Graphics and Multimedia

Game of Life

Project Documentation

by Arman Mann

13BCE0073

and Ishaan Chaturvedi

13BCE0061

slot: E2

Index

1. Abstract

Cellular automata are a simple structure which lends itself to some remarkable ideas. They are simple to construct but have the complex behaviour. They can be studied in physics, mathematics, computer science, biology to study the natural process like self reproduction. In this paper, we are going to show that how Pascal's triangle can be created using cellular automata and how the next generations can be created in the Game of Life.

2. Introduction

The history of the cellular automata starts with the Stanislas Ulam. The curiosity of the Ulam in the evaluation of graphical constructions generated by some simple rules caused the evaluation of the Cellular Automata. There are several types of cellular automata in different dimensions, viz. one dimensional, 2 dimensional etc. Cellular automata can be seen where complicated patterns of behaviours can be produced by many simple components for a system.

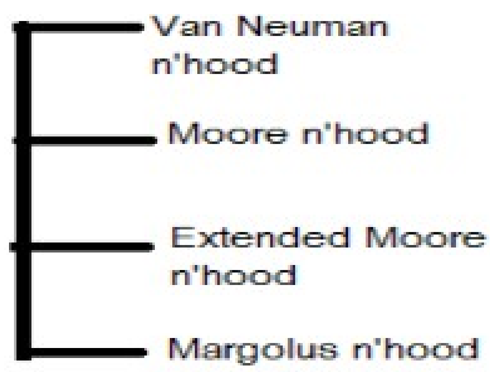
Dimensional cellular automata consist of a row of "cells" and a set of rules. Whereas Ulam's 2 Dimensional cellular automata is a 2 Dimensional space which is divided into cells, much like a matrix. This 2 Dimensional space can be visualized as a kind of grid where each cell is having two states "ON and OFF". Starting from a pattern, a new generation is generated by following some neighbourhood rules. For say, if a cell comes into contact with a "ON or active" cell, it will also become an active cell; on the other hand, if a cell comes in contact with one or no active cell or four or more than four active cells, it will become inactive cell. A set of definite rules determines the value at each site which can be calculated based on the values of the neighbouring cells.

The introduction of the cellular automata by Van Neuman and Ulam was done in order to study some processes like self reproduction. Any system having many discrete elements which are undergoing local deterministic interaction can be seen as cellular automata.

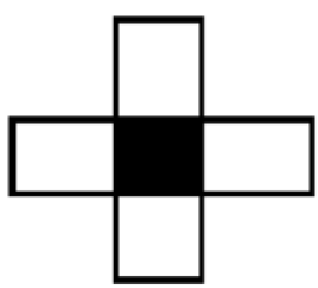
3. Theory

3.1 Types of Neighbourhoods

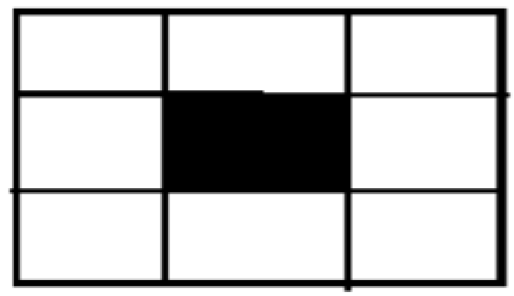
Various types of neighbourhoods are there in cellular automata. Some of them are:



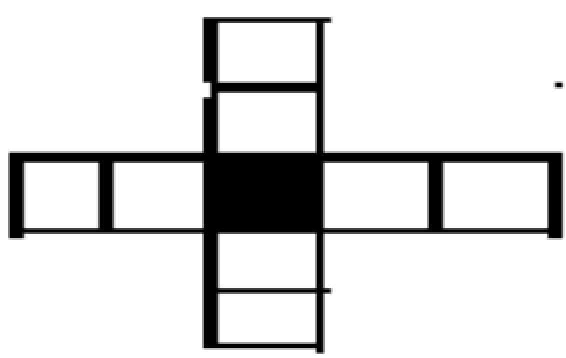
1. In Van Neuman neighbourhood, north, south, east and west neighbourhoods are taken:



2. In Moore neighborhood, along with the Van Neuman neighborhoods, diagonals are also added.



3. In Extended Moore case, the neighbourhood is extended by a distance of one beyond the one.



3.2 PASCAL'S TRIANGLE FORMATION USING CELLULAR AUTOMATA

In mathematics, Pascal's triangle is a triangular array of binomial coefficients. Pascal's triangle can be formed using the addition modulo 2 formula which is the ordinary addition of two numbers where the sum is divided by 2, giving the resultant answer. The triangle thus formed has one major feature that the triangle pattern formed using modulo method inside any triangle looks like a sub triangle. If the Pascal's triangle is extended to infinite rows & every time the picture scale is reduced to half of its size, the resultant pattern looks like a self similar. That is to say, the

the picture or the triangle can be reproduced by taking its sub triangle and then magnifying it.

Let say, a state can be defined as

$$A(n) = \begin{cases} 1, & \text{if } n \text{ is odd} \\ 0, & \text{if } n \text{ is even} \end{cases}$$

Now, applying the addition modulo 2 formula, we can jump to the next state.

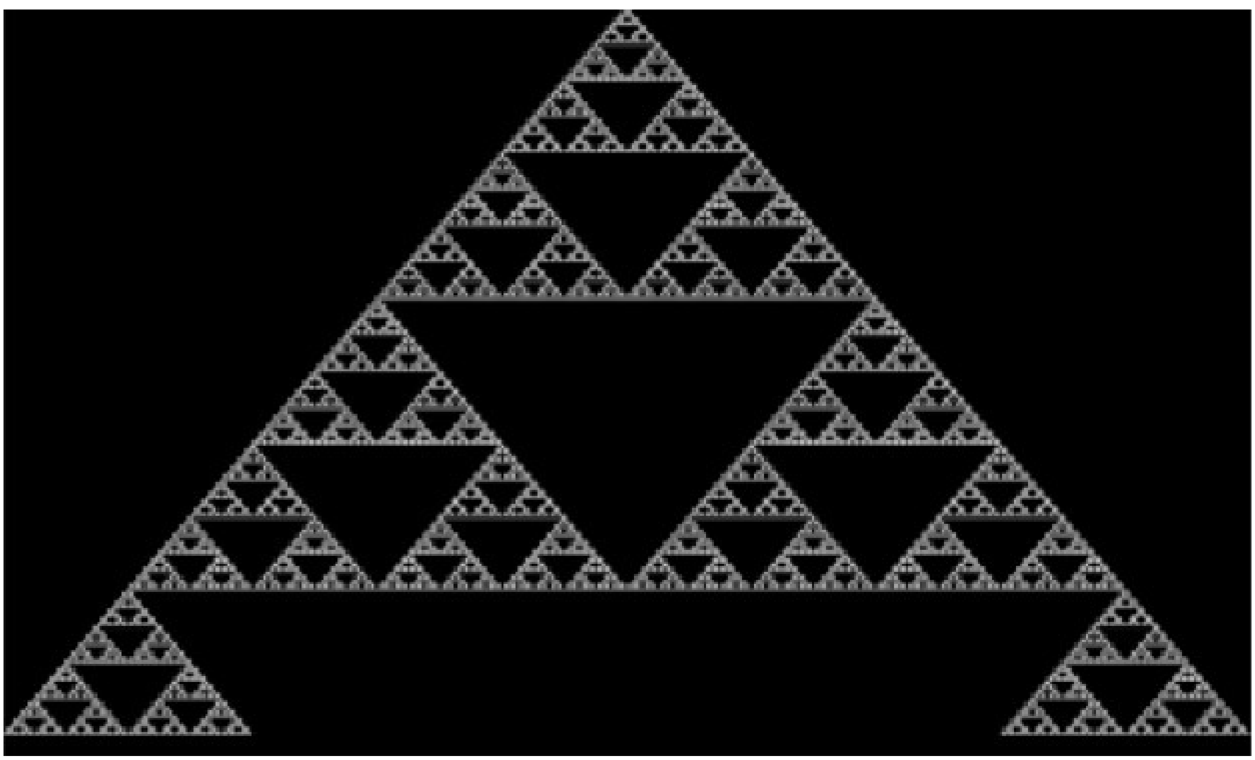
Let, the initial state can be defined as,
A (n) = {1, if n=0; 0 otherwise}

In every other state, we have an entry of 1 on both end and 0 present in between 1. We know that every entry in the triangle is the sum of the adjacent entries which are just lying above, after which the addition modulo 2 is applied on that entry. After doing this, the triangle formation is done using the 1's whose value is the same as that of the Pascal's triangle. The two triangles remain independent of each other till they meet each other in the rth row.

Various values calculated in the Pascal's triangle are

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

The entries in the third row have two copies of the previous row except for the fact that the middle term will be formed when two previous row's 1's overlaid on each other. Therefore the row formed will have 1's on both ends whereas a 2 present in between them.



3.3 Game of Life

The Game of Life was originally presented as a mathematical game.
In Game of Life, all what is needed to do is to give some initial input, an initial configuration is created and this configuration evolves thereafter by its own. It is also known as a "Zero Player Game". Its evolution depends on the initial state. If the game is studied well, it can give us the opportunity to understand the cellular automata better.
Much like a 2 Dimensional cellular space, the Game of Life too made up of a grid of cells much like a matrix.

Certain rules to be followed in the Game of Life are

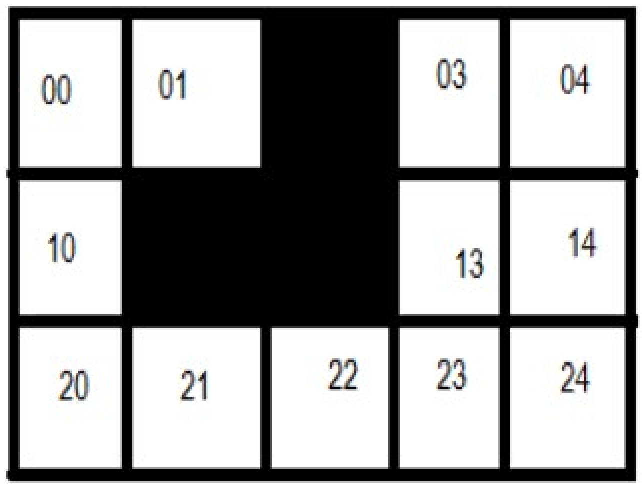
For a populated space

Each cell with one or no active neighbours dies.
Each cell with four or more active neighbours dies.
Each cell with two or three active neighbours survives.

For an unpopulated space

Each cell with three neighbours becomes populated.

For the simplicity, let's take a simple example
Suppose we have a grid of 5X3, and we have given an initial input as follows



For the sake of simplicity, we number the above grid starting from 00 to 24 respectively in three rows. In the above figure, cell number 02, 11, 12 are the active or ON cells.

Now by following the above mentioned rules, we can have the next generation of the cells as

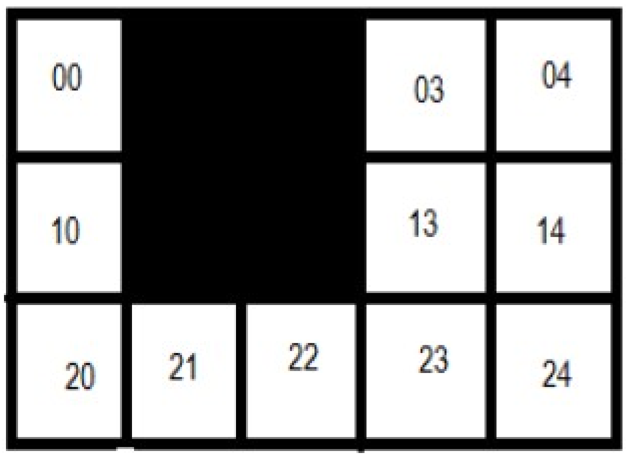
For the populated space

Cell number 02, 11 and 12 has three active neighbours; therefore they will survive

For the unpopulated space

Cell number 00, 10 and 20 has one active neighbour; therefore they will not get populated.

Cell number 03, 13, 23, 22 and 21 has two active neighbours; therefore they will not get populated.



Cell number 01 has three active neighbours; therefore it will also get active in the next generation. In the same way, next generations can be obtained for a number of designs. Some popular designs are:

- Gosper Glider Gun
- Glider
- Small Exploder
- Exploder
- 10 cell row

4. Advancements

In this paper we took the liberty to extend the reach of cellular automata to multiple stages, namely : Alive, Dead, Virus and Mutation.

We have defined each variable and their Grammar below. This grammar acts like the rules and conditions responsible for the life and death of each of the type of cells (stages).

4.1 *Neigbours* **N**

We define variables for Neighbours of cells as the life and death of each cell directly depends on the number of and the type of cells surrounding it.

Thus, total number of Neighbours= **N**

N = NA+NV+NM

where,

NA = number of Alive cells

NV = number of Virus infected cells

NM = number of Mutated cells

4.2 *Condition for Alive Cells*

Alive cell to an alive cell:

Cell should have 2, 3, 4 or 6 neighbors.

N = 2, 3, 4 or 6

Alive cell to a virus cell:

Cell should have 2, 3, 4 or 6 neighbors

and with at least 1 virus.

N = 2, 3, 4 or 6 && **NV** > 1

Else dead.

4.3 *Condition for Mutated cells*

Mutations to mutations

Cell should have either 3 or 4 neighbors.

N = 3 or 4

Mutations to virus

Cell should have exactly four virus neighbors.

N = 4 and all of those are virus

Therefore, **NV** = 4

Else dead.

4.4 Conditions for Virus Cells

Virus cell to virus cell

Cell should have either 3 or 4 neighbors.

N= 3 or 4

Virus cell to mutation

Cell should have either 3 or 4 neighbors with at least 1 mutation

N= 3 or 4 && **NM** > 1

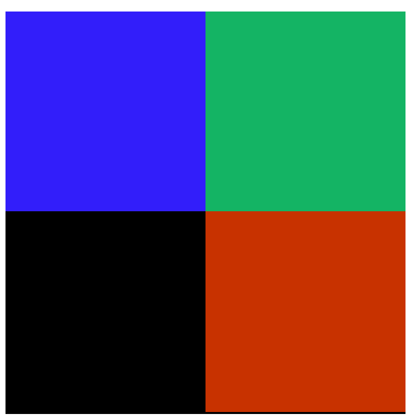


Fig. (Top left: ***Mutation****, top right* ***Alive*** *bottom left:* ***Dead,*** *bottom right:* ***Virus****)*

4.5 Conditions for dead cells

When there are more than 4 neighbors **(N>4)**

50 percent chance that it becomes a mutation

4.5.1 If it has 6 neighbors **(N=6)**

If at least 4 alive **(NA > 4)**

It becomes alive

Else if at least 2 virus **(NV > 1)**

It becomes virus

Else dead.

4.5.2 If it has 4 neighbors **(N=4)**

If exactly 4 virus neighbors or 2 virus and alive neighbors **(NV=4 or (NV=2 and NA= 2))**

Becomes virus

Else if at least 2 mutations **(NM>1)**

Becomes mutation

Else dead.

4.5.3 If it has 3 neighbors **(N=3)**

If all 3 neighbors are mutations

Becomes mutation

else if there are alive and mutation neighbors with at least 1 alive cell **(NA+NM==3 and NM<2)**

Becomes alive

else if there is 1 virus and 2 alive neighbors **(NV=1 and NA =2)**

Becomes virus

4.5.4 If it has 2 neighbors **(N=2)**

If there is 1 mutation and 1 virus neighbor **(NM=1 and NV =1)**

50% probability for it to be a virus or a mutation.

Using these set of rules, we call Grammar, we're able to compute the multiple stages version of **Game of Life.**

5. Cellular Automata Application

Next, we are giving some applications of the cellular automata. These are:

5.1 Cellular automata games

One of the major cellular automata game is the "Game of Life" given by James Conway. Apart from this, there are several other games created using the cellular automata. These games also provide some insights about the synchronization problem for example
firing squadiring mob and queen bee

5.2 Parallel computing machine

The 2 Dimensional cellular automata are being used for image processing and pattern recognition. Toffoli and others developed a machine called CAM (Cellular Automata Machine) which operates in the autonomous mode. A higher order of magnitude at a comparable cost can be achieved using CAM as compared to the conventional computers.

5.3 Cellular Automata for Physical and Biological systems

Cellular automata can be used to model several chemical processes like inter
diffusion of atoms of two materials.

In the biology, cellular automata models are being used for tumour development, developing drug therapy for HIV infections and various other things.
Other areas where cellular automata can be applied are

- VLSI (very large scale implementation) implementations
- Pattern Recognition

6. Future Work and conclusion

In this paper, a sketch of different developments in the field of cellular automata is given. These developments provide a vast field for research in the field of cellular automata. Various tools can be used for the research. Using the theoretical concepts along with the tools, various new things can be done like with help of Cellular automata we can get a good help in 3D technology and 3D techno based games as it helps in the speed of showing graphics of the games as well as in theprocessing speed of the game which will also be fast as comapre to 2 D game. It will also give support to ANN technology because it will help todo parallelprocessing in easy way.