# Cellular Automata on Graph

**Vikrant Singh**

Enrollment No.: 2023ITM011 Date: 06.06.2024

*A progress report submitted in partial fulfillment for the degree of*

**Master of Technology**
**in**
**IT Engineering**

*Under the guidance of*

**Dr. Sukanta Das**
Associate Professor
Department of Information Technology
Indian Institute of Engineering Science and Technology, Shibpur

**Department of Information Technology**
**Indian Institute of Engineering Science and Technology, Shibpur**
**Howrah, West Bengal, India – 711103**
**2022**

**Department of Information Technology**
**Indian Institute of Engineering Science and Technology, Shibpur**
**Howrah, West Bengal, India – 711103**

# CERTIFICATE OF APPROVAL

This is to certify that the work presented in this term paper entitled "Cellular Automata on Graphs", submitted by Vikrant Singh, having the examination roll number 2023ITM011, has been carried out under my supervision for the partial fulfilment of the degree of Master of Technology in Information Technology during the session 2023-25 in the Department of Information Technology, Indian Institute of Engineering Science and Technology, Shibpur.

**(Dr. Sukanta Das)**
Associate Professor
Dept. of Information Technology
Indian Institute of Engineering Science and Technology,
Shibpur, Howrah, West Bengal, India –711103

# Acknowledgement

I would like to take this opportunity to express my sincere thankfulness and deep regard to Dr. Sukanta Das, for his impeccable guidance, valuable feedback and constant encouragement throughout the project. His valuable suggestions were of tremendous help in preparing the term paper. Working under him was an extremely knowledgeable experience for me.

I would like to convey my gratitude to the department of Information Technology, our respected and honourable head of Information Technology, Dr. Prasun Ghosal, and to our Central Library for providing the necessary resources. I would also like to give my sincere gratitude to all my friends for their valuable cooperation. Without their help this study would have been incomplete.

**Dated:**
**Indian Institute of Engineering Science and Technology, Shibpur**
**Howrah, West Bengal, India**

..........................
**(Vikrant Singh)**
**Enrollment No.: 2023ITM011**
**Date: 06.06.2024**

# Abstract

A general property of Cellular Automata is reversibility. Even though the reversibility of traditional CAs has already been studied. In this report we discuss about the Cellular Automaton, types of Cellular Automaton and an introduction to ECA and 2D-CA.

Later in the report the implementation of CA on Ceyley Tree. Some definitions and useful equation are used and finally we classify Cayley Tree with an example.

# Contents

# Chapter 1

# Introduction

Cellular automata are mathematical models that simulate the dynamic behavior of complex systems. They consist of a grid of cells, each with a set of rules that determine how it changes state over time based on the states of its neighboring cells.

The marvels of nature have always fascinated the scientific world, with its unpredictable dynamics and intricate interplay of organisms. Traditional computational models, such as the Turing Machine and von Neumann's computer design, relied on centralized control. However, observations of natural phenomena, like the patterns in snowflakes or the behavior of ants, suggest a decentralized form of control where each unit operates independently.

In the early 20th century, Network Science emerged to study these individualistic and parallel behaviors. This led to the development of various models, many inspired by biological systems, that support decentralized computing. A notable breakthrough in this field was the creation of cellular automata.

The concept of decentralization gained prominence with the advent of widely used distributed systems like Ethernet. The rise of the Internet, a distributed system itself, further reinforced this idea, leading to its application in numerous fields. Thus, the shift from centralized to decentralized systems represents a significant evolution in computational models, closely mirroring patterns observed in nature. A distributed system is composed of interconnected, yet autonomous, computer components that communicate to synchronize their operations. Viewed from a process perspective, it's a network of geographically dispersed processes that interact solely through message exchange. In such a system, no single entity oversees or controls the computation. Each component and process in a distributed computation can be identified by unique identifiers. However, the need for a central organization to assign these identifiers contradicts the core principles of distributed control, implying that a truly distributed system should be anonymous. Numerous formal models have been developed for distributed systems, offering valuable insights. Cellular automata, due to their inherent parallelism, are often considered a natural fit for distributed computing frameworks. In the 1950s, Jon von Neumann introduced the concept of self-reproducing automata, later known as Cellular Automata. He explored the idea of computational universality, where a machine can simulate any other machine. However, he clarified that a universal constructor, a machine capable of emulating other machines, doesn't necessarily need to be a universal computer.

The study of biological phenomena has been a focal point across various fields. Christopher Langton pioneered the field of "Artificial Life", using Cellular Automata as a foundation. Some Cellular Automata exhibit biological characteristics like self-replication and self-organization. Conway's "Game of Life" is a notable example of such Cellular Automata.

The global scientific community is striving to create intelligent machines. A living system is often considered an intelligent machine. The Turing Test, developed by Alan Turing in 1950, is a key tool to assess a machine's intelligence. However, this test focuses on functional intelligence, not intrinsic intelligence. The Chinese Room Argument, which differentiates between true intelligence and simulated intelligence, led to the development of the strong AI paradigm.

The term "intelligent machine" is often used to describe a machine that exhibits properties similar to a living system. Defining "intelligence" is crucial to answer the question, "How intelligent has a machine become?". It's a challenging task to define intelligence in a way that closely aligns with its typical usage and provides a meaningful answer to the question of a machine's intelligence.

## 1.1 Organization of the report

In this section, we provide an organization of the report along with a summary of each chapter. The contribution of the report to the aforementioned topic is divided. Before going to discuss the different chapters, the introduction of the CAs is provided in Chapter 2.

- **Chapter 2.** This chapter describes the principle of CAs, neighborhood and types of neighborhood, different forms of CAs including ECA(Elementary Cellular Automata), and a well known example of 2-D CAs also referred to as Game of Life.
- **Chapter 3.** This chapter addresses an introduction to the Cayley Tree. After this the chapter introduces CA on Cayley Tree of order 2 and ends up with an example.
- **Chapter 4.** With this chapter this report ends after having discussion about the future scope of CA on Ceyley Tree.

# Chapter 2

# Survey on Cellular Automata

A Cellular Automaton (CA) is a network of cells, each being a finite automaton with a finite state set, S. The cells change their states over time based on the states of their neighboring cells, using a local rule or next-state function. The collective state of all cells at any given time is called the CA's configuration, and it changes as the CA evolves. In essence, a CA is a dynamic system where each cell's state is updated based on its neighbors.

**Definition 1** *A cellular automaton is a quadruple ($\mathcal{L}$, $\mathcal{S}$, $\mathcal{N}$, $\mathcal{R}$) where,*

- $\mathcal{L} \subseteq \mathbb{Z}^{\mathcal{D}}$ *is the $\mathcal{D}-$dimensional cellular space. A set of cells are placed in the locations of $\mathcal{L}$ to form a regular network.*
- $\mathcal{S}$ *is the finite set of states; e.g. $\mathcal{S} = \{0, 1, \cdots, d-1\}$.*
- $\mathcal{N} = (\vec{v_1}, \vec{v_2}, \cdots, \vec{v_n})$ *is the neighborhood vector of $n$ distinct elements of $\mathcal{L}$ which associates one cell to it's neighbors.*
- *The local rule of the CA is $\mathcal{R} : \mathcal{S}^n \rightarrow \mathcal{S}$. A cell's subsequent state is determined by the expression $f(s_1, s_2, s_3, \cdots, s_n)$ where $s_1, s_2, s_3, \cdots, s_n$ denotes the states of it's $n$ neighbors.*

Cellular automata can be of many types. One of a cellular automaton's most fundamental properties is the kind of grid that it evolves on. The most fundamental grids of this kind are one-dimensional arrays (1D CA). In case of two dimensional CA, grids of square can be used.

## 2.1 Neighbor in CA

In Cellular Automata (CA), the concept of "neighboring" refers to the cells that are adjacent to a given cell and whose states can influence the state of that cell. The type of neighborhood depends on the geometry and

dimensionality of the CA. Here are some common types of neighborhood for CA:
1. **1-D neighborhood**: In a 1D (one-dimensional) Cellular Automaton, the neighborhood of a cell usually includes the cell itself and its immediate neighbors on either side. This is often referred to as a "radius 1" neighborhood in 1D CA. So, for a cell at position i, its neighborhood would typically be the cells at positions i-1, i (the cell itself), and i+1. The exact definition of a neighborhood can vary depending on the specific rules of the Cellular Automaton.
2. **Von Neumann Neighborhood**: In a 2D grid, a cell's neighbors are the cells that are horizontally or vertically adjacent to it. This results in 4 neighbors for each cell (except for edge cells).
3. **Moore Neighborhood**: In a 2D grid, a cell's neighbors are the eight cells that surround it horizontally, vertically, and diagonally.
The choice of neighborhood can significantly impact the behavior of the CA. Different rules might use different types of neighborhoods. The local rule or next-state function of the CA, which determines how a cell updates its state, takes the states of the cell's neighbors as input. Therefore, the definition of "neighbor" is crucial to the evolution of the CA.

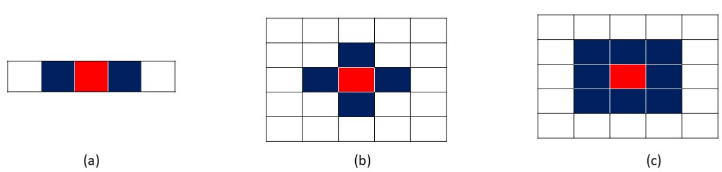

Figure: (a) 1-D Neighborhood (b) Von Neumann Neighborhood (c) Moore neighborhood

## 2.2 Types of CA

1. **Elementary Cellular Automata (ECA)**: These are the simplest class of one-dimensional cellular automata. They are composed of cells arranged in a line or an 1-D array, and each cell can be in one of two possible states, typically labeled 0 and 1.

   The state of a cell in the next generation is determined by its current state and the states of its immediate neighbors on either side. This

**Elementary Cellular Automata**

| | 111 | 110 | 101 | 100 | 011 | 010 | 001 | 000 | RULE |
|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | **30** |
| $f$ | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | **90** |
| | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | **160** |

**Figure 2.1:** ECA with rules - 30, 90, 160

rule to determine the state of a cell is often referred to as the local rule or next-state function.

In ECA, the local rule depends only on the current state of the cell and its two immediate neighbors. These three values can be encoded with three bits. ECA are commonly used in mathematics and computability theory due to their simplicity and the complex patterns they can generate. Despite their simplicity, ECA can model complex systems and have been used in various fields such as physics, biology, and computer science.

2. **2-D(two-dimensional) Cellular Automaton (CA)**: This is a discrete model of computation studied in automata theory. It consists of a regular grid of cells, each in one of a finite number of states. The grid can be in any finite number of dimensions, but in a 2D CA, it's typically a square or rectangular grid.

   Each cell in a 2D CA has a neighborhood, which is the set of surrounding cells that influence its state. The most common types of neighborhoods in 2D CA are the Von Neumann and Moore neighborhoods. In the Von Neumann neighborhood, a cell's neighbors are the cells that are horizontally or vertically adjacent to it. In the Moore neighborhood, a cell's neighbors are the eight cells that surround it horizontally, vertically, and diagonally.

   The state of each cell in a 2D CA is updated according to a fixed rule (generally, a mathematical function) that determines the new state of the cell based on the current state of the cell and the states of the cells in its neighborhood. This rule is applied to the whole grid simultaneously.

2D CAs can produce complex patterns and behaviors from simple rules, and they have found applications in various areas, including physics, theoretical biology, microstructure modeling, computer science, coding, communication, generative art, and music. A famous example of a 2D CA is Conway's Game of Life.

### 2.2.1 Game of Life

Conway's Game of Life, also known simply as Life, is a two-dimensional cellular automaton devised by the British mathematician John Horton Conway in 1970. It's a zero-player game, meaning its evolution is determined by its initial state, requiring no further input.

The universe of the Game of Life is an infinite, two-dimensional orthogonal grid of square cells, each of which is in one of two possible states, live or dead. Every cell interacts with its eight neighbors, which are the cells that are horizontally, vertically, or diagonally adjacent. At each step in time, the following transitions occur:

- Any live cell with fewer than two live neighbors dies, as if by underpopulation.
- Any live cell with two or three live neighbors lives on to the next generation.
- Any live cell with more than three live neighbors dies, as if by overpopulation.
- Any dead cell with exactly three live neighbors becomes a live cell, as if by reproduction.

The initial pattern constitutes the seed of the system. The first generation is created by applying the above rules simultaneously to every cell in the seed. Each generation is a pure function of the preceding one. The rules continue to be applied repeatedly to create further generations.

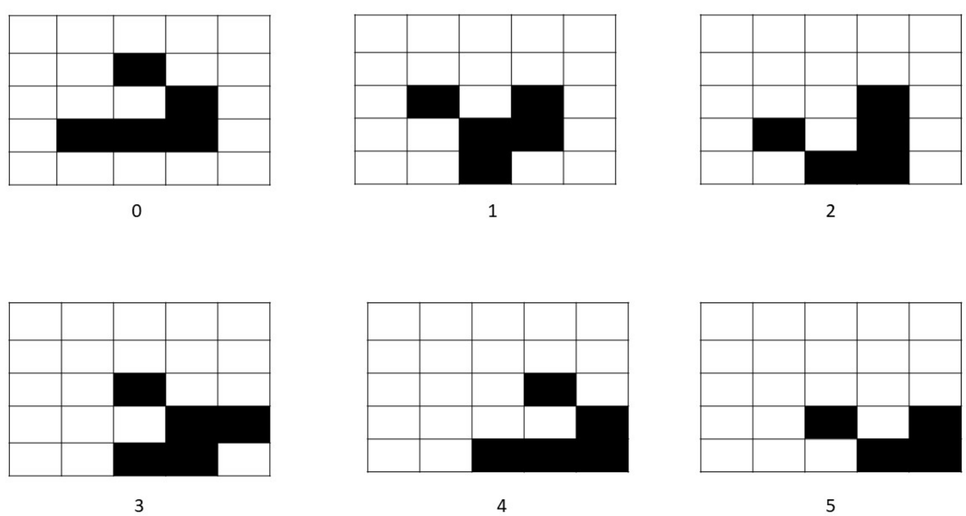


**Figure 2.2:** Game of Life from t = 0 to t = 5

# Chapter 3

# CA on Cayley Tree

## 3.1 Cayley Tree

A Cayley Tree is simple connected undirected graphs G = (V, E) where V is the set of vertices, E is the set of edges. It is also referred as prototypes of graphs containing no cycles (a cycle is a closed path of different edges), i.e., they are trees. The Cayley tree $T^k$ of order $k >= 1$ is an infinite tree, i.e., a graph without cycles, from each vertex of which exactly k+1 edges issue.
A Cayley tree, also known as a Bethe lattice, is a tree where each vertex has the same number of neighbors. The number of vertices in a Cayley tree depends on the degree of each vertex (the number of edges connected to it) and the number of generations or levels in the tree.

Let's denote:

- 'd' as the degree of each vertex (the number of edges connected to it)
- 'n' as the number of generations or levels in the tree

The number of vertices 'V' in a Cayley tree can be calculated using the formula:

$$V = 1 + d \sum_{i=0}^{n-1} (d-1)^i$$

Here's how it works: - The first term '1' represents the root of the tree. - The second term represents the remaining vertices. For each level 'i' of the tree (starting from '0'), there are $d * (d-1)^i$ vertices.

## 3.2 Finite Cellular Automata on Cayley Tree

A Cayley Tree $T^k$ = (V, L, i) where V is the set of vertices of $T^k$, L is the set of edges of $T^k$ and i is the incidence function associating each edge. i(l) = x, y, $l \in L$ ; then x and y are called the nearest neighboring vertices and we write l =< x, y >. A configuration on V is defined as a function $x \in V \rightarrow \sigma(x) \in \mathrm{Z}_p where \mathrm{Z}_p$ = {0, 1, . . . , (p-1)} and ($p >= 2$) be the field of the prime numbers modulo p.

**Example:** Cayley Tree of $order = 2$ and $level = 3$

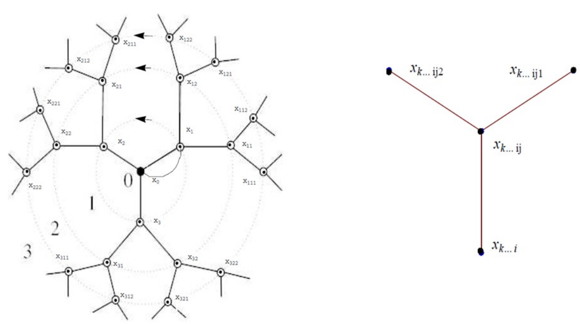


**Figure 3.1:** (a) Cayley tree of order two with levels 3, (b) Elements of the nearest neighborhoods surround the center $x_{k\ldots ij}$ , $k = 1, 2, 3$ and $i, j = 1, 2$.

In the Fig.3.1, we show Cayley tree of order two with levels 3 and the nearest neighborhood which comprises three cells which surround the center cell $x_{k\ldots ij}$ .

# Chapter 4

# Discussion and Future Scope

The future scope of cellular automata on Cayley trees is quite promising and spans across various fields:

1. **Topological Properties**: Research has been conducted to investigate the topological properties of cellular automata on trees, including permutivity, surjectivity, preinjectivity, right-closingness, and openness. These studies provide a deeper understanding of the behavior of cellular automata on Cayley trees.

2. **Reversibility**: Studies have been conducted on the reversibility of linear cellular automata defined on Cayley trees. This research can provide insights into the behavior of cellular automata under different conditions and can be beneficial in fields like cryptography and image processing.

3. **Entropy Analysis**: The measure-theoretical entropy of cellular automata on Cayley trees has been studied. Understanding the entropy of these systems can provide insights into their information dynamics and complexity.

# References


- Bhattacharjee, K., Naskar, N., Roy, S., Das, S. (2020). A survey of cellular automata: types, dynamics, non-uniformity and applications.Natural Computing,19, 433-461.
- Conway's Game of Life - Cellular Automata by PtitPoly (itch.io)
- Conway's Game of Life (conwaylife.com)
- Dalai, A., Paul, S. (2023, March). Layered Cellular Automata and Pattern Classification. In Asian Symposium on Cellular Automata Technology(pp. 127-141). Singapore: Springer Nature Singapore.
- AKIN, H. (2012). Cellular automata on cayley tree. arXiv preprint arXiv:1211.7362.
- simple but Stunning: Animated Cellular Automata in Python | iSquared