

Cellular Automata

Diego Castillo Reyes^a, Marthon Leobardo Yañez Martinez^a, Aldo Escamilla Resendiz^a and Muñoz González Eduardo^a

^aResearcher

Dra. Miriam Pescador Rojas

Abstract—Cellular automata are a mathematical and computational model used to simulate dynamic systems. This work presents a review of cellular automata, their history, classification, and applications. Additionally, an example of one-dimensional and two-dimensional cellular automata is shown.

Keywords—Automata, Cellular, Genetic, Algorithms, Simulation

Contents

1	Introduction	1
2	Background	1
2.1	Motivation	1
2.2	Developments	1
2.3	Applications	1
3	Cellular Automata Algorithm	2
4	Parameters Required on a Cellular Automata	2
5	Versions of Cellular Automata	2
6	Analogy with the Nature	2
7	Implementation Repositories	2
8	Usage Examples	2

1. Introduction

Cellular automata (CA) are a mathematical and computational model used to simulate dynamic systems. They are composed of a grid of cells, each of which can be in a finite number of states. The state of each cell is updated based on a set of rules that define the behavior of the system. CA are used in various fields, such as physics, biology, and computer science, to model complex systems and study their behavior. This work presents a review of cellular automata, their history, classification, and applications. Additionally, an example of one-dimensional and two-dimensional cellular automata is shown.

2. Background

The concept of cellular automata was invented by Stanislaw Ulam and John Von Neumann in the 1940s while they were working at the Los Alamos National Laboratory. The work on cellular automata began in the 1940s, with significant developments occurring throughout that decade. Von Neumann’s comprehensive work on self-replicating automata was published posthumously in 1966 in the book “Theory of Self-Reproducing Automata,” edited by Arthur W. Burks.

2.1. Motivation

The primary motivation behind cellular automata was to understand and model complex systems using simple, local rules. This idea was rooted in the study of biological processes and the desire to create self-replicating machines.

2.2. Developments

- Conway’s Game of Life (1970): British mathematician John Conway popularized cellular automata with his “Game of Life”, a bidimensional binary cellular automaton. This game demonstrated how simple rules could lead to complex emergent behavior, sparking widespread interest and research in cellular automata.

- Stephen Wolfram’s work (1980s): Wolfram conducted extensive research on cellular automata, classifying them into four types based on their behavior and demonstrating their potential as models of natural processes and as computational systems.

Fig. 2 An example of Conway’s Game of Life.

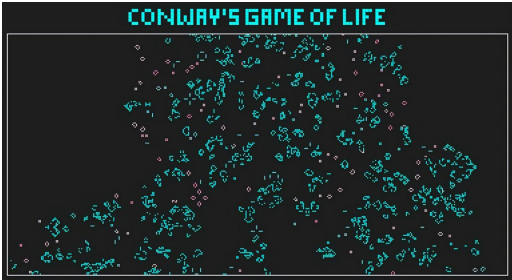


Figure 1. Conway’s Game of Life.

2.3. Applications

- Cellular automata have been used in various fields, including:
- Computer Science: Parallel computation, cryptography, and image processing.
 - Physics: Modeling physical systems, such as fluid dynamics and crystal growth.
 - Biology: Simulating biological processes, such as population dynamics and pattern formation.

Fig. 2 Application of cellular automata on Computer Science.

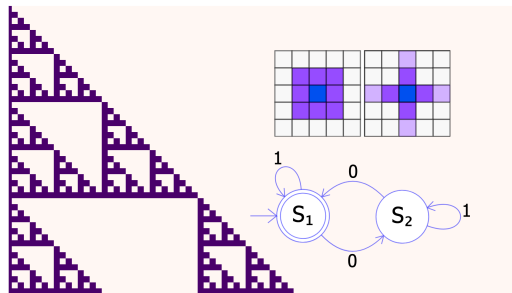


Figure 2. Example of cellular automata.

3. Cellular Automata Algorithm

Algorithm 1: Basic Cellular Automaton

Input: *gridWidth*: Width of the grid, *gridHeight*: Height of the grid, *states*: Set of possible states for the cells, *neighborhood*: Set of relative positions defining the neighborhood of each cell, *rules*: Set of state transition rules, *maxTimeSteps*: Maximum number of time steps

Output: The final state of the grid

```

1 Initialize gridHeight × gridWidth, set the initial states on
  the grid and create newGrid as a copy of the grid.;
2 while i < maxTimeSteps do
3   for x in gridWidth do
4     for y in gridHeight do
5       neighbors = getNeighbors(grid, neighborhood,
6         x, y);
7       newGrid[x][y] = applyRules(grid[x][y],
8         neighbors, rules);
9   Display the state of newGrid grid = newGrid;
10  i++;
  
```

4. Parameters Required on a Cellular Automata

5. Versions of Cellular Automata

6. Analogy with the Nature

7. Implementation Repositories

8. Usage Examples