Available online www.unicrossjournals.com

UNICROSS JOURNAL OF SCIENCE AND TECHNOLOGY, UJOST

RESEARCH ARTICLE VOL. 2(3) DECEMBER, 2023 ISSN:2814-2233

Date Accepted December, 2023 Pages 147 - 157

A DIVIDED BLOCK RANDOMIZATION TECHNIQUE IN THE TELECOM INDUSTRIES FOR GENERATING MOBILE TELEPHONE NUMBERS

Okim I. Ikpan[1*], John E. Usen[2], and Michael O. Onyegbata[3]

[1,2,3]Department of Statistics, University of Cross River State, Calabar, Nigeria.

[1*]Email:ikpanlouis@gmail.com

[2]Email:johnusen@unicross.edu.ng

[*]Corresponding Author's Email: ikpanlouis@gmail.com

Abstract

This research was an attempt at developing an algorithm that generates mobile telephone lines for operators in the mobile telecommunications industry whilst ensuring enhanced randomness and longevity in the numbers generated. The researchers reviewed some existing pseudorandom number generators with a view to understanding their advantages and lapses. The mid-square generator (MSG) and the linear congruential generator (LCG) algorithms were reformulated and combined to form what is termed a divided block randomization technique (DBRT). The proposed technique is an algorithm made up of three independent components; the first being the fixed block comprising three digits (based on preference), the second and third components being blocks of random numbers generated respectively through the MSM and LCG. The new algorithm has three different settings depending on the number of digits in each block. The proposed technique is implemented using arbitrarily chosen seeds for each setting, generating unique random telephone lines for different network providers. The results in each case of the combination were summarized as mobile telephone numbers with better longevity and enhanced randomness (or very reduced predictability).

Keywords: Random numbers, pseudo-random numbers, mean square generators, linear congruential generators, divided block randomizations

1.0 Introduction

Today, Telecom and Internet services sustain the fabric of modern work and life. Without connectivity, the critical services that keep businesses working, research institutes learning, and transport networks running, and that support people with everything they need from online shopping and mobile banking to healthcare, entertainment, and social contact would become unavailable (Jason Chien, 2023). A mobile phone is today a booster of the new electronic technology. In the digital age, the use of mobile networks is the most proficient of other modes of communication and global interaction. With the digitalization and increased use of many critical services and other digital activities through the customer's network or the telecommunications service operator - from mobile banking and online purchases to messages over social media and work emails, fraudsters have turned mobile phones into a goldmine of financial, health, business, and personal information, raising the need for security to the edge.

Communication infrastructures face numerous cyber threats, ranging from opportunistic and nuisance hacking to organized cyber warfare that may have financial or strategic objectives. In the Telecom industries, while adapting to handle vast quantities of data quickly and efficiently, the networks must meet additional critical requirements such as reliability, ruggedness, and security Research efforts and findings over time have shown that real-life phenomena are predictable to high levels of accuracy. However, while the desire of men is to continue to develop and improve data-dependent techniques based on which predictions about events can be made with high levels of accuracy, there is also a continual aim to develop and improve counterpart data-dependent (CDD) techniques with which he can generate numbers (e.g., mobile numbers, passwords, and PINs) with a high level of randomness or unpredictability. The CDD techniques are termed "pseudo-random number generators (PRNG)".

Telecom Industries require randomness and longevity in their generation of PINs for recharge vouchers, mobile numbers, various self-service customer care direction codes, etc., without which they could experience loss in revenue, especially from continuous system hacks. Pseudo-random number generators may constitute one algorithm[1] they may be made up of a combination of more than one algorithm. Statistically, a combination of more than one algorithm improves randomness. In Nigeria, the telecommunications industry has become a dominant force playing host to several mobile network companies ranging from MTN, to; GLO, Airtel, 9Mobile, etc., and common to all these networks in their operations is the relative absence of randomness and longevity in their generation of PINs for recharge vouchers, mobile numbers, various self-service customer care direction codes, etc. In particular, these networks have changed more than two mobile numbers within their periods of existence due to the poor longevity of their codes, and have also experienced several system hacks due to a high level of non-randomness in their coding system. It is the need for Telecom Industries to come up with a design, structure, functionality, or mechanism for the mobile phone security algorithm, a pseudo-random number generator, that will generate keys to encrypt clients' data securely and beat attackers who aim to steal data and intercept or even disable communications.

In this research, we propose the development of a new algorithm that will overcome the identified flaws, the divided block randomization algorithm for generating mobile telephone numbers through a combination of two varieties of pseudo-random number generators (PRNGs.

2.0 Literature review

Mobile cellular Telecom technology which evolved since the late 1970s in public access, cost, and quality over the years (Chavis, 2021), has enjoyed a rich literature including the works of Beer (2021) who gave the main benefit of mobile telecommunications. Chavis (2021) holds that basic transmission uses a series of protocols to send blocks or packets of information. They give independent opinions on specific actions for two mobile telecommunications devices to connect and receive information. The telecommunications sector consisted of a club of big notational and regional operators (James & Moneta, 2020; Chavis, 2021). (James & Moneta, 2020) maintain that wireless digital technology is becoming the primary form of communication.

Afflerbach, (1990); James & Moneta, (2020) opined that computer-generated random numbers are preferable to manually generated random numbers in terms of the degree of non-randomness. In this light, Van-Niel & Laffan, 2011; Nannipieri, Di-Matteo, Baldanzi, Corcetti, Belli, Fanucci & Saponara, 2021), noted that pseudo-random number generation could be referred to as pseudo-random number simulation. While the term “random” is reserved for the output of an unpredictable process, it is not used for the input. For Van-Niel & Laffan, 2010; Nannipieri, et al., 2021; Yu, Li, Tang, Cai, Song & Xu, 2019), though it is possible to produce an arbitrarily long string of random digits proven as random, it is not easy for humans to produce a string of random digits or computer programs to write them. Van-Niel & Laffan (2010) has it that Random number generation is important in scientific contexts ranging from physical and statistical simulations to cryptography and software testing.

Afflerbach (1990), James & Moneta (2020). outlined the common types of pseudo-random number generators in practice to include the mid-square method, linear congruential generator, combined linear congruential generator, random number streams, multiple recursive generators, Mersene twister generator, well-equi-distributed long period linear generator, and SIMD oriented fast Mersene twister generator. Jacak, et al. (2021) brought up the idea of quantum random number generators and presented a variety of tests utilized to assess the quality of randomness of generated bit sequences. Nannipieri, et al. (2021) designed and validated a digital true random number generator for cryptographically secure applications on field programmable gate arrays. James and Moneta (2020) reviewed pseudorandom number generators (PRNGs) of the highest quality, suitable for use in the most demanding Monte Carlo calculations, and recommended the ones based on the Kolmogorov-Anosov theory of mixing in classical mechanical systems. Yu, et al. (2019) did a systematic review of true random number generators based on chaos.

Van-Niel and Laffan (2010) talked about some characteristics of pseudorandom number generators and specific issues from a geospatial standpoint. Zeitler (2001) developed the machinery for a class of tests of spatial uniformity based on a multi-dimensional Fourier transform of the empirical probability density function. (Afflerbach, 1990) - on the construction of special simulation problems. Afflerbach (1990) discussed the most important criteria. criteria for the assessment of random number generators.

3.0 Materials and methods

Pseudo-random numbers are important in many kinds of technical applications, including Monte Carlo simulations, cryptography, and gambling (on game servers). Many common types of pseudo-random number generators (PRNGs) exist, regardless of quality or applicability to a given use case. Two among them considered in this work are the mid-square method (MSM) and linear congruential generator (LCG)

3.1 The mid-square method (MSM)

Mathematically, the mid-square method is a generator that produces pseudorandom numbers based on the square of a number; it was created by John von Neumann. The method starts with n-digit starting value (the seed). A sequence of n-digit pseudorandom numbers can be generated by squaring the seed, producing either of $2n$ or $2n-1$ digits, from which the resulting middle n digits are taken as the next numbers in the sequence. The resultant number is $2n$ digits if n is even and $2n-1$ digits if n is odd. If the resultant number is $2n-1$ digits for even n, and $2n$ digits for odd n, leading zeroes are added to compensate. The middle of the result would be returned as the random number. This process is then repeated to generate more pseudorandom numbers.

The algorithm for the mid-square method is given as follows:

Step 1: Start with an n-digit natural number (the seed).

Step 2: Compute the square of the n-digit natural number.

Step 3: Take the middle n digits for the next n-digit natural number.

In practice it is a highly flawed method for many practical purposes. it has the disadvantages of self-repeating chain, repeatedly generating the same number or cycle to a previous number in the sequence and loop indefinitely, or getting a chain of 0s.

3.2 The linear congruential generator (LCG)

The LCG is a generator defined by a transfer function of the type

$$y_n = \left(a\, y_{n-1} + c\right) \bmod m,\ n > 0 \qquad (1)$$

where y is the sequence of pseudo-random values, and $m > 0$ the modulus, $0 < a < m$ the multiplier, $0 \le c < m$ the increment, with $a, c, m \in \mathrm{N}$ (the set of all natural numbers), are integer constants that specify the generator.. Typically, c and m are chosen to be relatively prime, but a is chosen such that $\forall y \in \mathrm{N}$, $a\, y \bmod m \ne 0$. The start value $\left(0 < y_0 < m\right)$, which corresponds to $n = 1$ in the recurrence relation (1), is called the seed. The limitation of this generator is that the sequence is not always "random" for all choices of a, c, m, and y_0. The congruential sequences always "get into a loop"; i.e., there is ultimately a cycle of numbers that are repeated endlessly. The linear congruential generator is either multiplicative or mixed, depending on the value of the increment c. when $c = 0$, the generator is a multiplicative congruential method and when the $c \ne 0$, it is of mixed congruential method. The values $m, a, c,$ *and* y should be chosen appropriately to get a period almost equal to m.

According to L'Ecuyer and Pierre (2017), a benefit of LCGs is that an appropriate choice of parameters results in a period which is both known and long though not the only criterion. They maintained that too short a period is a fatal flaw in a pseudorandom number generator. Press, William (*1992)*. Knuth, Donald *(1997)*, Steele, Guy*;* Vigna, Sebastiano *(2020)*,. Marsaglia, George *(1968)*, Park, Stephen.; Miller, Keith. (1988), Hörmann, Wolfgang; Derflinger, Gerhard (1993), and. L'Ecuyer, Pierre (1999*)*,stated that LCGs may be capable of producing pseudorandom numbers which can pass formal tests for randomness, but the quality of the output is extremely sensitive to the choice of the parameters m and a. That for $a = 1$ and $c = 1$, it produces a simple modulo-m counter, which has a long period, but is obviously non-random.

There are three common families of parameter choice: m prime, $c = 0$ and m a power of 2, $c = 0$. When $c \neq 0$, correctly chosen parameters allow a period equal to m, for all seed values. This will occur if and only if

1. m and c are relatively prime,
2. $a-1$ is divisible by all prime factors of m,
3. $a-1$ is divisible by 4 if m is divisible by 4.

Hull, and Dobell, (1962), Severance, Frank (2001), refer to these three requirements as the Hull–Dobell Theorem.

Although the Hull–Dobell theorem provides maximum period, it is not sufficient to guarantee a *good* generator. Knuth and Donald *(1997)*, hold that most multipliers produce a sequence which fails one test for non-randomness or another, and finding a multiplier which is satisfactory to all applicable criteria is quite challenging.

3.3 The proposed divided block randomization technique (DBRT)

The proposed divided block randomization technique is an algorithm formed by combining the MSG of Von-Newmann and Metropolis and the LCG of D. H. Lehmer. This proposed algorithm is presented as follows:

Step 1: Specify a fixed choice number $U_j^{(0)}\ (j \in \square,\ 0 < j < 9)$ of three digits to occupy a constant block. This choice number may depend on the network service provider.

Step 2: Generate the output $U_j^{(1)}$ $(j \in \square,\ 0 < j < 9)$ of $n_i^{(2)}$, $i = 1,2,3$ $\left(n_1^{(2)} = 4, n_2^{(2)} = 5, n_3^{(2)} = 6\right)$ digits for the second block using the MSG method with an appropriate seed.

Step 3: Generate the output $U_j^{(2)}$ $(j \in \square,\ 0 < j < 9)$ of $n_i^{(1)}$, $i = 1,2,3$ $\left(n_1^{(1)} = 4, n_2^{(1)} = 3, n_3^{(1)} = 2\right)$ digits for the third block using the LCG method with an appropriate seed. The digits in each output may be less or greater than the required digits. In this case, add a corresponding number of zeros or truncate at the required number, respectively.

Step 4: Combine the outputs U_0, $U_j^{(1)}$, and $U_j^{(2)}$ to form a single block of $n_j^{(D)} = 11$ digits; the divided block of the form,

$$D = U_0 \left| U_j^{(1)} \right| U_j^{(2)}.$$

Table 1 gives the DBRT output for various values of $n_i^{(\square)}$; where $n_i^{(\square)}$, $i = 1,2,3$, is the number of digits for the output variables $U_j^{(1)}$ and $U_j^{(2)}$, with $U_j^{(0)}$ remaining constant.

Table 1: The divided block randomization output

$n_i^{(\square)}$	$U_j^{(0)}$	$U_j^{(1)}$	$U_j^{(2)}$	$D = U_j^{(0)} \| U_j^{(1)} \| U_j^{(2)}$
$n_1^{(\square)}$	3	4	4	11
$n_2^{(\square)}$	3	5	3	11
$n_3^{(\square)}$	3	6	2	11

3.4 Justification for the proposed technique

Tendencies, abound that the proposed divided block randomization technique could ensure longevity in the generation of random numbers as it overcomes the traditional problems of the LCG and MSM with the modification proposed in the study. More so, the algorithm of the proposed method surely improves randomness in the generation as the combination of the proposed blocks makes predictability of the digits in each block almost impossible.

4. Results and discussions

We demonstrate generating telephone lines by the proposed divided block randomization technique starting with a fixed block of three digits for four different network service providers; the Mtn, Glo, Airtel, and 9Mobile. In generating a unique mobile telephone line, we start with a fixed block of three digits (say, 080) in the first block, implement MSM with an arbitrarily chosen seed (say, $U_0^{(1)} = 221921$) and obtain numbers for the second block and next, we obtain the numbers in the third (final) block by implementing the LCG component of our proposed algorithm using (say, $(y_0, a, c, m) = (10, 2, 13, 19)$), y_0 being the second arbitrarily chosen seed. The results for the three blocks are then combined in the order – first block (080), second block (MSM output), and third block (LCG output).

Tables 1, 2, and 3 show the divided block randomization results for the three blocks of sizes $n^{(0)} = 3(\text{constant})$, $n^{(1)} = 4$, and $n^{(2)} = 4$, as the number of digits, and with different and appropriate seeds

Table 1: Divided blocks with seed $U_0^{(1)} = 2520$ and $(y_0, a, c, m) = (1916, 13, 113, 3017)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	070	2520	1916	07025201916
1	070	5040	0885	07050400885
2	070	4016	2567	07040162567
3	070	1282	0297	07012820297
4	070	4352	0957	07043520957
5	070	9399	0486	07093990486
6	070	3412	0397	07034120397
7	070	6417	2257	07064172257
8	070	1778	2301	07017782301
9	070	6128	2873	07061282873

Table 2: Divided blocks with seeds $U_0^{(1)} = 1974$ and $(y_0, a, c, m) = (133, 5, 577, 1213)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	080	1974	0133	08019740133
1	080	9667	0171	08096670171
2	080	4508	0021	08045080021
3	080	3220	0023	08032200023
4	080	3684	0029	08036840029
5	080	5718	0047	08051780047
6	080	6955	0101	08069550101
7	080	3720	0037	08037200037
8	080	8384	0071	08083840071
9	080	2914	0060	08029140060

Table 3: Divided blocks with seeds $U_0^{(1)} = 7052$ and $(y_0, a, c, m) = (93, 5, 317, 1411)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	091	7052	0093	09170520093
1	091	7307	0443	09173070443
2	091	3922	0596	09139220596
3	091	3820	0256	09138200256
4	091	5924	0698	09159240698
5	091	1937	0970	09119370970
6	091	5196	0052	09151960052
7	091	9989	0730	09199890730
8	091	6802	4195	09168024195
9	091	2672	0209	09126720209

Table 4: Divided blocks with seed $U_0^{(1)} = 43339$ and $(y_0, a, c, m) = (27, 3, 17, 313)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	071	43339	027	07143339027
1	071	82689	215	07182689215
2	071	74707	036	07174707036
3	071	11358	188	07111358188
4	071	90041	268	07190041268
5	071	73816	195	07173816195
6	071	88018	289	07188018289
7	071	71683	258	07171683258
8	071	84524	165	07184524165
9	071	43065	199	07143065199

Tables 4, 5, and 6 show the divided block randomization results for of sizes $n^{(0)} = 3(\text{constant})$, $n^{(1)} = 5$, and $n^{(2)} = 3$

Table 5: Divided blocks with seeds $U_0^{(1)} = 28314$ and $(y_0, a, c, m) = (17, 3, 73, 113)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	081	28314	017	08128314017
1	081	16825	091	08116825091
2	081	30806	007	08130806007
3	081	90096	094	08190096094
4	081	72892	016	08172892016
5	081	32436	008	08132436008
6	081	20940	097	08120940097
7	081	84836	025	08184836025
8	081	71468	035	08171438035
9	081	76750	065	08176750065

Table 6: Divided blocks with seeds $U_0^{(1)} = 80155$ and $(y_0, a, c, m) = (151, 11, 67, 213)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	090	80155	151	09080155151
1	090	48240	024	09048240024
2	090	70976	118	09070976118
3	090	75925	087	09075925087
4	090	16056	172	09016056172
5	090	77951	042	09077951042
6	090	63594	103	09063594103
7	090	29250	135	09029250135
8	090	55625	061	09055625061
9	090	41406	099	09041406099

Tables 7, 8, and 9 show the divided block randomization results for of sizes $n^{(0)} = 3(\text{constant})$, $n^{(1)} = 5$, and $n^{(2)} = 3$

Table 7: Divided blocks with seed $U_0^{(1)} = 95235$ and $(y_0, a, c, m) = (11, 5, 23, 37)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	071	952350	11	07109523511
1	071	304705	04	07130470504
2	071	451370	06	07145137006
3	071	734876	16	07173487616
4	071	042735	29	07104273529
5	071	655280	20	07165528020
6	071	391878	12	07139187812
7	071	568366	09	07156836609
8	071	039909	31	07103990931
9	071	272828	30	07127282830

Table 8: Divided blocks with seeds $U_0^{(1)} = 221921$ and $(y_0, a, c, m) = (10, 2, 13, 19)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	080	221921	10	08022192110
1	080	248930	14	08024893014
2	080	966144	03	08096614403
3	080	434228	00	08043422800
4	080	553955	13	08055395513
5	080	866142	11	08086614211
6	080	201964	01	08020196401
7	080	894572	15	08089457215
8	080	259063	05	08025906305
9	080	136379	04	08013637904

4.1 Discussion of results

The above results are grouped into three sets of telephone lines generated according to the number of digits in each of blocks two and three. The constant blocks reflect the identities of the different network providers, 070, 071, 080, 081, 090, and 091. The results show that divided block randomization technique (DBRD) justifies the modification of the MSM to contain the squares of odd number digits in generating the next required random numbers.

Table 8: Divided blocks with seeds $U_0^{(1)} = 198131$ and $(y_0, a, c, m) = (16, 5, 83, 97)$

j	Bk1 $U_j^{(0)}$	Bk2 $U_j^{(1)}$	Bk3 $U_j^{(2)}$	Telephone line D
0	091	198131	16	09119813116
1	091	558931	66	09155893166
2	091	403862	25	09140386225
3	091	104515	14	09110451514
4	091	233852	56	09123385256
5	091	867579	72	09186757972
6	091	693321	55	09169332155
7	091	694009	67	09169400967
8	091	648492	30	09164849230
9	091	541874	39	09154187439

5. Conclusion

From the results it can be concluded that a variety of pseudorandom number generators can be combined to generate mobile telephone numbers. The divided block randomization technique produced independent improvements of the mid-square method (MSM) and the linear congruential generator (LCG). It is therefore a variant method for generating mobile numbers, and as well, recharge voucher PINs, various self-service customer care direction codes, etc., possessing very high degrees of randomness. The divided block algorithm will invariably, help the Telecom Industries in preventing revenue loss attributable to system hacks; providing a ley way for telecom operators, app developers and end-users.

6. Future scope

This study will no doubt become a major contribution to an already existing bank of literature on the theory and applications of statistical computing. The researchers therefore recommend the development of computer programming languages codes for the proposed divided block randomization technique for the generation of thousands to millions of mobile telephone lines.

References

Afflerbach L. (1990) Criteria for the assessment of random number generatorsJournal of Computational and Applied MathematicsVolume 31, Issue 1, Pages 3-10

Chavis, J. T. (2021), Improving Service through Simulation: A Principled Approach to Decision Making Using Stochastic Service Networks,Cornell University ProQuest Dissertations Publishing, 28494224

Hörmann, Wolfgang; Derflinger, Gerhard (1993). "A Portable Uniform Random Number Generator Well Suited for the Rejection Method" (PDF). *ACM Transactions on Mathematical Software. **19** (4): 489–495.*

Hull, T. E.; Dobell, A. R. (July 1962). "Random Number Generators" (PDF). *SIAM Review. **4** (3): 230–254.*

James, L. Moneta (2020), “Review of high-quality random number generators,” Computing and Software for Big Science, doi: 10.1007/s41781-019-0034-3.

Jacak M. M., Jóźwiak, P., Niemczuk, J. & Jacak, J. E. (2021), Quantum generators of random numbers Scientific Reports, Spriger.

Jason Chien (2023) Journal of social science How FerriSSD Ensures Availability, Longevity, and Security in Networking and Telecom

Knuth, Donald *(1997). Semi numerical Algorithms.* The Art of Computer Programming. *Vol. 2 (3rd ed.). Reading, MA: Addison-Wesley Professional. pp. 10–26.*

L'Ecuyer, Pierre (1999). "Tables of Linear Congruential Generators of Different Sizes and Good Lattice Structure". Mathematics of Computation. *68 (225): 249–260.*

L'Ecuyer, Pierre (13 July 2017). History of Uniform Random Number Generation (PDF). *Proceedings of the 2017 Winter Simulation Conference. Las Vegas, United States.* hal-01561551.

Marsaglia, George *(September 1968).* "Random Numbers Fall Mainly in the Planes" (PDF). PNAS. ***61** (1): 25–28.*

Nannipieri, P., Di-Matteo, S., Baldanzi, L., Corcetti, L., Belli, J., Fanucci L. & Saponara, S. (2021) True Random Number Generator Based on Fibonacci-Galois Ring Oscillators for FPGA;*Applied Sciences.* , *11*(8), 3330; https://doi.org/10.3390/app11083330

Park, Stephen K.; Miller, Keith W. (October 1988). "Random Number Generators: Good Ones Are Hard To Find" (PDF). Communications of the ACM. ***31** (10): 1192–1201.*

Press, William H.; et al. (1992). Numerical Recipes in Fortran 77: The Art of Scientific Computing *(2nd ed.).* ISBN 978-0-521-43064-7.

Severance, Frank (2001). System Modeling and Simulation. John Wiley & Sons, Ltd. p. 86. ISBN 978-0-471-49694-6.

Steele, Guy*;* Vigna, Sebastiano *(15 January 2020). "Computationally easy, spectrally good multipliers for congruential* *pseudorandom number generators"*. Mathematics of Computation. https://github.com/vigna/CPRNG.

Van Niel K.P. and Laffan S. W. (2011), There is no good excuse for a bad random number generator: a reply to Barry; International Journal of Geographical Information Science, Volume 25, 2011 – Issue 4, Pages 531-539

.

Yu, F., Li, L., Tang, Q., Cai, S., Song & Xu, Q. (2019), A Survey on True Random Number Generators Based on Chaos, Discrete Dynamics in Nature and Science, Volume 2019, https://doi.org/10.1155/2019/2545123

.