An Implementation and Analysis of Conway's Game of Life (and Other Cellular Automata)

Kieran Cheung (SID: 201324332)
Daniel Jennings (SID: 201428817)
Scott Wilkie (SID: 201392537)

Contents

Introduction

Conway's Game of Life

The Game of Life is a game that takes place on a two-dimensional grid in which each grid 'cell' has two states: 'alive' or 'dead' [1]. Each cell in the grid determines its next state by following a simple set of rules, based on its current state and the state of cells within its radius=1 Moore neighbourhood. Figure 1 describes the evaluation performed by a cell during each generation of the game, to determine the value of its next state.

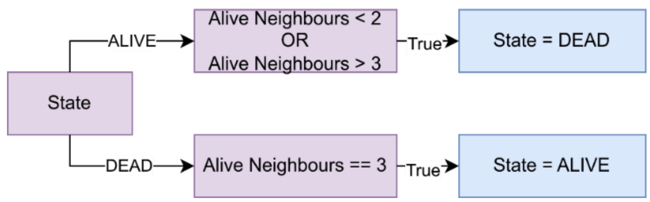

Figure 1 - Flowchart showing how each cell in the Game of Life evaluates its next state.

The Game of Life was created by John Conway in 1970, and has remained a topic of interest in computing, because it exemplifies emergent behaviour [1]. Emergent behaviour describes the phenomenon of seemingly complex behaviour forming due to the interaction of many entities following relatively simple rules.

Cellular Automata

The Game of Life is one example of a broader range of mathematical models known as 'cellular automata.' Cellular automata are models in which space and time are represented by discreet values [2]. The discreet entities that comprise the space of a cellular automata are known as cells, and each cell has a finite number of states.

Application of Cellular Automata as a Biological Model

The emergent behaviour observed in cellular automata such as the Game of Life is a phenomenon that can also be observed in certain animals. For example, flocking in birds may appear like a coordinated manoeuvre, but the behaviour emerges because each bird matches its bearing with that of neighbouring birds [3].

This report will detail the development of a cellular automata application, including Conway's Game of Life and other cellular automata. The aim of this work is to demonstrate collective behavioural concepts through the automata developed, and to evaluate the effectiveness of using cellular automata for biological simulations.

Implementation of Cellular Automata

The application developed for this project was made using pygame-ce, a game development library for Python [4].

All the cellular automata logic takes place within a ‘Grid’ object (shown in Figure 2.) A grid is an object which contains a discreet array of ‘Cells’ in a specified shape. The grid also has a ‘mode’ which determines the types of cells that will be added to the automaton. The grid has a ‘draw’ method which can be used to render its contents graphically, and an ‘update’ method which signals all cells to calculate their next status, then update to this status in parallel.

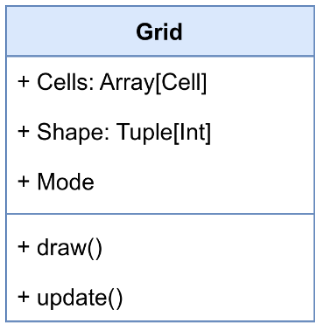

Figure 2 -Diagram for Grid class.

A ‘cell’ (shown in Figure 3,) is an abstract class that contains a reference to the grid in which it exists, as well as its position within it. A cell has a method to calculate its next state, and a method to update its state, allowing all cells to compute their next state in parallel. The way in which a type of cell uses the information available (the state of other cells in the grid, and its position,) is dependent on the implementation.

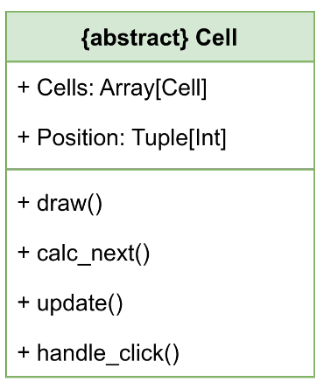

Figure 3 - Diagram for Cell class.

These two classes provide a framework which satisfies Stephen Wolfram’s definition of a cellular automaton [2]. The requirement of a discreet space is reflected by the array of cells, and discreet time is reflected by the number of ‘updates’ to the grid, or ‘generations.’

Conway's Game of Life

To implement Conway's Game of Life, a new class was made called 'VanillaCell,' which inherited from the Cell class shown in Figure 3. This cell stores a binary state 'ALIVE' or 'DEAD,' and implements calc_next() with the rules outlined in Figure 1. When 'draw' is called to render the cell, it selects a distinct colour depending on whether it is 'ALIVE' or 'DEAD.'

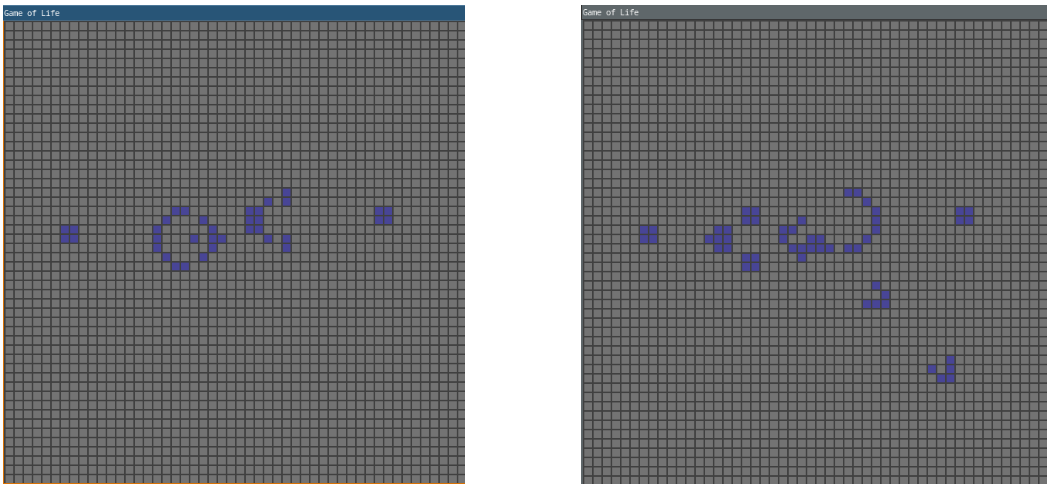

Figure 4 - Glider gun in Game of Life

Figure 4 demonstrates the graphical output of this implementation. The screenshot on the left shows the 'Gosper Glider Gun,' a well-known, initial structure drawn in the game. The screenshot on the left shows the output after multiple generations, in which the structure has demonstrated its ability to grow indefinitely by producing several 'Gliders.'

The Game of Life can be viewed as an extremely simple simulation of life because its rules are often explained as an abstraction of biological concepts. An alive cell dying with one or less neighbour represents isolation, dying with more than three neighbours represents overpopulation, and a 'dead' cell becoming 'alive' with three neighbours represents reproduction [1].

However, descriptions of popular structures indicate that the Game of Life does not always produce outputs that intuitively resemble life. A moving pattern has become known as a 'glider,' as opposed to a 'migration,' for instance.

Probabilistic Game of Life

One shortcoming of the Game of Life as a biological model is that its rules have binary evaluations, therefore there is no way to simulate an environment with different properties. Seeing how outcomes may change under slightly different conditions is often the aim of a simulation.

By making the evaluations of the rules probabilistic, the influence of a rule can be adjusted by altering the probability. This was implemented in the application by creating a 'Probabilistic Cell' type. Figure 5 shows the output of a test in which the Gosper Glider Gun was made again, except the cells only have a 50% chance of dying due to isolation. As might be expected, the output differs significantly, and the pattern seems to show a chaotic form of growth. This outcome is intuitive when viewing the game as a model of life, the cells being less likely to die due to isolation means they have a better chance of surviving and reproducing.

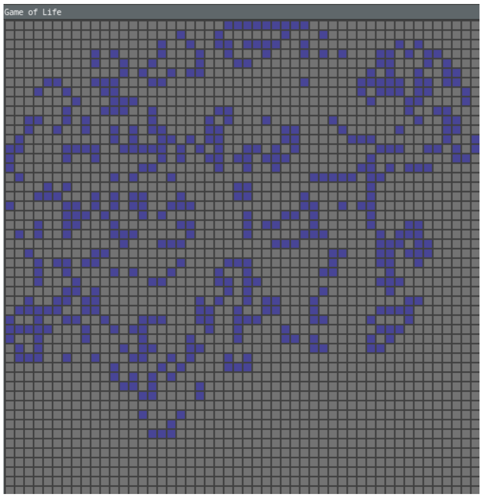

Figure 5 - Output of Glider Gun after several generations when cell death is less likely.

Rock Paper Scissors Automaton

A variant of Conway's Game of Life was developed called the 'Rock Paper Scissors' game. This game consists of three cell types, each of which will die when neighbouring one other type of cell, unless it neighbours more of the other type of cell. These rules tend to create extended spiral patterns as shown in Figure 6.

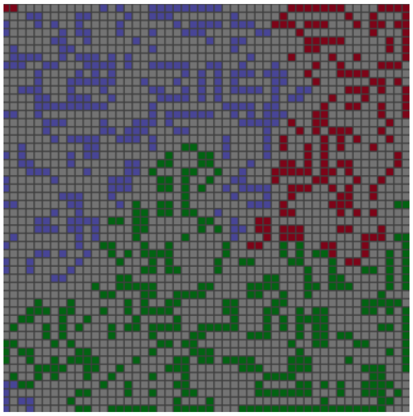

Figure 6 - Spiral pattern emerging from Rock Paper Scissors Automaton

This automaton demonstrates the ecological concepts of coexistence and cyclical dominance. Biodiversity is maintained in the environment when all three species limit one another, but if one eventually completely dominates another, the biodiversity will collapse completely. This concept has been observed in E-coli strains, where toxin-producing, toxin-sensitive and toxin-resistant strains may exhibit cyclically dominant behaviour [5].

Shell Pattern Creation Automaton

A distinctive style of cellular automaton can be seen with that of the shell pattern cell. There are still only two states where we have one that is coloured and one that is not. Cells are considered top-down for colouring. For a given cell we decide if it will be coloured by considering the Moore neighbourhood of radius one but only the cells that are above our given cell. From this neighbourhood, if 1 or 3 of the neighbours are coloured then our given cell is coloured.

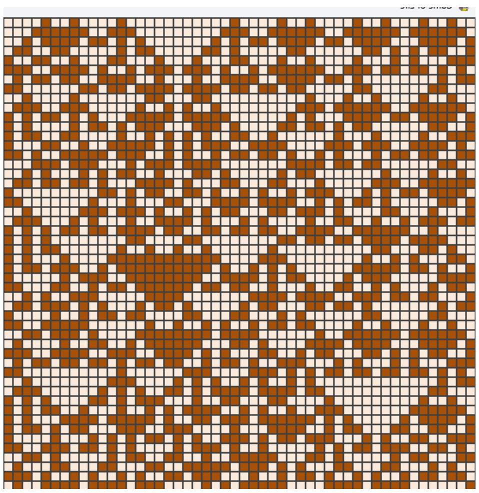

Figure 7 – Comparison of cellular automata formed pattern and a textile cone shell.

This result again shows the emergence of complex and seemingly random patterns from a simple set of well-defined rules imposed on a single cell. We are able to observe some quite visually pleasing results that resemble patterns formed in nature as seen in Figure 7 which takes away from the abstract style of ideas that the other automata display.

Iterated Prisoner's Dilemma

For this cellular automaton, the cells engage in an iterated version of the prisoner's dilemma. The rules of the game are simple and involve two players at each instance. The players have two choices: to cooperate or to defect. The payout for the player's choices is shown in Figure 8, where T > R > P > S and 2R > T + S. This results in a net payout for the two players being higher if they both cooperate and the lowest if they both defect.

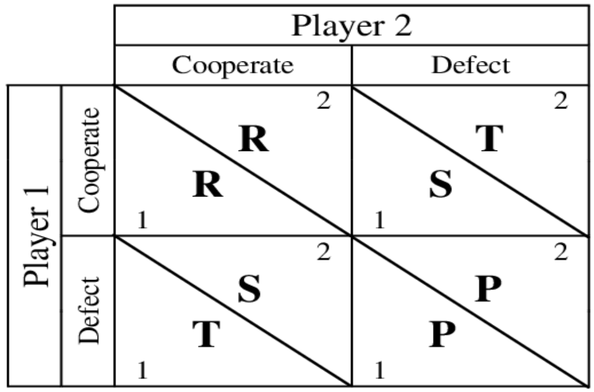

Figure 8 - Payoff matrix for player choices.

Players play the game against their neighbours in the Von Neumann neighbourhood and collect a total score from the four games. These scores are what differentiates if a cell has played well or badly during a round. How a cell chooses to either defect or cooperate for the next round is determined by its strategy and/or the choices of it's neighbours in the round previous. We implemented 4 simple strategies: always cooperate, always defect, tit-for-tat and anti-tit-for-tat. The former two speak for themselves where as tit-for-tat considers the choices of the neighbouring cells and picks the average choice from the previous round. Anti-tit-for-tat considers choices in the same way but picks opposite to tit-for-tat.

Following a round of the game, cells adopt the highest scoring strategy between itself and it's neighbours. This strategy is then used in the following game.

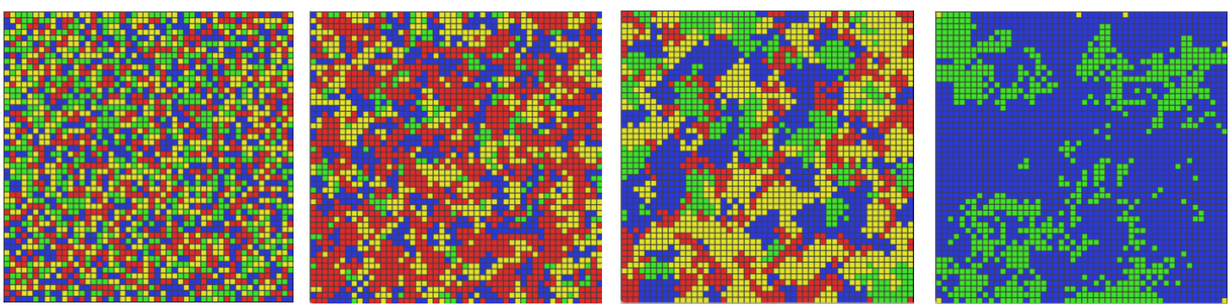

Figure 9 - Iterated Prisoner's Dilemma played with four strategies.

Figure 9 shows the development of the game over time where each colour represents one of the strategies with green, red, blue, yellow being always cooperate, always defect, tit-for-tat and anti-tit-for-tat, respectively. What we can observe is that initially cells adopt the defect strategy as it is the most profitable for the individual. As time continues, we see the emergence of cooperation with the always cooperate and tit-for-tat being the only two strategies to remain. This result shows a collective behaviour in the cells over time as they seem to realise that cooperating eventually results in the most beneficial outcome.

Immigration Automaton

Another biological model that we simulated using our program is known as the Immigration model. We used the rules from Conway's game of life and attempted to simulate the effects of an evasive species that can breed with a native species into a habitat by adding a rule known as the rainbow game of life [6]. In this model, each cell representing a species is assigned a unique colour, and when a new cell is generated from existing ones, it inherits the average colour of its parent cells. This concept mirrors real-world scenarios such as the case of the Scottish Wildcat, an endangered species due to hybridization with domestic cats introduced to the UK since the Roman empire, resulting in a significant decline in the native cat population [6].

Our implemented model enables the simulation of populations over time for multiple species capable of interbreeding, which can be modelled by Conway's Game of Life.

To achieve this, we took the cells that we used in our program to simulate game of life and gave each one a colour property. From the main menu when the option for the Immigration game was selected, we made a simple colour grid. The user can then select a colour from that grid and add cells of that species to the cell grid. As the program then steps through each time step when it hits an instance for a new cell to be spawned it takes the colours from the parent cells and generates the rbg value using the average of each from the parent cells.

We can show an example of how this works using the example of a model based off the Scottish wildcat scenario. We initially have a population of wildcats (red) and if left on their own they will reach a stable state shown in Figure 10.

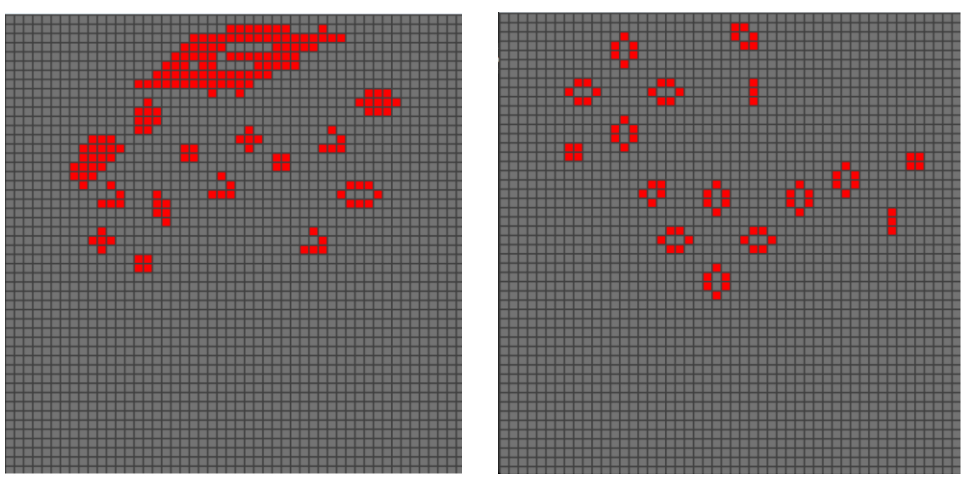

Figure 10 – Demonstration of cells left without interference.

However, if another species is introduced (blue) as shown in Figure 11. We see that the number of wildcats decreases, there are several new breeds of cat and there are also several cats that it is hard to tell. Which follows from what happens in the current case we have with Scottish wildcats [7].

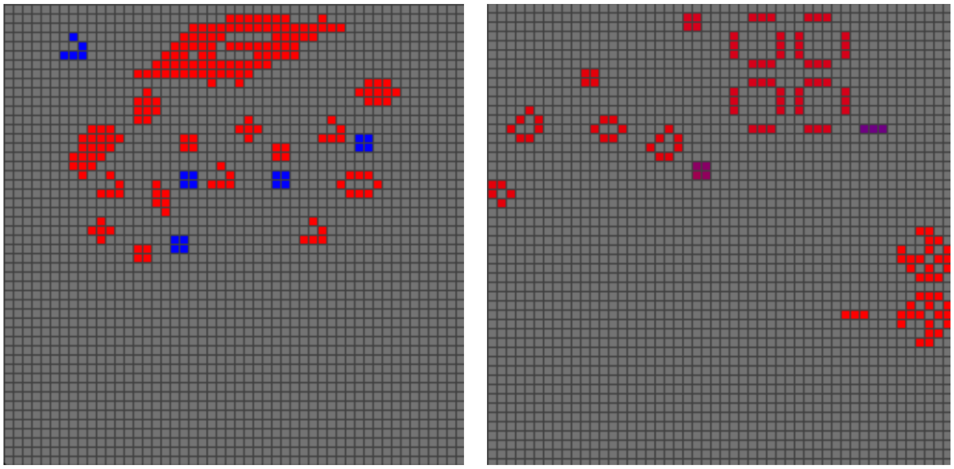

Figure 11 – Demonstration of adding an invasive species.

Furthermore, if more species of cat keep being introduced (green and pink) and then breeding with the wildcats as we see in real life [7]. This is demonstrated in Figure 12 Where part way through the running of the blue and red cells simulation we add green and pink cells. In this simulation we see many new hybrids of cat before it eventually stabilises with a small number of different hybrids that have become dominant including some that strongly resemble the original wildcat.

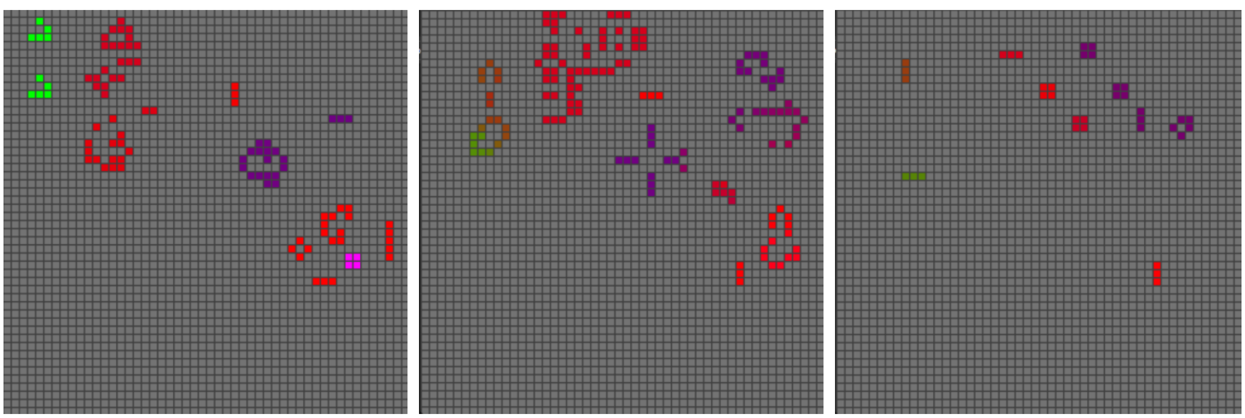

Figure 12– Demonstration of adding more invasive species.

Overall Discussion

Concepts Demonstrated

Each of the different Cellular Automata has allowed us to demonstrate different ways that cells will behave under different rule sets and offering models to demonstrate real-world scenarios. While acknowledging their inherent simplicity compared to real-world dynamics, these models serve as valuable tools for researchers studying pattern formation [8] . Our program specifically investigated biological examples for the formation of shells; using the rock, paper, scissors game to investigate patterns in a biological system where three species counter each other to maintain population sizes in an ecosystem [9] and used the immigration game to see the patterns that form from the ruleset when adding evasive species that are able to breed with existing species.

All the examples we see with the game of life have demonstrated forms of collective behaviour between the cells as we see how they need to interact with each other for a species to survive and the population is managed due to the rules against overpopulation. The Prisoners Dilemma model starts to show how we can incorporate some level of evolution into the model and watch the cells use swarm intelligence to learn good habits from each other.

When deciding which rulesets to implement and looking at our models we decided to prioritize the demonstration of biological concepts however, the game of life is also known to be used in cryptography, physics engines and music [8]. By extension we could apply some of these to our project however chose not to focus on them for the purposes of our study.

Concepts not Compatible

Although we have started to look at simple ways the cells can learn from each other and in a sense evolve. Due to the limited nature of cellular autometer only having a finite number of states, by definition, this has ment we will always reach a limit when it comes to modelling forms of evolution in a cell.

In our model we stuck to using a square grid this prevented us from exploring other concepts that require other shapes of grid or modelling concepts that would have been more appropriate in a different shaped space. We found that experiments had been done with grids that were shaped as a Penrose rhombii, 3-Isohedral convex pentagon grid and a mathematical graph to name a few [10]. This allows for modelling scenarios where cells have different numbers of adjacent cells and new patterns to form.

Advantages and drawbacks

Through the variety of interesting simulations, we have produced it is easy to see where cellular automata can be useful for modelling biological concepts and systems. Equally, it is apparent that due to their simplicity, these models hold flaws when it comes to accounting for the complexity of real-world scenarios. It is important to understand when cellular automata can be utilised and when we should opt for other solutions to modelling concepts.

Advantages

Throughout all automaton variations that we have explored, a common factor is the ease of visualisation. Alterations in colour allowed for a simple yet effective representation of different population types. Our immigration simulation made fantastic use of this feature with its ability to display inheritance between parent and child cells. The immigration simulation also highlighted cellular automata's capacity to isolate a concept to allow for ease of understanding. Where a more accurate depiction of a population may allow us to model a larger variety of factors, it could easily take away from the concept we are trying to display - hence the need for simplistic simulation.

Using cellular automata simulations possessed potential to scale very well. Due to the simple nature of the simulation, we could increase the size of the simulation without the need for significant computational power. Often the aim of the simulations would be to observe the emergence of patterns or collective behaviour in a population. With the ability to increase the number of individuals in the population, these patterns and behaviours become even easier to observe. This feature is like that of the capability to control time steps: we can increase the speed of time in the simulation. Any behaviours that might emerge over a longer period could be brought forward without the need to excessively wait on the simulation.

Drawbacks

Though the simplistic nature of cellular automata can prove useful, it also limits what the simulations can represent. We are restricted to death, survival, and reproduction all within our square lattice of cells. We also have the addition of colours for cells which helps add slightly more depth to our chosen concept. This is more a restriction of what we can observe simply by looking at the simulation rather than what we can potentially model, as seen by the slightly more complex models like the iterated prisoner's dilemma.

Another difficulty with using these models is designing the rules for which the cells must follow. A slight alteration of the rules can cause massive changes in how the simulation reacts so choosing our rules is key to modelling concepts accurately. Due to the simplicity of our set up, it is often unlikely that we can directly model a biologic system. Even with a simplified version of a biological concept mapped onto our set up, the parameters that we use are not always directly measurable in the real world and we have to make estimations which can lead to inaccurate results.

Conclusion

Through this mini project we have successfully implemented Conway's Game of Life using python's pygame library. We have explored other biological systems and concepts in the setting of cellular automata, finding some concepts can be modelled and visualised more wholly than others. Our set up has allowed us to produce some interesting results and has left us the potential to expand further into more complex concepts.

References

[1] E. M. Izhikevich, "Game of Life," 2015. [Online]. Available: http://www.scholarpedia.org/article/Game_of_Life.

[2] S. Wolfram, "Statistical Mechanics of Cellular Automata," *Reviews of Modern Physics,* vol. 55, 1983.

[3] J. T. Emlen, "Flocking Behaviour in Birds," *The Auk: Ornithological Advances,* vol. 69, no. 2, 1952.

[4] Pygame CE, "Pygame CE," [Online]. Available: https://github.com/pygame-community/pygame-ce. [Accessed 23 04 2024].

[5] K. Thierry, J. Pilar, B. Redouan, T. Céline, G. Diego and R. X.-Y. Li, "Nutrients and flow shape the cyclic dominance games between Escherichia coli strains," *Phil. Trans. R. Soc.,* vol. 378, 2022.

[6] T. Wong, "Variations on the Game of Life," Stanford.edu, 2008. [Online]. Available: https://cs.stanford.edu/people/eroberts/courses/soco/projects/2008-09/modeling-natural-systems/gameOfLife2.html. [Accessed 8 May 2024].

[7] K. Kilshaw, "Scottish wildcats," 2011. [Online]. Available: https://digital.nls.uk/pubs/e-monographs/2020/216547659.23.pdf. [Accessed 8 May 2024].

[8] Ateethkj, "Conway's Game of Life: Simple yet Complex," Medium.com, 26 March 2023. [Online]. Available: https://medium.com/@ateethkj1592/conways-game-of-life-simple-yet-complex-50dd0104dc60#:~:text=Conway's%20Game%20of%20Life%20provides,physical%20systems%20to%20neural%20networks.. [Accessed 8 May 2024].

[9] M. Hoffman, S. Suetens, U. Gneezy and M. A. Nowak, "An experimental investigation of evolutionary dynamics in the Rock-Paper-Scissors game," *Scientific Reports,* vol. 5, no. 1, 2015.

[10] Code Golf, "Implement the Game of Life on Anything but a Regular Grid," Stack Exchange, 2021. [Online]. Available: https://codegolf.stackexchange.com/questions/35827/implement-the-game-of-life-on-anything-but-a-regular-grid. [Accessed 8 May 2024].

[11] C. K. Hemelrijk and H. Hildenbrandt, "Some Causes of the Variable Shape of Flocks of Birds," *PLoS ONE,* vol. 6, no. 8, 2011.