Research Journal of Computer and Information Technology Sciences ISSN 2320 – 6527
Vol. **6(4),** 1-4, May **(2018)** *Res. J. Computer and IT Sci.*

Short Review Paper

Applied purview of Cellular Automata: a survey

Deepak Kumar Deshmukh
S.O.S. Computer Science and IT, Pt. Ravishankar Shukla University, Raipur, CG, India
deepakdeshmukh.bit@gmail.com

Available online at: www.isca.in
Received 10th April 2017, revised 26th April 2018, accepted 10th May 2018

Abstract

Cellular automaton is a field which is applied in computing to basify the large computing problem to smaller reversible unit. As cellular automata is flexible to break up large problem into smaller one and each small piece of problem can be modelled using cellular automata. This property of CA is widely used for enriching solution to many complex computing problems. Objective of this paper is to jot application of CA explored in last decades.

Keywords: Cellular automata, grid, neighbourhood, reversible computing, parallel computing.

Introduction

The concept of cellular automata is based on modelling of a complex problem by a sequence of relatively simple decision making rules. For engineering implementation it requires to decompose the considered domain into a group of cells which form a uniform lattice. The particular cell including cells which surround it is called its neighbourhood. It is assumed that the cells interact only within their neighbourhood[1].

Cellular automata (CA) are one of the earliest models of natural computing. At first John von Neumann introduced CA in the late 1940s were biologically motivated. The objective that CA was to model self-replicating artificial systems that are also computationally universal. Von Neumann investigated for synthetic computing devices that are analogous to human brain in which storing capability and the processing units are kept together, that are massively parallel and that are able to repair and build themselves if provided the necessary raw material[2].

A cellular automaton is a model made of "cell" objects shows following characteristics: The cells collectively form a grid. The dimension of cellular automata must be any finite number.

Neighbourhood of cell→

0	0	1	0	1	1
1	0	0	0	1	1
1	0	1	1	1	0
0	0	1	0	1	1
1	1	0	0	1	0
1	1	1	0	0	1
1	0	0	1	1	1
0	0	1	0	1	0

Figure-1: Grid of cells each 1 or 0.

Each cell have finite number of states. Taking a simple example that has the two state of 1 and 0 (also referred to as "on" and "off" or "alive" and "dead"). Each cell have neighbouring cells at up(U), down(D), left(L), right(R). There are number of ways to define it, but it is typically represented by a list of adjacent cells[3].

Elements of CA

Grid: One-dimensional is a line of cells.

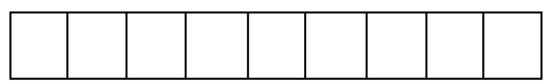

Figure-2: One dimensional grid.

States: Set of states would be two states: 0 or 1.

1	0	1	0	1	1	1	0	0

Figure-3: One dimensional grid with two states 0 and 1(may be on and off also).

Neighbourhood: Neighbourhood of a certain cell in certain dimension is cell itself and its adjacent cells at both side left and right.

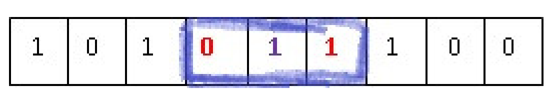

Figure-4: A neighbourhood cell of three cells.

So in a queue of cells, each with an initial state which might be random, and each with two neighbours. We need to figure out how to handle cells on the edgessince those have only one neighbour either at left or right.

1	0	1	0	1	1	1	0	0

Figure-5: Edge cell only has a neighbourhood of two only.

Cellular automata work time is important phenomenon that is the CA, alive for a period of time, which might also be termed as a generation. The figures below show that the CA at its 0^{th} generation or time equals 0. The matter of question is that: What would be the procedure to find new states for all cells at each generation and moving further.

1	0	1	0	1	1	1	0	0

Figure-6: At Generation 0 of CA

?	?	?	?	?	?	?	?	?

Figure-7: At Generation 1 of CA

?	?	?	?	?	?	?	?	?

Figure-8: At Generation 2 of CA

Computer operations can be bifurcated in two categories extremely simple and extremely, complex. The simplicity can be achieved by the operational rules of the computations, and in the engineering structure. The way we apply those rules, on that structure, determine its complexity. The concept is called "cellular automata" (CA)[3].

Working of CA

CA works on structure like grid or lattice like a chase board. Each node of this grid is called cell. The individual cell will not work strongly because it is not capable of computing much individually. All it will require to remember its current state (i.e., its yes/no value), and also the states of its four just adjacent neighbours to the up, down, left and right.

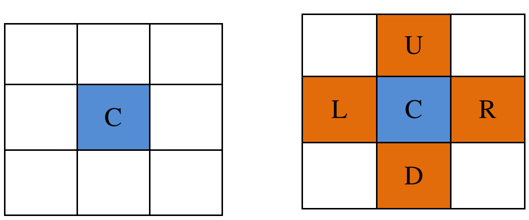

Figure-9(a): The cell. **(b)**: The cell and its neighbours.

Each cell autonomously perform its operations. Individual cell perform as if it is the only computing unit if does not take consideration to the current state of the neighbours at any cycle as mention above. The word “Automata” for it defines this.

Each behaves like an automaton or robot i.e. it follows the set of instruction strictly. In this context, we can consider the automata (automatons) of a cellular automata network woks like computing machine, because the computer executes strict set of instructions (known as program) as an individual cell perform. The intensity of simplicity of the instruction in both varies as instruction of computer program may be complex while cell’s instruction seems to be easier than that. The objective of cell's calculation is to predict it’s value for the next round. If at first round it determine its value “1” then on the basis of this value and the values of its neighbours cumulatively determine, cell’s value on next cycle whether ”0” or “1”.

The transition rule of values at every cycle for a cell can be represented in table. Transition rule would be finite due to finite number of possible situation which causes cell to transit fast.

Considering an example, suppose we have a transition rule that when C="0", while its adjacent neighbours have the values (UP = 0, Right = 0, Down = 1, Left = 1), then the cell C would have new value "1" at the next step. This may be tabulated like in Figure-10. Once every cell completed its one cycle of whole cellular automata then it will move to next cycle[3].

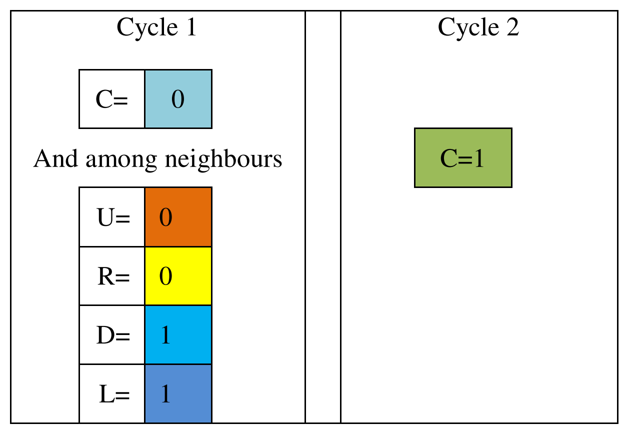

Figure-10: Calculation of cell value.

Application of cellular automata

Reversible computing: In paradigm of reversible computing, computing models are defined so to reflect physical reversibility, as it is essential fundamental physical properties of Nature.

It is easy to define reversible computing system. In these systems, every computation cycle can be traced backward uniquely because each of their computation cycle has at least one previous configuration. Such systems are called backward deterministic system. Physical reversibility is one of the features of CA.CA accommodate physical reversibility at different levels of computing from an elementary level to system level[4].

Logic gates like AND, OR and NAND shows irreversibility as its output cannot be retrieved backward to its input because logic functions of these gates do not map one to one. Unlike to these gates, a NOT gate is reversible as it shows one to one mapping between its input and output[5-6].

Reversible Turing machine is mode of reversible logic elements. Since in theory of computation Turing machine model is benchmark so reversible Turing machines must be studied first[7]. Bennett[8-10] experimented on simulation of irreversible Turing Machine by RTM with three tapes with no garbage information on tapes while halting. Therefore garbage less RTMs are computation-universal and reversible logic elements are used to build RTMs.

Parallel Computing: In area of machine design, parallel multipliers, parallel computers and sorting machines can be built up using CA modelling. Toffoli et al. recognize the CA as fault tolerant computing machine. For image processing and pattern recognition two dimensional CAs are used frequently. CAMs (CA Machines or CA based Machines), developed by Toffoli et al. these CAMs work in autonomous mode. The degree of parallelism is very high in such CAMs, therefore seems well suited for simulation of complex systems. The magnitude of simulation performance of CAMs are comparatively much higher than that of conventional computer. Since the publication of Von Neumann's[11] seminal work in the late 1950s, reveals the possibilities of the artificial self-replicating structures to be used for computations. Das et al.[12] identifies CA as a self-replicating structure, which can solve NP-complete problem. CA is considered as typical computing device at present[6].

Modelling Data Mining Tasks: Data mining researchers focus on mining frequent item set of transaction databases. The researchers presented algorithm to apply cellular automata to mine the frequent item set at the more efficient process time. Research paper presented by Mohammad Karim Sohrabi[13], Reza Roshani[13], Novel CLA based distributed frequent item set mining was represented, which performed on constructed proximity list for the neighbour's iteration for each cell of CLA and then applied this list for mining the frequent item set[13].

Network Modelling: "Cellular automata (CA) are normally modelled as nodes of an N-dimensional grid that forms a square lattice ZN (Conway and Sloane, 1998)[14]. Bonacich[1] demonstrated that this is not the case; that is, dimension counts. He pointed out various ways how grid dimensionality affect the numerous properties of CA networks. Bogdan Bochenek and Katarzyna Tajs-Zielińska[1] discussed Topology optimization algorithm using CA rules. The power-law approach known as SIMP (solid isotropic material with penalization) oftenly used while optimal topology generation, is considered within the CA formalism.

Detection of Malicious Network Packets: Various experiments was conducted by Robert L. Brown[15] to validate that at least some types of malicious network packets can be discriminated from ordinary network traffic using cellular automata.

This research was conducted in four phases. In first Phase specification and establishment of normal and malicious test data packets which were taken from the DARPA Intrusion Detection System Evaluation data for 1998, described by Haines[16] were performed.

In second phase implementation of the cellular automaton and genetic algorithm was proposed. Firstly the cellular automaton test bed was produced. Experiment is initiated with the simple binary, rule-uniform, linear cellular automaton using with r = 1 and 256 cells.

Third phase of research was formulation of a fitness function. Although determination of fitness functions involve cluster analysis, the suitability of functions are computed by combinational logic or direct comparison.

In phase four testing the rules of cellular automaton, evaluation of effectiveness of CA, and refinement of the fitness function iteratively was performed[17].

CA Games: Conway and his colleagues proposed that Cellular Automata have been used for modelling different games. They illustrated how a CA rules can be applied to materialize terribly complex system behaviour such as the game of life. Many variations of the game of life such as games of proto-life which modelled the emergence of a crystalline precursor to life from an initial random prebiotic soup. Among these are the games which provide proper understanding about the synchronization problems-for example, the firing squad, firing mob, and queen bee. A CA simulation has also been proposed for famous game of iterated prisoner'sdilemma[11].

Modelling Physical and Biological Systems

Cellular automata can be a modelling alternative of differential equations in modelling laws of physics. This lead to invention of new CA model for physical system like spin system, models for pattern formation in reaction-diffusion systems, modelling of hydro dynamical systems etc[17].

The Lattice Gas Automaton (LGA) is specifically for the simulation of hydrodynamics and reaction-diffusion processes. Even if LGA shows the discrete dynamics, it simulates the behaviour prescribed by Navier-Strokes equations of hydrodynamics. Celada, Seiden[18], and De Boer et al.[19] explored the successful application of cellular automata in modelling the immune system. CA is successfully applied to modelling for drug therapy for HIV infection, Tumour Development and on detecting genetic disorders of cancerous cells CA has been applied in field of DNA sequences largely. CA has explored lot of biological applications. Author has analysed elementary cellular interaction which is a elementary biological activity using CA modelling, it has also been analysed for modelling adhesive interaction of biological cells without growth or loss of cess applying CA. CA modelling for pattern formation in salamander larvae is analysed and discussed about effect of chemo taxis and pressure on biological pattern formation. By using above mentioned elementary biological pattern formation step, CA is applied to model tumour growth[17].

Conclusion

This paper's objective is to jota detailed survey of the modelling applications of CA done in different fields. The survey also focuses to enquire about different theoretical developments which have taken place over the recent years in the research area of CA and tries to sum it up comprehensively. Mentioned application and developments have been proven as the immense potential of CA in modelling. It also maintains the versatility of the field applied. CA can be applied in many more computing areas to solve complexity of computation problem. Application area of CA maintains the multidimensionality. In present era CA is more useful as most of the systems are being larger every day and faces many challenges so it is difficult to handle lager systems cumulatively while any problem here the CA becomes prominent to resolve its problem by dividing it into smaller pieces. It is deduced that CA has immense scalability to model various system like Biological Systems, Physical Systems, Computer Games, Computer Network Systems etc. and may be applied to resolve complex problems.

References

1. Bonacich P. (2003). Cellular automata for the network researcher. *Journal of Mathematical Sociology*, 27, 263-278.

2. Kari Jarkko (2005). Theory of cellular automata: A survey. ELSEVIER, *Theoretical Computer Science*, 334, 3-33.

3. Daniel Shiffman (2018). The nature of code. http://natureofcode.com/book/chapter-7-cellular-automata.(last accessed 15th Mar, 2018).

4. Morita K. (2008). Reversible computing and cellular automata—A survey. *Theoretical Computer Science*, 395(1), 101-131.

5. Petri C.A. (1967). Grundsätzliches zur Beschreibung diskreter Prozesse. Colloquium über Automatentheorie, Birkhäuser, Basel, 3, 121-140.

6. Fredkin E. and Toffoli T. (1982). Conservative logic. *Int. J. Theoret. Phys*., 21, 219-253.

7. Lecerf Y. (1963). Machines de Turing reversibles-Reursive insolubilite en n∈ N de l'equation u= θ^ nu, ou θ est un isomorphisme de codes. Comptes Rendus hebdomadaires des seances de l'academie des sciences, 257, 2597-2600.

8. Bennett C.H. (1973). Logical reversibility of computation. *IBM journal of Research and Development*, 17(6), 525-532.

9. Bennett C.H. (1982). The thermodynamics of computation—a review. *International Journal of Theoretical Physics*, 21(12), 905-940.

10. Bennett C.H. (1988). Notes on the history of reversible computation. *ibm Journal of Research and Development*, 32(1), 16-23.

11. Ganguly N., Sikdar B.K., Deutsch A., Canright G. and Chaudhuri P.P. (2003). A survey on cellular automata. Technical report, Centre for High Performance Computing, Dresden University of Technology.http://www.cs.unibo.it/bison/publications/CAsurvey.pdf(last accessed 15th Mar, 2018)

12. Das A.K. (1990). Additive Cellular Automata : Theory and Application as a Built-in Self-test Structure. Ph.D thesis, I.I.T., Kharagpur, India.

13. Sohrabi M.K. and Roshani R. (2017). Frequent itemset mining using cellular learning automata. *Computers in Human Behavior*, 68, 244-253.

14. Conway J.H. and Sloane N.J.A. (1998). The Kissing Number Problem" and "Bounds on Kissing Numbers". Ch. 2.1 and Ch. 13 in: Sphere Packings, Lattices and Groups.

15. Brown Robert L. (2014). Application of Cellular Automata to Detection of Malicious Network Packets. Doctoral thesis. Graduate School of Computer and Information Sciences, Nova Southeastern University 2014, FLORIDA.

16. Haines J.W., Rossey L.M., Lippmann R.P. and Cunningham R.K. (2001). Extending the DARPA off-line intrusion detection evaluations. Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX) 2000. Los Alamitos, CA: IEEE Computer Society Press, 1, 35-45.

17. Das D. (2012). A Survey on Cellular Automata and Its Applications. Krishna P.V., Babu M.R., Ariwa E. (eds) Global Trends in Computing and Communication Systems, *Communications in Computer and Information Science*, 269. Springer, Berlin, Heidelberg.

18. Celada F. and Seiden P.E. (1992). A computer model of cellular interactions in the immune system. *Immunology today*, 13(2), 56-62.

19. De Boer R.J., Hogeweg P. and Perelson A.S. (1992). Growth and recruitment in the immune network. *In Theoretical and Experimental Insights into Immunology*, Springer, Berlin, Heidelberg, 223-247.

20. Ross Rhodes (1971). Commentary on Edwin Roger Banks. Information Processing and Transmission in Cellular Automata, Banks' Ph.D. thesis, 1971 (MIT, Dep't of Mechanical Engineering). http://www.bottomlayer.com/bottom/banks/banks_commentary_02.htm. (last accessed 15th Mar,2018).

21. Péli G. (2013). Network Modeling with Cellular Automata: Revisiting the Issue of Dimensionality. *The Journal of Mathematical Sociology*, 37(4), 195-202. DOI: 10.1080/0022250X.2013.799157.