International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 02 | Feb 2022 www.irjet.net p-ISSN: 2395-0072

A Study on Evolving Self-organizing Cellular Automata based on Neural Network Genotypes

Eric Mervin Anandraj[1], Ahnaf Rehan Shah[2]

[1,2]U.G. Student, Dept. of Computer Science Engineering and Technology, Bennett University, Greater Noida, Uttar Pradesh, India

---

Abstract - *This review discusses Wilfried and Istaváns' idea of using Artificial Neural Networks for designing self-organizing systems. Although a lot of researchers have been able to write functions for regenerating simple shapes such as flags and squares, it has been difficult to generate a function to recreate complex images and paintings. We agree with the researchers that using a genotypic template for the cells in the automation is something that should be further researched for generating self-organizing multicellular systems having the capability to recreate complex images. This research and its experiments do not provide concrete results but provide us with sufficient evidence that the evolutionary approach especially using Artificial Neural Network into Cellular Automata has a lot of potential uses and should be further researched.*

Key Words: **cellular automata, artificial neural network, self-organizing systems, evolutionary algorithm**

1. INTRODUCTION

For the last two to three decades a lot of systems have been identified as having self-organizing properties. Self-organizability has been identified not only in the domain of computer science but they were first observed in fields of natural and social science such as physics, chemistry and most extensively in the domain of biology. One naturally occurring example of self-organizing can be the crystallization of chemicals/liquids when frozen at very low temperatures or exposed directly to a gas (external factors).

Self-organizability can be explained as the ability of a system, either natural or artificial, to adapt and modify its internal structure in response to the system's external factors. In the technical field, it is one of the fundamental concepts of Systems Science and such systems are called self-organizing systems (SOS). Elements in such systems not only change their behaviour but also have the power to manipulate other elements in the system for the sake of maintaining the stability of either the system's structure or the function of the whole against external fluctuations.[2] These systems have self-repairing capabilities which for a long time has been only seen in the field of natural science.

W. Ross Ashby's experiments which led to his book on the foundations of cybernetics[1] are the main reason for the modern idea of self-organizability and the recent studies on self-organizing systems. Although this concept has been identified in various systems, both natural and artificial, the fundamental questions of how and why are still unanswered. Due to this uncertainty still being prevalent, there are still a lot of active researches happening in this field.

Scientists up until now, have been able to identify many examples of self-organizing systems in various fields exhibiting the aforementioned characteristics and have studied their mechanism in detail but understanding the design of such systems remains a critical challenge. Recent results however indicate that the evolutionary approach to designing self-organizing systems to be very promising.

2. History of Cellular Automata and Self-Organizing System

Cellular automata (sing. Cellular Automaton) also known as cellular spaces is a system of coloured cells on a grid of usually a specified shape that behaves according to a particular set of rules based on the state of neighbouring cells. The system evolves through several discrete time steps while the rules can be applied iteratively for as many time steps as desired. This concept was first discovered and study on by two scientist-friends Stanislaw Ulam and John von Neumann in the 1940s.

2.1 Game of Life

One of the first examples of cellular automata and the one that presumably popularized this concept is "Game of Life" [8] (or simply "Life") developed by John Horton Conway in the year 1970. Similar to every other cellular automaton, the cells of this system also have only two possible states, either dead or alive. Through each generation of the system, the cells are objected to the following 3 simple rules:

1. Survival: a particular cell can survive for the next generation/round only if it is surrounded by 2 or 3 alive neighbours.

2. Birth: a new cell is born or in other words, a dead cell becomes alive in the next generation if it is surrounded by exactly 3 alive cells.

3. Death: an alive cell can die because of one of two reasons :

(a) Overpopulation: an alive cell dies in the following generation if it is neighboured by 4 or more alive cells.

(b) Loneliness: an alive cell can also die in the next generation if it is not at all surrounded or surrounded by only one alive cell.

2.2 Discovery of Self-Organization

Although the idea of self-organization can be dated back to as early as Greek philosophy and Buddhism, the term was first introduced in the year 1947 by Ashby[11]. Haken further researched in this field and concluded that for any system there is a deep association between self-organization and selection. He also noticed that the different forms of collective behaviour are competing against one another, and compared it with Darwin's Survival of the fittest, calling it the Darwinian selection in the artificial world.

Recently, self-organizing systems are the main points of discussion in all almost all the branches of science including engineering where engineers are starting to see its applicability[10] in relation to the emergence of nano-scale applications and the increasing complexity of human artefacts.

2.3 Self-organization in Game of Life

Conway's system allows us to understand how a cellular automaton behaves under a certain set of rules and also how they exhibit self-organizability in the system. For example, when considering the R-pentomino pattern in the Game of Life system it can be noticed that structures like boats, ships, loaf etc are generated solely through the self-organization of the system. The R-pentomino pattern continues through many generations generating and consuming such structures along the way to finally stabilize at generation 1103.

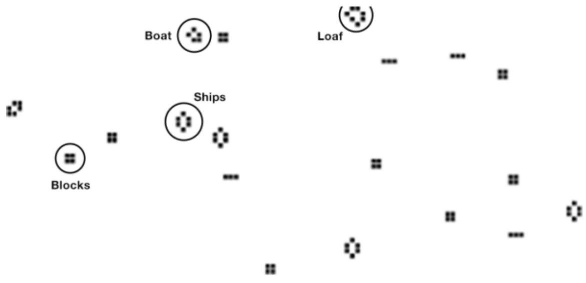

Fig. 1. R-pentomino stabilized after 1103 generations

3. Evolutionary Programming for designing Self-Organizing Systems

Designing a self-organizing system has always proved to be a challenging task for scientists and researchers even after extensive study on the structure and functioning of these systems. Since there is no straightforward way for designing a system with local rules where the system as a whole exhibits self-organizing behaviours.[9]

Researchers resorted to adapting methods inspired by the naturally occurring self-organizing systems[5]. However, self-organizing systems designed based on such inspiration deliver promising results, it requires the existence and discovery of a naturally occurring system that answers the exact same problem at hand.

Evolutionary Approach	Traditional Approach
Supports discontinuous objective function and evolutionary multimodal optimization	Supports linear programming problems and evolutionary singular strategies and optimization
Based on the concept of transitional probabilities	Based on deterministic system
Moves from one point to next point randomly without the current state being dependent on the past state	Moves from one point to next point in search space linearly and no randomness is involved
Fitness function is used to guide simulations towards optimal design solutions	Information derived from previous is used to find the optimal design solution
Parallel search algorithms such as Breadth-first searches is used in the particular search space	Sequential search algorithms such as Linear Search is used in the particular search space

Table 1. Evolutionary vs Traditonal Approach for designing a Cellular Automata

Another approach, the one generally used, was to subject the systems to trial and error. But these systems often exhibited contradictory relationships between the emergent behaviour and the local rules[5]. Thus there was a need for scientists to look into any other possible approach, where there was no need for any knowledge of any existing self-organizing system.

One such promising approach was the use of neural networks to evolve the cells and their local interactions, a method first proposed by Wilfried Elmenreich and István Fehérvári. Their approach was more on the side of incorporating genetic algorithms in the system.[6]

Evolutionary Algorithms are search algorithms that function based on the population. and each algorithm has its own number of approaches. The initial population of the arrangement must have a portrayal in finding the solution. The computational complexity of the issue relies upon the size of the underlying populace.

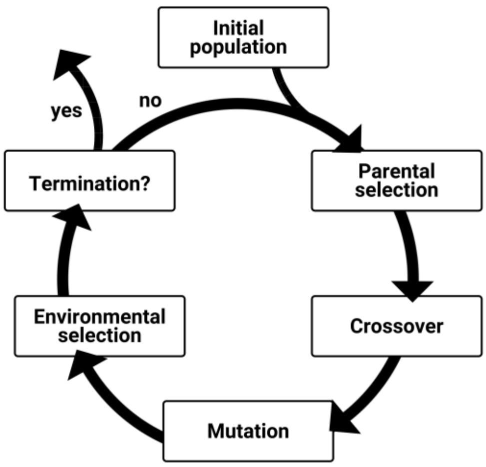

Fig. 2. Evolutionary Computation Cycle

The evolutionary activities, for example, hybrid and transformation are applied iteratively in the evolutionary algorithms until a halting condition is fulfilled. The halting models might be utilized to end when an adequate arrangement has been found or when there is no improvement over various sequential ages. Evolutionary Computation Algorithms varies from the typical Classical techniques in many ways[7] (see Table 1).

Several potential conventional tools have since been founded based on the evolutionary approach which provides increased adaptability, optimization, and scope of solutions. Traditional techniques try to provide one solution which is the most optimal, whereas evolutionary algorithms include and can provide various potential solutions.

4. Model proposed by Wilfried and Istvan

P. Bentley and S. Kumar during their research[3] noticed that when a cellular automaton that is to be evolved becomes more complicated, the evolutionary approach results in various problems such as disruption of inheritance, the method not being able to find the solution to the problem, among others. In technical evolution, there is 1-1 mapping from genotype to phenotype, which is not the case in natural evolution where a single cell can evolve into complex systems. Thus there was a need in cellular automata to introduce genotype descriptions from which complex self-organizing cellular automata could emerge.

Keeping all of this in mind, Wilfried and István developed a model consisting of one genotypical controller that is responsible for the state and transition of each cell in the system[4]. The state of each cell is an instance of this controller and the algorithm responsible is realised with the help of a simple artificial neural network (ANN) which is modelled as a time-discrete and recurrent ANN.

4.1 Structure of the Cellular Automaton Model

Similar to other cellular automata, this model also consists of a grid containing cells, but the dimensions of the model are exactly the same as the reference image. The reference images were often small and ranged from 50-500 pixels in size along one edge.

But unlike Conway's Game of Life, this model has the ability to display colour. A scale is developed from the colours of the reference images by assembling a 24-bit colour code for each colour in where the bits of the 3 channels, Red, Green and Yellow are arranged from most to least significant for their respective channel.

In this model, each cell of the system is under the control of the artificial neural network where all the neurons are connected to each other and also to itself. All the connections between neurons has an allocated weight (w ∈ R) whereas every neuron has a particular bias.

For every generation, an i^{th} neuron builds over the sum of its bias b_i and weights $w_j i$ of the input connections multiplied by current outputs of the neurons j = 1,2,...n feeding the connections. A neuron's output in the next generation can be determined by using the activation function F:

$$o_i(k+1) = F(\sum_{j=0}^{n} w_{ji} o_j(k) + b_i)$$

where F is a simple linear threshold function

$$F(x) = \begin{cases} -1.0 \text{ if } x \leq -1.0 \\ \quad x \quad \text{if } -1.0 < x < 1.0 \\ 1.0 \text{ if } x \geq 1.0 \end{cases}$$

The artificial neural network of each cell is made up of a total of 20 neurons. They are divided into 5 output, 6 hidden and 9 input neurons. The reason for selecting hidden neurons according to Wilfried and István is to find a balance between the neural network's capability and reducing the search space. Out of the 5 output neurons, one determines the colour of the cell and the other output neurons and 4 pairs of the input neurons connect with the neighbouring cells artificial network.

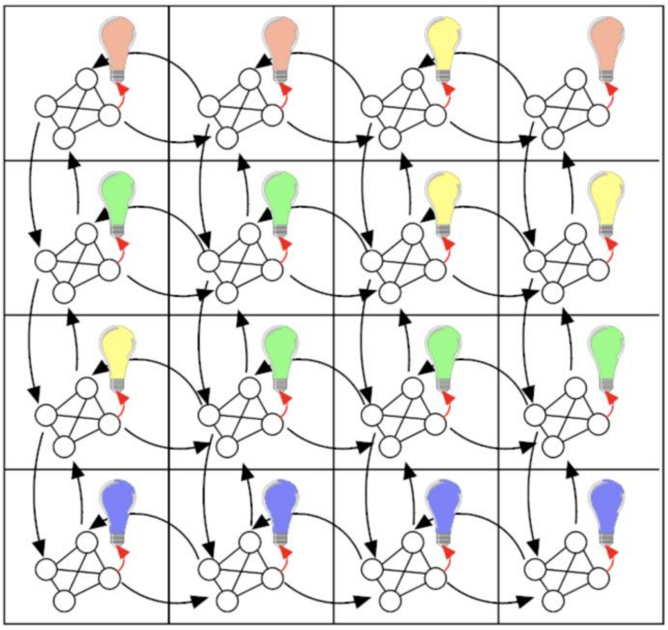

Fig. 3. ANNs interconnected with the other cells

4.2 Evolutionary Programming of the ANNs

Wilfried and István used a Java framework called "Frevo" to evolve the system and study its behaviour with each generation. The software offers the option of clubbing different representations and optimization methods for the problem at hand. In Frevo, each problem has a generic interface to 1 or many instances of the representation, the optimizer evolves the control algorithm that was formed.

Algorithm 1 Evolutionary algorithm used as optimization method

```
create n networks in a population and initialize them with random values

for generations
    for i=0 to n
        evaluate network_p, i and store score
    rank networks according to their score (best first)
    select elitist networks
    select randomly networks (bias for better ranked and diverse networks)
    create mutations of selected networks
    create recombinations of selected networks
    create some networks anew and initialize them with random values
```

Fig. 4. Algorithm used for EA

5. Experiments Conducted

The easiest patterns to replicate using cellular automata are flags, and Figure 2 represents how the Hungarian flag when fed to the developed model behaves at each generation. The complexity of the Hungarian and the French flag are similar as they both are made up of 3 colours and are placed one after the other

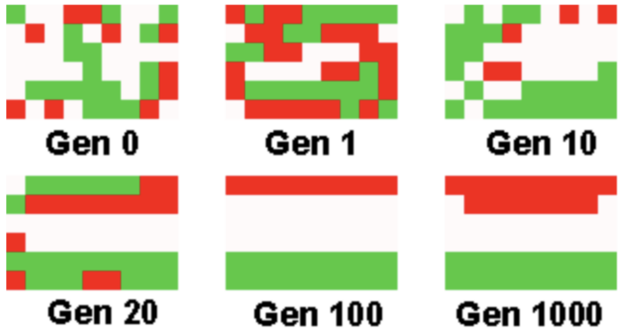

Fig 5. Hungarian flag through generations

It is often observed in evolutionary algorithms that the cellular automata after around a hundred generations reach a maximum accuracy state therefore improvements rarely happen after reaching the local cost minima.

Fig 6. Austrian flag through generations

When subjected to simpler pictures, such as the Austrian flag, which is made up of 2 colours, a perfect reproduction of the image is achieved after only 90 generations (Refer to Figure 6). But when objected to complex paintings such as the Mona Lisa, the best possible reproduction of the downsized image, with the dimensions 20x29, was obtained after 500 generations. The result had proper regeneration of the background however, the subject, Mona Lisa couldn't be processed by the system (Refer to Figure 7). The reason for this can be the distance between the cells at the edges and the cells in the middle of the system, as the information propagates from the edges to the centre of the system.

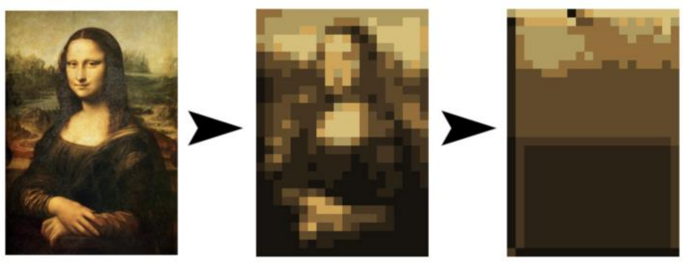

Fig. 7. CA attempting to recreate Mona Lisa

6. Conclusions

Although the model developed by Wilfried and István might not be able to generate a constructive solution, the experiment carried out proves that integrating Artificial Neural Networks into a Cellular automaton is one possible approach to designing a Self-organizing System.

The reason the model is unable to come to a feasible solution would be the poor fitness function used while developing the model. The model for each solution was comparing the image generated with the reference image, pixel for pixel. As a result, a solution quite similar to the reference image was not considered by the system, whereas we humans perceive a similar pattern as being closer to the reference than an image in which only parts of the reference image was reproduced.

The model can be improved by adding more connections between the neurons in order to reduce the time required to propagate data from the corner cells to the center and rewriting the fitness function such that the fitness is based on the type of emerging structure instead of comparing the solution pixel-to-pixel with the reference image

REFERENCES

[1] W. R. Ashby, An introduction to cybernetics, Chapman & Hall Ltd, 1961.

[2] W. Banzhaf, Self-organizing systems., Encyclopedia of complexity and systems science 14 (2009) 589.

[3] P. J. Bentley, S. Kumar, et al., Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem., in: GECCO, Vol. 99, 1999, pp. 35–43.

[4] W. Elmenreich, I. Fehérvári, Evolving selforganizing cellular automata based on neural network genotypes, in: International Workshop on Self-Organizing Systems, Springer, 2011, pp. 16–25.

[5] W. Elmenreich, G. Friedrich, How to design selforganizing systems, 2009.

[6] I. Fehérvári, W. Elmenreich, Evolutionary methods in self-organizing system design., in: GEM, 2009, pp. 10–15.

[7] T. Ilango, S. Murugavalli, Advantage of using evolutionary computation algorithm in software effort estimation, International Journal of Applied Engineering Research 9 (24) (2014) 30167– 30178.

[8] E. M. Izhikevich, J. H. Conway, A. Seth, Game of life, Scholarpedia 10 (6) (2015) 1816.

[9] C. Prehofer, C. Bettstetter, Self-organization in communication networks: principles and design paradigms, IEEE Communications Magazine 43 (2) (2005) 78–85.

[10] G. D. M. Serugendo, M.-P. Gleizes, A. Karageorgos, Self-organising systems, in: Self-organising Software, Springer, 2011, pp. 7–32.

[11] A. WR, Principles of the self-organizing dynamic system., The Journal of General Psychology 37 (2) (1947) 125–128.