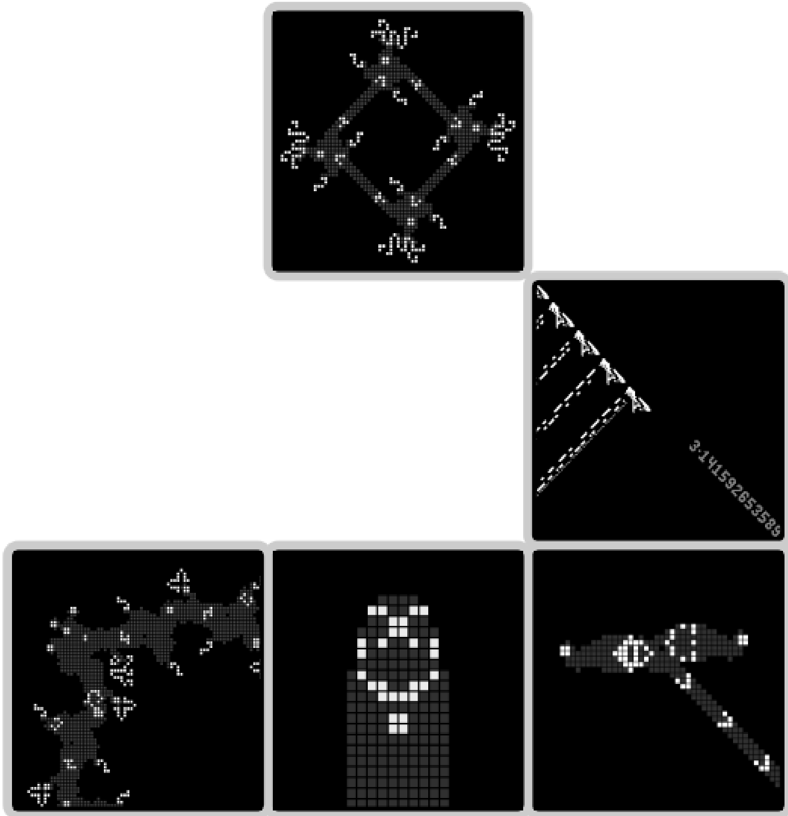


Conway's Game of Life

June 2023

Prof. Larijani

Shadmehr Salehi

Nazanin Zarei

Foozhan Fahimzade

Introduction

Conway's Game of Life is a cellular automaton that is played on a 2D square grid. Each square (or "cell") on the grid can be either alive or dead, and they evolve according to the following rules:

- Any live cell with fewer than two live neighbors dies (referred to as underpopulation)
- Any live cell with more than three live neighbors dies (referred to as overpopulation)
- Any live cell with two or three live neighbors lives unchanged to the next generation.
- Any dead cell with exactly three live neighbors comes to life.

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 — births and deaths happen simultaneously, and the discrete moment at which this happens is sometimes called a tick. (In other words, each generation is a pure function of the one before.) The rules continue to be applied repeatedly to create further generations.

Origin

Conway was interested in a problem presented in the 1940s by renowned mathematician John von Neumann, who tried to find a hypothetical machine that could build copies of itself and succeeded when he found a mathematical model for such a machine with very complicated rules on a rectangular grid.

The Game of Life emerged as Conway's successful attempt to simplify von Neumann's ideas.

The game made its first public appearance in the October 1970 issue of Scientific American, in Martin Gardner's "Mathematical Games" column, under the title of The fantastic combinations of John Conway's new solitaire game "life".

From a theoretical point of view, it is interesting because it has the power of a universal Turing machine: that is, anything that can be computed algorithmically can be computed within Conway's Game of Life.

What Are Cellular Automata ?

Cellular automata are a class of mathematical models that describe the behavior of systems composed of discrete, locally interacting elements. The elements are often represented as cells on a regular lattice, such as a two-dimensional grid, and the state of each cell is updated according to a set of rules that depend on the state of its neighbors. Cellular automata have been used to model a wide variety of phenomena, including physical systems, biological systems, and social systems.

The Rules

Like Chess and Go, Life is played with pieces on a board. But unlike Chess and Go, it requires no players. A “zero-player game” with no winners or losers, which result is fully determined by the initial configuration of the pieces on the board.

A player is only needed to advance the state of the game to the next turn—a “generation”—following three simple rules.

1. Survival. Every piece surrounded by two or three other pieces survives for the next turn.

2. Death. Each piece surrounded by four or more pieces dies from overpopulation. Likewise, every piece next to one or no pieces at all dies from isolation.

3. Birth. Each square adjacent to exactly three pieces gives birth to a new piece.

Patterns

Many different types of patterns occur in the Game of Life.

One of the most interesting aspects of the Game of Life is the rich variety of patterns that can arise from simple initial configurations. Patterns in the Game of Life can be classified into three categories: still lifes, oscillators, and spaceships.

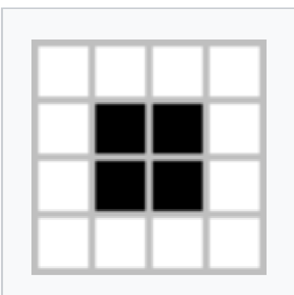
Still lifes are patterns that do not change over time. They consist of a set of live cells that are arranged in a stable configuration. Examples of still lifes include the block, beehive, and loaf.

Oscillators are patterns that repeat themselves periodically. They consist of a set of live cells that oscillate between two or more configurations. Examples of oscillators include the blinker, toad, and beacon.

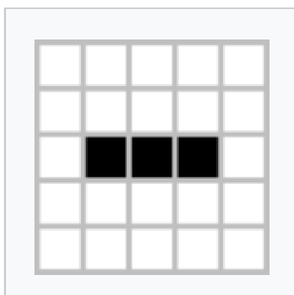
Spaceships are patterns that move across the grid at a constant speed. They consist of a set of live cells that move in a specific direction, while maintaining a stable shape. Examples of spaceships include the glider, lightweight spaceship, and middleweight spaceship.

Many interesting patterns have been discovered in the Game of Life, including patterns that exhibit complex behavior, such as the Gosper glider gun, which generates a stream of gliders that move across the grid indefinitely.

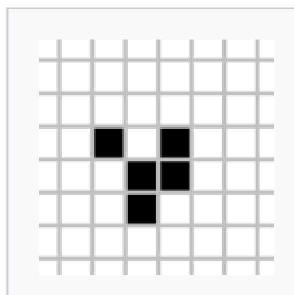
Common examples of three classes are shown below, with live cells shown in black, and dead cells shown in white.



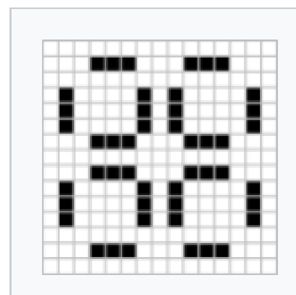
Block (still life)



Blinker
period-2 oscillator



Glider (spaceship)



Pulsar
period-3 oscillator

Iteration

From a random initial pattern of living cells on the grid, observers will find the population constantly changing as the generations tick by.

The patterns that emerge from the simple rules may be considered a form of beauty. Small isolated subpatterns with no initial symmetry tend to become symmetrical. Once this happens the symmetry may increase in richness, but it cannot be lost unless a nearby subpattern comes close enough to disturb it.

In a very few cases the society eventually dies out, with all living cells vanishing, though this may not happen for a great many generations. Most initial patterns eventually "burn out", producing either stable figures or patterns that oscillate forever between two or more states (known as ash); many also produce one or more gliders or spaceships that travel indefinitely away from the initial location.

Classification

After its first publication in Scientific American, Life got so popular among Mathematicians that a quarterly newsletter called “LIFELINE” started appearing.

It is in there that its editor, Robert Wainwright, published a system to classify the many objects—patterns, as they’re called—that he saw appearing in the game.

Characteristics				Class	Example
Stable	Inactive			**Class I** Still Lifes	block
	Active	Stationary		**Class II** Oscillators	blinker
		Moving	Constant cells	**Class III** Spaceships	glider
			Increasing cells	**Class IV** Guns	glider gun
Unstable	Predictable			**Class V**	n-ominoes
	Unpredictable			**Class IV**	?

Class I are the so-called “still lifes”: patterns that do not change over time.

Class II are called “oscillators”, and they repeat over a certain number of generations. They are classified based on their period.

The blinker, for instance, repeats after two generations; hence it has period two.

Many believe that oscillators of any period can be constructed in Life. And indeed, finite oscillators are known to exist for all periods …except 19, 38 and 41.

Class III groups some of the most studied patterns: spaceships. Those are oscillators that, at the end of their cycle, somehow find themselves in a different position. They effectively …move! The most well known and loved is, without any doubt, the glider.

Discovered in 1969 by the British Mathematician Richard Kenneth Guy, it was named by John Conway himself, due to a property it exhibits called glide symmetry.

Gliders are the smallest spaceship known to exist. And yet, they play a fundamental role for all Mathematicians and Computer Scientists interested in studying Life from a more “academic” point of view.

A major breakthrough occurred in 1970, when Conway himself offered $50 to the first who could find a configuration which grew indefinitely.

The American Mathematician and Programmer Bill Gosper responded with what is now known as the Gosper glider gun. An oscillator that, every 30 generations, spawns a new glider. That was the first **Class IV** object to be discovered.

Class V patterns behave seemingly erratically—chaotic, as we would say today—until they eventually collapse to one of the aforementioned classes.

But some patterns are doomed to a different fate: remaining in a perpetual state of chaos—forever evolving, yet never stabilizing onto something predictable. This is the mysterious, elusive **Class VI.**

Life is and remains a fully deterministic game, with clear rules and no randomness of any kind. Yet, generally speaking, the fate of a pattern cannot be predicted without simulating it directly. One could simply wait until a pattern eventually falls into a stable configuration. But if it does not—if it truly belongs to **Class VI**—then you would wait forever for an answer which would never come.

The Game of Life is ultimately undecidable: there are many patterns whose fate is easy to predict, but in general, this cannot be done for an arbitrary pattern.

That may sound like a bold statement, but if we want to understand why, we need to go deeper. Hold on tight to your glider, because we are about to build a computer in Life.

Computability of the Game of Life

The question of whether the Game of Life is computable or not is an interesting one. A system is said to be computable if there exists an algorithm that can compute its behavior for any given input. In the case of the Game of Life, the input is the initial state of the grid, and the behavior is the evolution of the grid over time according to the rules of the game.It turns out that the Game of Life is computable, in the sense that there exists an algorithm that can simulate the behavior of the game for any given input. This algorithm is known as a universal Turing machine, which is a theoretical construct that can simulate the behavior of any other Turing machine. Since the Game of Life can be implemented as a Turing machine, it follows that the Game of Life is computable.

Decidability of the Game of Life

Many patterns in the Game of Life eventually become a combination of still lifes, oscillators, and spaceships; other patterns may be called chaotic. A pattern may stay chaotic for a very long time until it eventually settles to such a combination.

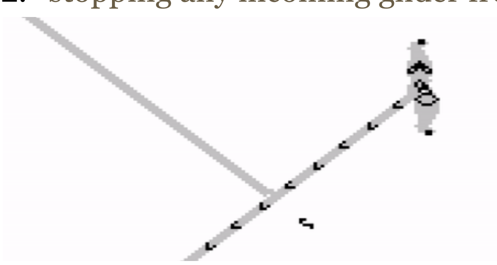
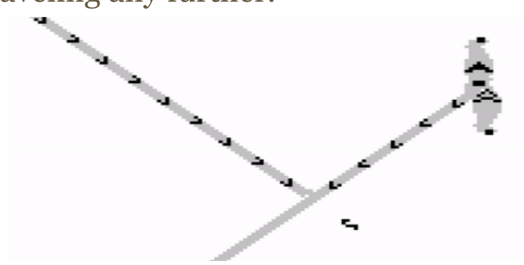
The Game of Life is undecidable, which means that given an initial pattern and a later pattern, no algorithm exists that can tell whether the later pattern is ever going to appear. Given that the Game of Life is Turing-complete, this is a corollary of the halting problem: the problem of determining whether a given program will finish running or continue to run forever from an initial input.

Turing Completeness

It turns out the Game of Life is Turing complete, meaning it is also capable of universal computation. Gliders are key to this. In general, if the behavior of cells would be either repetitive (still life or oscillators cycle through 1 or more patterns) or chaotic, it would be hard to perform any computation. But gliders move and can interact with each other, thus enabling some non-chaotic processes.Now we will see how we can create logical gates with a combination of patterns.

- **NOT Gate**

  The simplest logic gate we can construct is a NOT gate. It takes a signal and it inverts its state. In Life, this means constructing a pattern that will do two things:
  1. producing gliders when no gliders are received, and
  2. stopping any incoming glider from traveling any further.

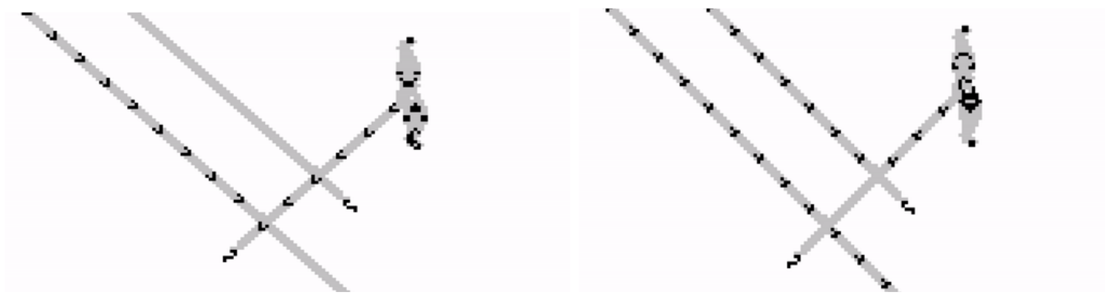



- **AND Gate**

  An AND Gate takes two inputs—hence two streams—and produces a new glider only when it receives two at the same time. We can construct such a pattern by modifying and extending an existing NOT Gate.

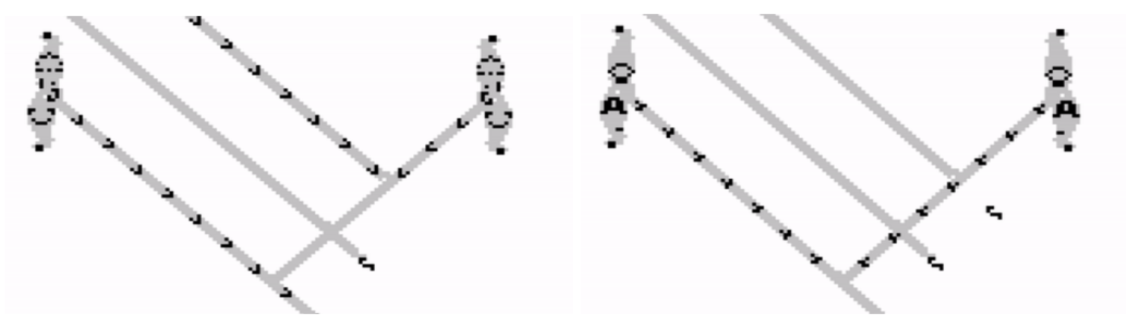
  The gun on the right is placed in such a way that its glider stream will annihilate any glider coming from A.

  In order for the signal A to survive, the gliders from B must block the incoming stream.



- **OR Gate**

  We will construct an OR gate. As the name suggests, it produces a glider when it receives at least one from its two input streams.



But perhaps one of the most intriguing aspects of being Turing complete, is not really the fact that you could build a computer in Life. It is that you can simulate Life within Life itself. As shown in this video.

The Role of Gates & The Power of Cellular Automata

What makes the NOT and OR gates so “fundamental” is that they are a functionally complete set of logic gates. It means that they can be chained together to compute the result of any arbitrarily complex binary expression.

With these gates above we can make latches to store binary data Thus we really do have everything we need to build a computer.

We again encountered a system capable of computing anything computable, based only on a matrix of cells and a couple of rules (live cells with 2 or 3 neighbors stay alive, dead cells with exactly 3 neighbors become live).

Conclusion

Conway’s Game of Life is undoubtedly one of the most successful examples of cellular automaton ever discovered. Some might undoubtedly struggle to understand how a simulation that requires—well—no players could even be described as a “game”. The reality is that Life, pretty much like any other product worth of the title, is a game because for decades it has been capturing the attention of millions of people. And fifty years after its original publication, thousands are still not just playing it, but conducting actual research on it. As it turns out, Life is more than just a game.

Conway’s achievement was not just discovering what is possibly the most interesting cellular automaton, but also to make this entirely new field appealing to a much larger audience.

Conway's Game of Life is a classic example of a two-dimensional cellular automaton that has attracted a great deal of interest from mathematicians, computer scientists, and enthusiasts alike. The game is played on a two-dimensional grid of cells, and the state of each cell at time t+1 depends on the state of its eight neighboring cells at time t, according to a set of simple rules. The game has been studied extensively, and many interesting patterns and behaviors have been discovered. While the Game of Life is computable, the question of whether it is decidable or not is an open problem. Nevertheless, the Game of Life has found applications in a variety of fields, and it continues to be a source of fascination and inspiration for researchers and enthusiasts alike.

One of the fascinating aspects of the Game of Life is the way in which simple rules can give rise to complex behavior. The four rules that govern the behavior of the game are very simple, and yet they can lead to a wide variety of patterns and behaviors, including stable structures, oscillations, and movement.

One of the reasons for the popularity of the Game of Life is the fact that it is easy to implement and experiment with. The game can be played on a computer screen using software programs, or it can be implemented using physical devices, such as electronic circuits or mechanical devices. This makes the Game of Life accessible to people with different backgrounds and interests, from mathematicians and computer scientists to hobbyists and enthusiasts.

Another interesting aspect of the Game of Life is its relationship to other areas of mathematics and science. The game can be viewed as a type of dynamical system, in which the behavior of the system evolves over time according to a set of rules. This makes the Game of Life relevant to the study of dynamical systems in mathematics and physics. The game can also be viewed as a type of cellular automaton, which makes it relevant to the study of complex systems in computer science and other fields.

The study of the Game of Life has led to many interesting discoveries and insights. For example, researchers have discovered a wide variety of patterns and behaviors in the game, including gliders, which are patterns that move across the grid at a constant speed; puffers, which are patterns that emit other patterns as they move; and guns, which are patterns that produce other patterns in a repeating pattern. These patterns have been studied extensively, and many interesting properties have been discovered, such as the fact that gliders can be used to construct complex structures and circuits.

The question of whether the Game of Life is decidable or not is an important one, as it has implications for the study of computability and complexity. The fact that the behavior of the game is highly unpredictable suggests that the problem of determining whether a given configuration will eventually reach a stable state may be undecidable. This has led to a number of interesting research questions and challenges, such as the development of algorithms and techniques for analyzing the behavior of the game.

Overall, the Game of Life is a fascinating example of a simple system that exhibits complex behavior. It has attracted the interest of researchers and enthusiasts from a wide range of fields, and it continues to inspire new ideas and insights. Whether it is viewed as a mathematical curiosity, a tool for exploring complex systems, or a source of inspiration for creativity and innovation, the Game of Life remains a fascinating and intriguing topic of study.

Implementation

A python implementation of the game of life can be found at GitHub

References

- Conway's Game of Life
- Conway's Game of Life - Wikipedia
- Conway's Game of Life - Alan Zucconi
- Let’s BUILD a COMPUTER in CONWAY's GAME of LIFE
- John Conway’s ‘Game of Life’ and How Complex Systems Can Arise From Simple Rules | by Sunny Labh | Cantor’s Paradise
- Computability Part 4: Conway's Game of Life