integral corresponding to different combination of random numbers

4.1.1.2 (Using Equispaced Nodes)

In order to evaluate the integral using equispaced nodes, we now present a tiny program

```
10   REM "First Integral (2-D)-Equi-spaced Nodes"
20   DEF FNI(A,B)=  EXP(B)*(COS(A))
30   CLS:XLO = 0:XUP = 1.57:YLO = 0:YUP = 1
40   LOCATE 10,10: INPUT "No of Equi-spaced
     Nodes ";N
50   DIM A(N): DIM B(N)
60   A(0)= XLO :B(0)= YLO
70   H=(XUP - XLO)/N:K =(YUP - YLO)/N
80   FOR I= 1 TO N
90   A(I)= A(I-1)+H:B(I)= B(I-1)+K
100  NEXT I
110  FOR I = 1 TO N
120  FOR J = 1 TO N
130  ESUM = ESUM + FNI(A(I),B(J))
140  NEXT J
150  NEXT I
160  ESUM = ESUM*H*K
170  LOCATE 15,10: PRINT "Divisions = "; N,
     "Value = ";ESUM
180  END
```

Here we are making equal numbers of divisions in x & y range. Corresponding to different numbers of division from 20 up to 400 with step size of 20 we get the following observations.

***Table* 4.4:** Error in the evaluation of the first integral corresponding to equi-spaced points

S. No.	No. of Equi-spaced Nodes	Error
1	20	-0.02668
2	40	-0.01277
3	60	-0.00839
4	80	-0.00625
5	100	-0.00497
6	120	-0.00414
7	140	-0.00353
8	160	-0.00308
9	180	-0.00274
10	200	-0.00247
11	220	-0.00225
12	240	-0.00206
13	260	-0.00189
14	280	-0.00176
15	300	-0.00164
16	320	-0.00153
17	340	-0.00145
18	360	-0.00137
19	380	-0.00129
20	400	-0.00124

The following ***Fig* 4.2** displays the values of first integral using different no. of equi-spaced points

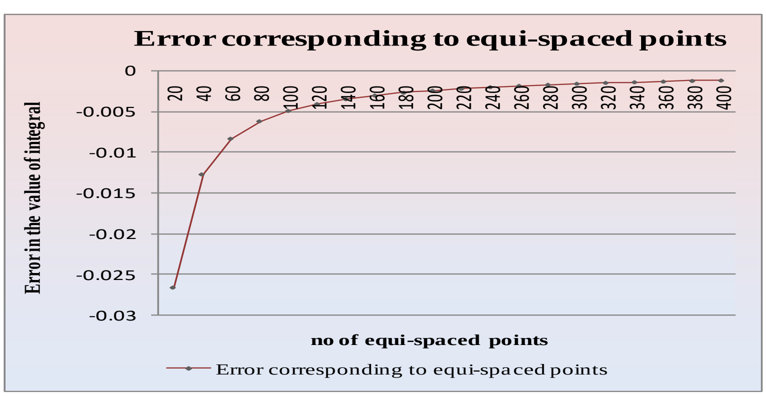

***Figure* 4.2:** Value of first Integral using Equi-spaced points

4.1.2 Second Integral

The second integral under investigation is following, of which the exact value is **30.75**

$$I_2 = \int_0^3 \int_1^2 xy(1 + x + y)dydx$$

4.1.2.1 (Using Random Nodes)

For the evaluation of our second integral with random nodes, we shall make use of the same program as in case of first integral with a change in line 390 which is corresponding to the integrand and limit of our integral. The modified form of this line should be

370 *DEF FNI(A,B)= A*B **(1 + *A* + *B*).

By repeated execution of the program to evaluate the integral for the above 30 combinations of files for x and y series for different data size, each time we get the different values of the integral given in the following ***Table* 4.5**

***Table* 4.5:** Error in the evaluation of the first integral corresponding to different data size of random numbers

S.No.	File Combination	No of Random Points				
		1000	2000	3000	4000	5000
1	(1,2)	0.68739	0.8731	0.50389	-0.21843	0.17533
2	(1,3)	-0.38242	0.17063	-0.48562	-0.26221	-0.23931
3	(1,4)	-0.22649	0.51525	0.06432	-0.55092	0.08877
4	(1,5)	0.33795	-0.10327	0.1723	-0.04857	0.37194
5	(1,6)	0.71704	0.12801	0.35889	0.14425	-0.05125
6	(2,1)	-0.57884	-0.25659	0.05084	-0.24966	0.16622
7	(2,3)	0.56632	0.89652	-0.35583	-0.14421	-0.14014
8	(2,4)	0.22057	1.05289	0.08923	-0.50922	0.05638
9	(2,5)	0.85234	0.44431	0.34344	-0.17728	0.40211
10	(2,6)	1.21182	0.877	0.41709	0.0478	0.01805
11	(3,1)	-0.93095	-0.52355	-0.27571	-0.32399	0.08473
12	(3,2)	1.33098	1.31994	0.27125	-0.17266	0.19092
13	(3,4)	0.20432	0.73865	-0.28985	-0.6144	-0.08092
14	(3,5)	0.50104	0.18175	0.16808	-0.13837	-0.03861
15	(3,6)	0.63055	0.51737	-0.02228	0.30537	-0.28115
16	(4,1)	-0.74456	-0.31848	0.07094	-0.33188	0.20884
17	(4,2)	0.98992	1.34504	0.56246	-0.27532	0.17807
18	(4,3)	0.23626	0.58867	-0.45599	-0.33575	-0.29005
19	(4,5)	0.42065	0.33956	0.1582	-0.33416	0.21872
20	(4,6)	0.80417	0.58671	0.12106	0.20341	-0.1379
21	(5,1)	-0.51242	-0.51602	-0.06132	-0.07299	0.35125
22	(5,2)	1.2658	1.1318	0.54243	-0.18941	0.39019
23	(5,3)	0.1884	0.44252	-0.26213	-0.10559	-0.38221
24	(5,4)	0.07553	0.75337	-0.0983	-0.56874	0.09071
25	(5,6)	1.08072	0.34093	0.24161	-0.01607	-0.06091
26	(6,1)	-0.41571	-0.52064	0.12649	-0.1454	0.17736
27	(6,2)	1.392	1.35566	0.63354	-0.21886	0.27853
28	(6,3)	0.02216	0.53968	-0.43079	0.0716	-0.3643
29	(6,4)	0.1783	0.78782	-0.13822	-0.30138	-0.00375
30	(6,5)	0.85284	0.10392	0.25903	-0.26995	0.17493

The following ***Fig* 4.3** displays the values of our first integral corresponding to the same combination using different size of random numbers.

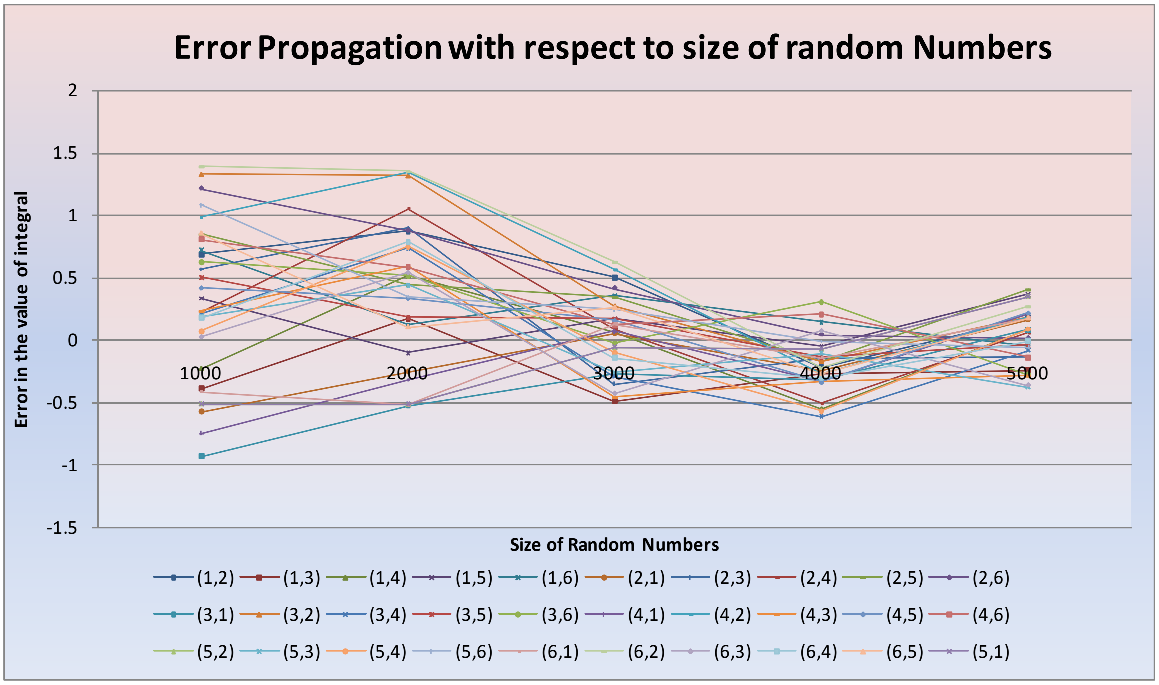

***Figure* 4.3:** Value of second Integral corresponding to different combination of random numbers

4.1.2.2 (Using Equi-spaced Nodes)

In order to evaluate the value of the second integral by using equi-spaced nodes, we will make use of the same program with following modification in line 20 and 30

20 $DEF\ FNI(A, B) = A * B * (1 + A + B)$.

30 $CLS\ : XLO = 1 : XUP = 2 : YLO = 0 : YUP = 3$.

Here we are making equal numbers of divisions in x and y range. Corresponding to different numbers of division starting from 20 up to 400 with step size of 20, we get the following observations.

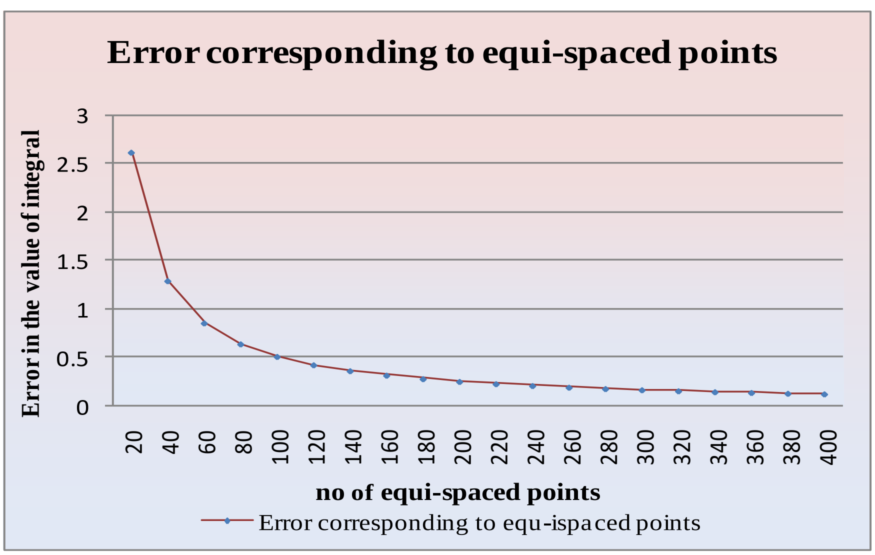

***Figure* 4.4:** Value of second Integral using Equi-spaced points

***Table* 4.6:** Error in the evaluation of the first integral corresponding to equi-spaced points

S. No.	No. of Equi-spaced Nodes	Error
1	20	2.60849
2	40	1.2896
3	60	0.85644
4	80	0.64122
5	100	0.51226
6	120	0.42656
7	140	0.36555
8	160	0.31975
9	180	0.28382
10	200	0.25549
11	220	0.2323
12	240	0.21317
13	260	0.1966
14	280	0.18232
15	300	0.17027
16	320	0.15947
17	340	0.14991
18	360	0.14209
19	380	0.13413
20	400	0.12804

The following ***Fig* 4.4** displays the values of second integral using different size of equi-spaced points.

5 Observations

In the evaluation of the integrals we observe that whatever the combination of files we take the length of the interval of the error decreases as we increase the size of the random numbers [see ***Tables* 4.3** and **4.5**].The supporting results are shown in the following table.

***Table* 5.1:** Length of the interval of error corresponding to different size of random numbers

Integral	**Size of the random numbers**				
	1000	2000	3000	4000	5000
First	.119945	.110287	.059540	.048118	.041154
Second	2.32295	1.87921	1.06433	.92013	.766410

Also the path of error corresponding to each combination while increasing the random numbers seems to be zigzag i.e. the error in the value of both the double integrals corresponding to different size of random numbers, don't follow any pattern and also seems to be random in natural though as per the theory of interpolation a fourth degree polynomial (since the number of data points are five in this case) may be obtained as a trend line to analyze the path of error but it will not be giving satisfactory results as we are dealing with random numbers.

In case of equi-spaced points if we take n points then we are actually dealing with $n \times n$ points i.e. by taking only $n = 100$ function needs to be evaluated for $100 \times 100 = 10,000$ points. Although error follows a smooth pattern and accuracy is assured [see ***Figs.* 4.2** and **4.4**] but here we observe that whatever the accuracy in the value of the integral we obtained corresponding to (say) $400 \times 400 = 1,60,000$ points, same or better accuracy may be achieved by using only 5000 or less random numbers.

6 Conclusion

If the randomness of random numbers is justified with reference to their uniformity and independence then convergence of approximations of a bi-variate integral by Monte Carlo method is assured with the increment in the size of random numbers i.e. if we increase the random numbers used in this process error gets decreased and the propagation of error will be random in nature.

Also we conclude that random numbers plays a better role in comparison to equi-spaced points with reference to the accuracy and computational work involved in the evaluation of the integral and as a result Monte Carlo integration is considered to be time efficient as well. Hence by assuring the randomness of numbers and increment in the size of the numbers,

" Efficiency of random numbers may be accepted in case of bi-variate Monte Carlo integration."

Acknowledgement. We want to thank the editor of the journal and reviewer of my paper for their valuable suggestions and reviews for my paper.

References

[1] L. Bhar, *Non-parametric Test*, Indian Agricultural Statistics Research Institute, Library Avenue, New Delhi, 2004.

[2] R. E. Caflisch, Monte Carlo and quasi-Monte Carlo methods, *Acta Numerica*, **7** (1998), 1-49.

[3] G. J. Chaitin, Randomness and Mathematical Proof, *Scientific American*, **232(5)** (1975), 47-52

[4] R. Eckhardt, Stan Ulam, John von Neumann and the Monte Carlo method, *Los Alamos Science*, Special Issue, **15** (1995), 131-137.

[5] D. Eddelbuettel, *Random: True random numbers using random.org*, 2007. http://www.random.org.

[6] G. Evans, *Practical Numerical Integration* , Wiley, New York, 1993.

[7] J. E. Gentle, *Random Number Generation and Monte Carlo Methods*, 2^{nd} ed., Springer: New York, 2003.

[8] H. Gould, J. Tobochnik and W. Christian, *An Introduction to Computer Simulation Methods: Applications to Physical Systems*, Chapter 11, 3^{rd}ed., Pearson Addison Wesley, 2007.

[9] P. Hellekalek, Good random number generators are (not so) easy to find, *Mathematics and Computers in Simulation,* **46** (1998), 485-505.

[10] M. H. Kalos and P. A. Whitlock, *Monte Carlo Methods*, Wiley-VCH, ISBN9783527407606, 2008.

[11] N. Metropolis, The beginning of the Monte Carlo method, *Los Alamos Science*, Special Issue, **15** (1987), 125-130.

[12] S. K. Park and K. W. Miller, Random number generators: Good ones are hard to find, *Communications of the ACM*, **31** (1988), 1192-1201.

[13] S. Saxena and A. K. Saxena, One Dimensional Numerical Integration by Monte Carlo Method using random and equispaced points, *Journal of International Academy of Physical Sciences*, **17(2)** (2013), 203-222.

[14] G. K. Smyth, *Numerical Integration*, John Wiley & Sons, Ltd, (ISBN 0471 975761) Edited by, Peter Armitage and Theodore Colton, Chichester, 1998.

[15] G.Yang, A Monte Carlo Method of Integration, *Extended Essay in Mathematics*, Denmark, 2002.