Trends in Applied Sciences Research

ISSN 1819-3579

Academic
Journals Inc.

www.academicjournals.com

Trends in Applied Sciences Research 7 (1): 96-102, 2012
ISSN 1819-3579 / DOI: 10.3923/tasr.2012.96.102

Fuzzy Cellular Automata Based Random Numbers Generation

[1]Ramin Ayanzadeh, [1]Azam S. Zavar Mousavi and [2]Ehsan Shahamatnia
[1]Department of Computer, Bojnourd Branch, Islamic Azad University, Bojnourd, Iran
[2]Department of Electrical and Computer Engineering, UNINOVA, FCT, Universidade Nova de Lisboa, Caparica, Portugal

Corresponding Author: Ramin Ayanzadeh, Department of Computer, Bojnourd Branch, Islamic Azad University, Bojnourd, Iran

ABSTRACT

Due to the steady increasing trend toward computer simulations, generating random numbers has attracted many researches and several techniques have been introduced in recent years. In this study by employing fuzzy operators on the structure of cellular automata update rules a new approach has been proposed for uniformly distributed random number generation. The simulation results show that the uniformity of the proposed method is very promising. The nature of this approach also makes it very suitable for hardware implementation.

Key words: Cellular automata, computational simulations, fuzzy operators, fuzzy transition rules, random number generation, random variable

INTRODUCTION

The dynamic nature of complex systems around us leads us to think that many of real world processes are stochastic. Hence, the increasing trend in simulating complex systems in recent decades has mad the generation of random numbers an important research field (Ayanzadeh *et al*., 2010; Moghaddas *et al*., 2008; Jang *et al*., 2005). Lotteries, computer games, cryptography, Monte Carlo based computation, computer simulations, operational research and most of intelligent optimization approaches such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), tabu search and etc., are among the vastly used applications of random number generators (Ayanzadeh *et al*., 2009b, 2011; Shahamatnia *et al*., 2011; Banks *et al*., 2004). One of the primary techniques in random number generation is use of various mappings based on parameters such as time i.e., system clock (Sarkar, 2000). To this end, starting from an initial state and employing a recursive equation, a sequence of random numbers is generated. Linear congruential generator, multiple recursive generators and lagged Fibonacci generator are among these techniques (Benkiniouar and Benmohamed, 2004; Sarkar, 2000; Viega, 2003).

Such processes generate a sequence of random numbers which in practice are repeated after a (usually) long period. Random numbers generated within computer applications are sensitive to variables such as the state of initialization and are not completely random. Such systems are called pseudo random number generators (Ayanzadeh *et al*., 2010, 2009a; Jaberi *et al*., 2011; Sarkar, 2000).

Random number generators must be compatible with a variety of statistical distributions e.g., uniform, normal, exponential, Poisson, Erlang, etc. Due to the extensive use of uniform distribution in random number generation, this study also addressed uniform distribution but the proposed

approach is extendable to other distributions as well. The efficiency of random number generators is evaluated considering criteria such as long period frequency, fast and easy computation, low correlation between generated random numbers and better conformance with the desired distribution.

CELLULAR AUTOMATA

With respect to the nature of cellular automata in complex system simulations, it is widely used for random number generation (Szaban *et al.*, 2005; L'ecuyer, 1998). Fast algorithm, parallelization capability, hardware implementation capability and generation of better random numbers are the advantages of these approaches (Brent, 1994).

Cellular automata are discrete computational models that consist of lattice of identical cells communicating within a neighborhood structure. Many structures have been suggested for neighborhood but Moor neighborhood model and Neumann neighborhood model are two widely used. Figure 1 illustrates the Moor and Neumann neighborhood structure with neighborhood radius 1 and 2 (Brent, 1994; Azghadi *et al.*, 2007). The state (value) of cells is derived from a finite set and in each iteration is updated considering current state of cell and current state of the cell's neighbors, according to governing rules which are equally applied to all cells (Brent, 1994).

Binary cellular automata are one of widely used methods to model CA. In binary cellular automata, state of each cell can be either zero or one. Update rules are obtained from Boolean algebra rules which are based on basic AND, OR and NOT operators. The decimal value of bit sequence for next cell state is a method to name the CA rules, first suggested by Wolfram and widely adopted (Bar-Yam, 1997).

Some of the most employed binary cellular automata update rules which are named under Wolfram scheme are shown in Table 1. Table 1 the first line is the current state of the cell (the middle bit) along with right neighbor's states (right bit) and left neighbor's state (left bit). The next state of the cell according to each rule is shown in the lines below.

Initiating from a random state and applying the update rules shown in Table 1, the binary cellular automata generates pseudo random numbers. Due to the locality of update rules, the repeat period of cellular automata in generating pseudo random bits is admissible (Bar-Yam, 1997).

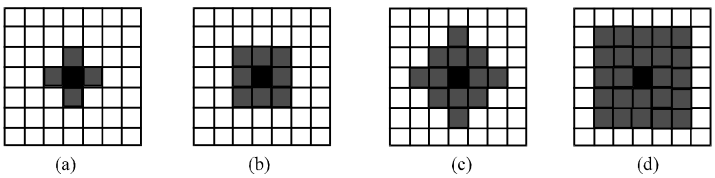

Fig. 1(a-d): (a) and (b) are Neumann and Moor neighborhood structures respectively, with neighborhood radius 1. (c) and (d) are Neumann and Moor neighborhood structures, respectively with neighborhood radius 2

Table 1: Update rules for the binary cellualr automate

Rule	000	001	010	011	100	101	110	111
0	0	0	1	1	1	1	0	30
0	1	0	1	0	1	1	0	90
0	1	1	0	1	0	1	1	105
1	0	0	1	0	1	1	0	150
1	0	1	0	0	1	0	1	165

FUZZY SETS AND FUZZY OPERATORS

Fuzzy set theory maps the state space {0, 1} in classic logic (binary valued logic) to the domain [0, 1] (infinite valued logic). For each member x of set A, membership function determines the measure of existence of x to set A and is shown as $\mu_A(x)$ (Rozyyev *et al*., 2011; Jang *et al*., 2005; Khanale and Ambilwade, 2011). Fuzzy sets are an extension to the classical sets and likewise fuzzy operators are an extension to Boolean algebra operators. The operators fuzzy complement, fuzzy s-norm and fuzzy t-norm are equivalents of operators NOT, OR and AND in binary valued logic (Darestani and Jahromi, 2009; Hatami-Marbini *et al*., 2009; Jang *et al*., 2005).

Various classes of fuzzy operators have been suggested for different types of applications. Eq. 1 shows the standard fuzzy complement, Eq. 2 shows the Yager class fuzzy complement and Eq. 3 shows the Sugeno class fuzzy complement. In these equations $\mu_A(x)$ specifies membership value of x in A. Also λ and ω are fuzzy complement parameters (Jang *et al*., 2005):

$$C_p(a) = \bar{a} = 1\text{-}a \tag{1}$$

$$C_\omega(a) = \bar{a} = \left(1 - a^\omega\right)^{\frac{1}{\omega}} \qquad \omega \in (0,\infty) \tag{2}$$

$$C_\lambda(a) = \bar{a} = \frac{1-a}{1+\lambda a} \qquad \lambda \in (-1,\infty) \tag{3}$$

Fuzzy s-norm of classes Dombi, Dubois-Prade, Yager and Max are the most commonly used and are provided at Eq. 4 to 7, respectively. In these equations $a = \mu_A$, $b = \mu_B$ and λ and ω are related parameters (Jang *et al*., 2005):

$$s_\lambda(a,b) = \frac{1}{1+\left[\left(\frac{1}{a}-1\right)^{-\lambda} + \left(\frac{1}{b}-1\right)^{-\lambda}\right]^{\frac{1}{\lambda}}} \qquad \lambda \in (0,\infty) \tag{4}$$

$$S_\alpha(a,b) = \frac{a+b-ab-\min(a,b,1-\alpha)}{\text{Max}(1-a,1-b,\alpha)} \qquad \alpha \in [0,1] \tag{5}$$

$$S_\omega(a,b) = \min\left(1, \left(a^\omega + b^\omega\right)^{\frac{1}{\omega}}\right) \qquad \omega \in (0,\infty) \tag{6}$$

$$S_{Max}(a, b) = \text{Max}(a, b) \tag{7}$$

For any of various classes of fuzzy s-norm, there is a fuzzy t-norm. Eq. 8 to 11 represent the t-norm formulation in classes Dombi, Dubois-Prade, Yager and Min. Parameters a, b, , λ, ω and α are same as respective s-norm classes (Jang *et al*., 2005):

$$T_\lambda(a,b) = \frac{1}{1+\left[\left(\frac{1}{a}-1\right)^{\lambda} + \left(\frac{1}{b}-1\right)^{\lambda}\right]^{\frac{1}{\lambda}}} \qquad \lambda \in (0,\infty) \tag{8}$$

$$T_{\alpha}(a,b)=\frac{ab}{Max(a,b,\alpha)} \quad \alpha\in[0,1] \tag{9}$$

$$T_{\omega}(a,b)=1-\min\left(1,\left((1-a)^{\omega}+(1-b)^{\omega}\right)^{\frac{1}{\omega}}\right) \quad \omega\in(0,\infty) \tag{10}$$

$$T_{min}(a,b)=\min(a,b) \tag{11}$$

FUZZY CA FOR RANDOM NUMBER GENERATING

The basis of most random number generators is to generate a sequence of random bits, convert the base and map it to the desired domain. Using binary cellular automata, one dimensional CA and applying the update rules such as rules 30, 90, 105, 150 and 165 (by Wolfram naming) one can generate pseudo random bits. In each step in linear binary CA one pseudo random bit can be generated. By iterating the process n times and generating n bits, or by running n parallel CA and generating one bit from each, a random number with desired precision in the specified domain is obtained. Variable n is used to adjust the precision.

Generating large number of pseudo random numbers either serially or in parallel, can be very time consuming and burdensome. It is the case for generating high precision decimal fraction random numbers which require many bits. To solve this problem, this study introduces a fuzzy cellular automaton which by exploiting fuzzy operators and mapping the update rules (such as rules 30, 90 and 165) to fuzzy space it can generate a random number in each step of automata. For example rule 90 can be formulated as in Eq. 12.

$$b'=ab\bar{c}+a\bar{b}\bar{c}+\bar{a}bc+\bar{a}\bar{b}\bar{c}=\bar{a}c+a\bar{c}=a\oplus c \tag{12}$$

where, $b = x_i(t)$, $a = x_{i-1}(t)$, $c = x_{i+1}(t)$ and $b' = x_i(t+1)$.

Now by replacing NOT operators by a fuzzy complement operator, OR operators by a fuzzy s-norm class, AND operators by a fuzzy t-norm class, it will give us the fuzzy equivalent of rule 90. There can be various expressions of fuzzy rule 90 by employing different fuzzy operator classes and also by choosing different values for operator variables.

If the cell values of linear CA consist of decimal numbers from domain [0, 1] (the state space is concrete) and border cells are assumed neighbors (linear CA is loop), applying fuzzy rule 90 to update the states of cells leads each of the cells of FCA behave like a random number generator.

The question arise here that in conversion of rule 90 to a fuzzy rule which fuzzy operators must be used and what should be the values of the parameters of these operators. In this study fuzzy complement, s-norm and t-norm operators are all from Yager class and value of parameter ω in all operators is same and equal to the value of central cell. As shown in Eq. 12, the value of cell in time t + 1 only depends on the value of neighbor cells in time t. Using the central cell value as the parameter for fuzzy operator can be useful in increasing the disorder and hence increasing randomness in the generated numbers.

EXPERIMENTAL RESULTS

To evaluate the proposed approach, first the capability of CA in random number generation is evaluated. To this end a binary linear CA is simulated. The number of cells is 100, neighborhood radius is 1 and cell update rule is rule 90. In this simulation 1000 random bits are generated and

number of ones in the sequence is counted. Running the simulation 100 times, statistical indexes average, standard deviation and scattering length is computed for the number of ones in sequences of 1000 bits. The simulation result is shown in Table 2.

The CA approaches proves to be a successful way in generating random numbers in that the generated numbers are almost uniformly distributed. To evaluate the performance of the proposed approach, the generated numbers are compared with the numbers generated with proposed FCA and numbers generated with MATLAB random number generator.

To this end, 800'000 random number is generated with FCA and MATLAB RNG, then each of the random number sets are divided into 20 equal intervals and the frequency of random numbers belonging to each interval is computed. Finally, the average, standard deviation and scattering length for every interval of each set is calculated. Table 3 presents the statistical indexes for FCA and MATLAB random number generator. According to the results the proposed FCA using the fuzzy rule 90 outperforms the MATLAB RNG in the terms of uniformity.

Figure 2 represents the histogram diagram of dividing FCA generated random numbers into sub-intervals and Fig. 3 represents the same diagram for random numbers generated with MATALB RNG. As it can be seen from the Fig. 2 and 3, the uniformity of random numbers generated with FCA RNG is improved compared to the MATLAB RNG.

According to the simulations fuzzy cellular automata incorporating the fuzzy rule 90 is capable of generating more unified random numbers and if implemented on hardware it can generate a random number on each clock pulse. Therefore, the very high speed of this approach along with the quality of generated random numbers is the advantages of the proposed approach. Meanwhile,

Table 2: Statistical index for evaluation of CA in random number generation

Average	Standard deviation	Setting length
MATLAB		
499.9486	15.6328	86
40000	203.4361	813

Table 3: Statical index for uniformity evaluation of proposed FCA compared to MATLAB RNG

Approach	Average	Standard deviation	Scattering length
FCA	40000	111.9324	504

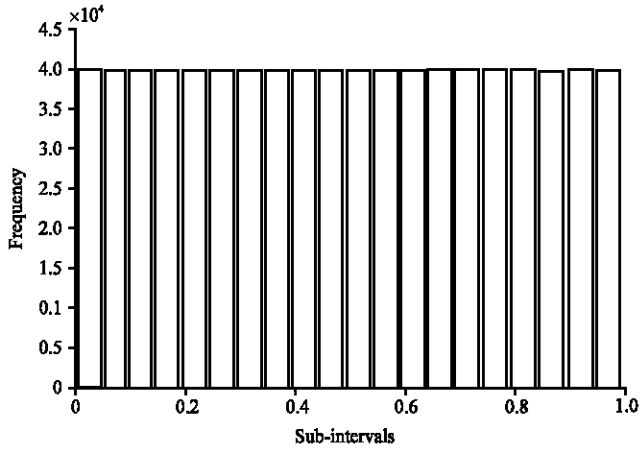

Fig. 2: Histogram diagram for FCA random number generator

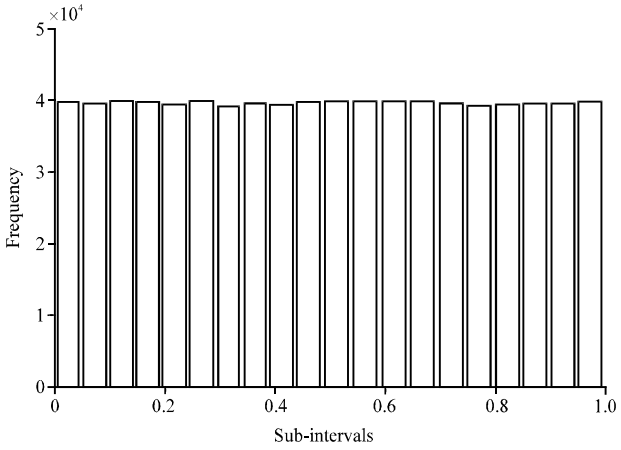

Fig. 3: Histogram diagram for MATLAB Random Number generator

the initialization of automaton is yet an issue, as it can lead the automaton to the Garden of Eden configuration which results in short frequency periods in the generated numbers.

CONCLUSION

This study introduces a novel approach for uniformly generating random numbers by modifying fuzzy operators to operate on the cellular automata update rules. Employing different classes of fuzzy operators complement, s-nor and t-norm makes the FCA behave in different ways. The simulation results showed that the proposed approach generates more uniform random numbers compared to the MATLAB RNG. The nature of the proposed approach makes it suitable for hardware implementation where it can generates the random numbers in a fast way.

As an open issue is the initialization of first run of FCA which is yet to be solved. An unsuitable initialization may lead the FCA to be trapped in Garden of Eden configuration. Considering the applicability of various update rules other than rule 90, studying the impact of different neighborhood structures of CA, combining different fuzzy operator classes and optimally tuning the relative variables for random number generation are the other issues which can be addressed to increase the quality of the random numbers generated within this approach.

REFERENCES

Ayanzadeh, R., A.S. Zavar Mousavi and H. Navidi, 2011. Honey bees foraging optimization for mixed nash equilibrium estimation. Trends Applied Sci. Res., 6: 1352-1359.

Ayanzadeh, R., E. Shahamatnia and S. Setayeshi, 2009a. Determining optimum queue length in computer networks by using memetic algorithms. J. Applied Sci., 9: 2847-2851.

Ayanzadeh, R., K. Hassani, Y. Moghaddas, H. Gheiby and S. Setayeshi, 2009b. Innovative approach to generate uniform random numbers based on a novel cellular automata. J. Applied Sci., 9: 4071-4075.

Ayanzadeh, R., Y. Moghaddas, S. Setayeshi, K. Hassani and H. Gheiby, 2010. Multi-layer cellular automata for generating normal random numbers. Proceedings of the 18th Iranian Conference on Electrical Engineering, May 11-13, 2010, Isfahan, Iran, pp: 495-500.

Azghadi, M.R., O. Kavehei and K. Navi, 2007. A novel design for quantum-dot cellular automata cells and full adders. J. Applied Sci., 7: 3460-3468.

Banks, J., J. Carson, B.L. Nelson and D. Nicol, 2004. Discrete-Event System Simulation. 4th Edn., Prentice Hall, UK.

Bar-Yam, Y., 1997. Dynamics of Complex Systems. Addison Wesley, UK.

Benkiniouar, M. and M. Benmohamed, 2004. Cellular automata for cryptography. Proceedings of the International Conference on Information and Communication Technologies: From Theory to Applications, April 19-23, 2004, UMC., Algeria, pp: 423-424.

Brent, R.P., 1994. On the periods of generalized Fibonacci recurrences. Math. Comput. Conf., 63: 389-401.

Darestani, A.Y. and A.E. Jahromi, 2009. Measuring customer satisfaction using a fuzzy inference system. J. Applied Sci., 9: 469-478.

Hatami-Marbini, A., S. Saati and A. Makui, 2009. An application of fuzzy numbers ranking in performance analysis. J. Applied Sci., 9: 1770-1775.

Jaberi, A., R. Ayanzadeh and A.S. Zavar Mousavi, 2011. Two-layer cellular automata based cryptography. Trends Appl. Sci. Res. (In Press).

Jang, J.S.R., C.T. Sun and E. Mizutani, 2005. Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Prentice Hall of India, New Dehli, India.

Khanale, P.B. and R.P. Ambilwade, 2011. A fuzzy inference system for diagnosis of hypothyroidism. J. Artificial Intel., 4: 45-54.

L'ecuyer, P., 1998. Uniform Random Number Generation. In: Encyclopedia of Computer Science and Technology, Kent, A. and J.G. Williams (Eds.). Dekker, USA., pp: 323-339.

Moghaddas, Y., R. Ayanzadeh and A.T. Hagigat, 2008. A new algorithm for improving the uniformity of random number generators based on calculation with monte carlo method. Proceedings of the of 2nd Joint Congress on Fuzzy and Intelligent Systems, October 28-30, 2008, Tehran, Iran.

Rozyyev, A., H. Hasbullah and F. Subhan, 2011. Indoor child tracking in wireless sensor network using fuzzy logic technique. Res. J. Inform. Technol., 3: 81-92.

Sarkar, P., 2000. A brief history of cellular automata. ACM Comput. Surv., 32: 80-107.

Shahamatnia, E., R. Ayanzadeh, R.A. Rebeiro and S. Setayeshi, 2011. Adaptive imitation scheme for memetic algorithms. Adv. Inform. Commun. Technol., 349: 109-116.

Szaban, M., F. Seredynski and P. Bouvry, 2005. Evolving collective behavior of cellular automata for cryptography. Proceedings of the IEEE Mediterranean: Electrotechnical Conference, May 16-19, 2005, Department of Computer Science, Podlasie University, Siedlce, Malaga, pp: 799-802.

Viega, J., 2003. Practical random number generation in software. Proceedings of the 19th Annual Computer Security Applications Conference, December 8-12, 2003, USA., pp: 129-140.