

# An experiment on reversing Game Of Life using Genetic Algorithms

Isabel Prieto Payo, Víctor Gutiérrez García, *Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid*

**Abstract**—Cellular Automata are useful in various fields, including biology, physics, and more. This study aims to harness the versatility and robustness of Genetic Algorithms to tackle challenging optimization problems, specifically in determining the initial state of Conway’s Game of Life from a given final configuration. Our focus is on leveraging Cellular Automaton modeling. We review previous implementations of this approach and introduce a set of metrics to evaluate their performance. Finally, we propose an alternative fitness function, incorporating a regularization term to favor lattices with a lower distribution of future dead cells.

**Index Terms**—Cellular Automaton, fitness function, Game Of Life, Genetic Algorithms.

## I. INTRODUCTION

CELLULAR Automata (CA) exhibit dynamical patterns that can be interpreted as the movement of particles through a physical medium. They are enough flexible to capture the essence of both continue and discrete processes, providing the ability to model the behaviour of complex systems from different fields of knowledge based on local and simple rules. For instance, in the biological field, Cellular Potts Model (CPM) and Lattice Gas Cellular Automata (LGCA) are combined to simulate the dynamic of avascular cancer growth [1]; in physics, a cellular-automata numerical method is used to build a comprehensive model of ice crystal growth from water vapor providing a detailed physical picture of the molecular structure and dynamics of the ice surface during solidification [2]; and in the social science field, a mesoscopic cellular automata model has been tailored for simulating evacuations in large-scale scenarios [3].

A CA is formally defined by a quadruple  $\{Z, S, N, f\}$ , where  $Z$  is the finite or infinite cell lattice,  $S$  is a finite set of states or cell values,  $N$  is the finite cell neighborhood and  $f$  is the local transition function defined by the state transition rule. For a time-discrete and binary-cellular automaton, the state of an observed cell  $c_x \in S = \{0, 1\}$  at time  $t + 1$  is determined by the evaluation of the local transition function  $f$  on the states of the neighboring cells  $N_x$  at time  $t$ , where  $x \in Z$  defines the position of each cell in the lattice, that is,  $f(N_x^t) = c_x^{t+1}$ . The global transition function  $F$  is defined by the local transition function  $f$  as  $C^{t+1} = F(C^t) =$

This work was supported by the Universidad Politécnica de Madrid. Finally, the authors would like to thank Escuela Técnica Superior de Ingenieros (ETSI) de Telecomunicación for its indirect support.

Isabel Prieto Payo and Víctor Gutiérrez García are with Escuela Técnica Superior de Ingenieros (ETSI) de Telecomunicación, Universidad Politécnica de Madrid, 28040, Madrid, Spain (e-mail: isabel.prieto.payo@alumnos.upm.es, v.garcia@alumnos.upm.es)

Manuscript submitted in May 20, 2024.

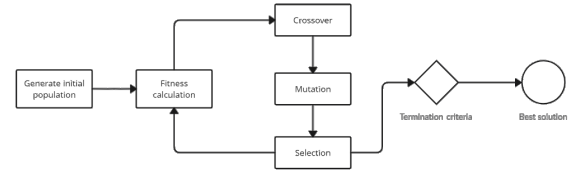


Fig. 1. GA sequence diagram of genetic operators that mimic the process of natural selection. First of all, a population of random solutions (chromosomes) is generated. The fitness of each candidate solution is calculated. After that, the genetic operators called crossover, mutation, and selection are performed in the depicted sequence.

$\dots f(N_{x-1}^t)f(N_x^t)f(N_{x+1}^t)\dots$ , where  $C^t \in S^Z$  is the global state that comprises the configuration at time  $t$  of all the cells in the CA [4].

This work will focus on the CA describing the Game Of Life (GOL) designed by John Conway and popularized by Martin Gardner in [5]. It resembles real-life processes thanks to analogies with the rise, fall and alternations of a society of living organisms. The basic idea is to start with a simple configuration of counters (organisms), one to a cell, then observe how it changes as you apply Conway’s “genetic laws” for births, deaths, and survivals. The rules are: every counter with two or three neighboring counters survives for the next generation; each counter with four or more neighbors dies (is removed) from overpopulation, and every counter with one neighbor or none dies from isolation; and each empty cell adjacent to exactly three neighbors (no more, no fewer) is a birth cell, with a counter placed on it at the next move.

Reversing GOL consists of guessing the initial configuration of all the cells in the CA lattice, given an end lattice and a number of time steps. To this end, this work will apply a Genetic Algorithm (GA), an optimization procedure for searching and learning inspired by the process of natural selection proposed by Darwin, where the fittest individuals survive and reproduce more similar offspring while weak individuals are eliminated with the passage of time. More specifically, they are based on reproduction, selection, crossover and mutation. The process flow is depicted in Figure 1.

The general GA designed for solving the reversing GOL challenge is summarized with the pseudo-code of Algorithm 0.

The rest of the paper is organized as follows. Section 2 is dedicated to introducing the main materials and methods to conduct our experiment; Section 3 presents the results; and Section 4 is dedicated to a discussion and drawing the main conclusions

---

**Algorithm 1** Genetic Algorithm tailoring for GOL Cellular Automaton
 

---

```

1: Initialize: population with random individuals  $X^0$ 
2: for each individual  $X^0$  in population do
3:   for time step  $t$  from 0 to  $T$  do
4:     Apply GOL global transition function to  $X^t$ 
5:   end for
6: end for
7: Evaluate: each  $X^T$ 
8: while max generation number is reached do
9:   select parents
10:  recombine pairs of parents
11:  mutate the resulting offspring
12:  evaluate fitness function
13:  select individuals for the next generation
14: end while

```

---

## II. MATERIALS AND METHODS

This sections starts by detailing the resources that were downloaded to carry out the experimentation detailed in this work. Then, it presents the key aspects of the basic GA along the modifications that aim to improve the algorithm performance in the reversing GOL challenge. It also summarizes the metrics used to evaluate the current optimization problem in a suitable way. Finally, some considerations about the resulting experimental set-up are commented.

### A. Materials: code repository and database

The concept for this experiment is inspired by a Kaggle competition that aimed to reverse-engineer Conway's Game of Life [6] as well as the proposed method described in "Reversing Game of Life Using Genetic Algorithms" [7]. Our proposal, based on a forum entry, is available at [8].

### B. Methods: fitness functions

First of all, the mapping between the CA and the GA frameworks is presented. The lattice is a  $N \times N$ -sized one, where  $N = 20$ . The initial lattices configurations  $\{C^0\}$  that generate the given end lattice  $C_{ref}^T$  represent the individuals of the population. Each cell  $c_x$  within  $C^0$  is a gene, and its state  $S \in \{0, 1\}$  is an allele, that represents whether the cell is alive (1) or has died (0). The benchmark fitness function applies the global transition function  $F$  recursively through  $T$  time steps, compares the resulting lattice  $C^T$  to the given one  $C_{ref}^T$  and measures the percentage of coinciding cells. Therefore, the benchmark fitness function  $f_{bm}$  is defined as follows:

$$f_{bm}(C_{ref}^T, C_i^0) = \frac{1}{N^2} \left( \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} XNOR(F^T(x_{ij}^0, c_{ij}^T)) \right) \quad (1)$$

where  $c_{ij}^T$  is each cell state belonging to  $C_{ref}^T$  and  $x_{ij}^0$  represents each cell state belonging to the evaluated individual over which the global transition function  $F$  is applied  $T$  times.

This work approach of improving the overall performance of the algorithm is based on adding a regularization term to the fitness function to penalize the presence of those cells that

in the first iteration of  $F$  are going to die (see the previously defined rules of the GOL). This phenomena is explained by the Garden of Eden concept: a CA has a Garden of Eden is and only if it has two different finite configurations that evolve into the same configuration in one step. Therefore, it is proposed a regularized version of the benchmark fitness function with the following expression.

$$f_{reg}(C_{ref}^T, C_i^0) = f_{bm}(C_{ref}^T, C_i^0) - k \cdot Q(C_i^0) \quad (2)$$

where  $Q$  accounts for the sets of cells that will die in the second iteration of the lattice and  $k$  is a regularization factor.

Extending this regularization hypothesis to improve the algorithm performance, a novel approach is considered: each lattice, from the initial one to the end one is analyzed in terms of cells to die in the next iteration to add up to the regularization term. The extended regularized version  $f_{reg,ext}$  of the benchmark fitness function is presented hereinafter.

$$f_{reg,ext}(C_{ref}^T, C_i^0, \dots, C_i^T) = f_{bm}(C_{ref}^T, C_i^0) - k \cdot \sum_{t=0}^{T-2} Q(C_i^t) \quad (3)$$

### C. Methods: evaluation metrics

The use of metrics in algorithms is essential, as they provide a quantitative basis for evaluating performance and effectiveness. Metrics enable developers to measure how well their solutions perform. By defining metrics, we can objectively compare different algorithms or models, identify areas for improvement, and make informed decisions about which approach to use in a given context. In [7], the only metric used is the Mean Absolute Error (MAE), which measures accuracy by comparing the initial and predicted boards. We believe that this method alone is insufficient to assess the algorithm's effectiveness. Since we are working with 20x20 sets, resulting in a total of 400 alleles, initial configurations with few points can lead to low error values without actually matching any points, as the influence of a few values gets diluted. To achieve more accurate results, we plan to implement the Hamming distance with local neighborhoods. The implementation of the Hamming distance that we did takes into account the local neighborhoods of each one of the cells, capturing the spatial pattern and structure of the data. This makes sense in this context since we will take into account the spacial patterns of group of cells instead of individual cells.

Additionally, we have chosen to implement other common classification metrics, including accuracy, precision, recall, F1 score, and mean squared error (MSE). We find accuracy and precision particularly relevant for this problem.

### D. Experimental setup

We possess a dataset comprising 50,000 entries from various boards. Each row includes an ID, a delta value representing the number of time steps (or generations in the Game of Life) until the final board is reached, and 800 binary values. The first 400 binary values correspond to an initial 20x20 board of alleles, while the next 400 represent a final board of the same dimensions. First, we opted to preprocess the data. Given

that fewer time steps make it more challenging to achieve an accurately scored board, we decided to exclude boards with fewer than four generations.

Additionally, we decided to implement the Game of Life in Python. Our goal is to evolve the guessed initial boards and compare the resulting final state with the dataset's provided final state. This will help us identify any discrepancies between the dataset's final state and the one derived from our guessed initial state.

We chose to retain most of the hyperparameters from the previous implementation of the genetic algorithm for simplicity. The only modification we made was changing the number of generations for the algorithm, setting it to 200 for each board. We decided to retain several hyperparameters from the previous implementation of the genetic algorithm for simplicity. The only modification we made was to set the number of generations to 200 for each board.

After extensive testing, we chose a value of 0.1 for the hyperparameter controlling our regularization term, as higher values proved detrimental to the initial boards. Due to the complexity of selecting this value, we will propose methods to optimize this hyperparameter to maximize the score for each population in the genetic algorithm.

### III. RESULTS

After evaluating our model, we chose two scenarios in which we can visualise the benefits of our modification to the fitness function. In the Figure 2 we observe that the implementation of the regularization term improves the initial board by reducing the number of single cells. Additionally, this approach helps us achieve a final state that more closely resembles the original one. For our approach we obtain a Hamming distance of 0.7375, an accuracy of 0.9575 and a precision of 0.21.

In the Figure 3, we can see that in challenging scenarios, the genetic algorithm proves to be resilient. However, our implementation preserves the density zones of the actual final state, resulting in a final state that differs partially, and seems better, as the one obtained with the basic implementation.

For our approach we obtain a Hamming distance of 0.3225, an accuracy of 0.07225 and a precision of 0.1915.

### IV. DISCUSSION AND CONCLUSIONS

This work faces the reversing GOL challenge, which is a hard optimization problem, by implementing a GA. The overall behavior of the setup has been analyzed in order to propose modifications to improve algorithm performance.

After implementing the GA featuring the benchmark fitness function, it arises the fact that cells that die in the first time step (from  $C^0$  to  $C^1$ ) are not tracked by the fitness function and therefore, adding or removing this kind of cells at the initial lattice has no effect on the fitness function. This phenomena is considered within the Garden of Eden theorem and implies that two different finite configurations evolve into the same configuration in one step. This phenomena is depicted in the following figures.

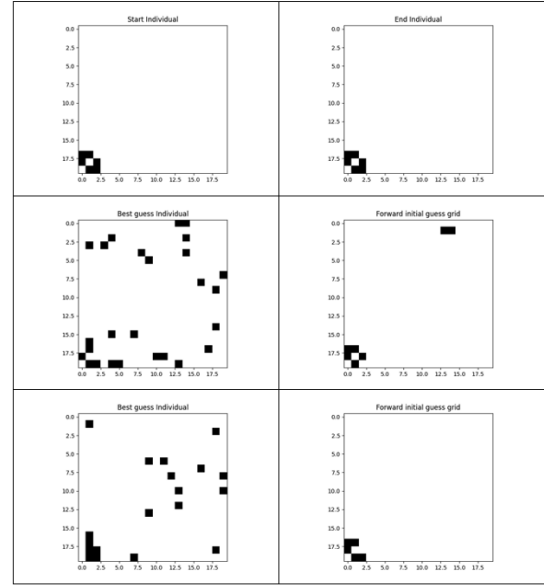


Fig. 2. *a. Real start individual; b. Real end Individual; Initial implementation without regularization term c. Predicted start individual; d. Predicted forward state based on predicted start individual; Implementation including regularization term on fitness function e. Predicted start individual; f. Predicted forward state based on predicted start individual.*

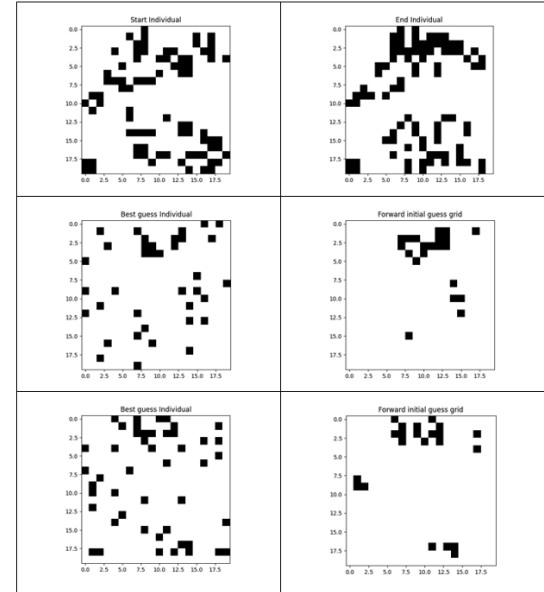


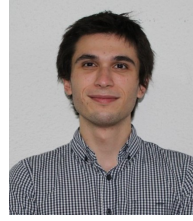
Fig. 3. *a. Real start individual; b. Real end Individual; Initial implementation without regularization term c. Predicted start individual; d. Predicted forward state based on predicted start individual; Implementation including regularization term on fitness function e. Predicted start individual; f. Predicted forward state based on predicted start individual. It struggles sometimes.*

Also, regarding the benchmark metric used for evaluating the performance of the algorithm, computing the percentage of matching cells seems not to be the most suitable decision. For instance, the Garden of Eden phenomena when appearing lonely cells in a lattice, is barely reflecting on the previously employed metrics.

Due to lack of resources an exhaustive study in terms of number of experiments has not been possible. Instead, the results of this work comprises samples that we consider representative, in which we observe a notable improvement compared to the previous implementation, leading to the validation of the proposed algorithm modifications that aimed to improve the performance of the performed experiment.

Furthermore, with the experiments conducted in this work, it is reinforced the flexibility that GA provides to solve optimization problems, both by being able to approach to a solution of the GOL challenge and for allowing modifications that improve algorithm performance.

In addition, it has to be noted that a good understanding and analysis of the dynamics and behaviors of the CA has been crucial to develop the proposed line of improvement.



**Víctor Gutiérrez García** was born in Barcelona, Spain. He received the B.Sc. degrees in Telecommunications Engineering from the Universidad Politécnica de Madrid (UPM), Spain, in 2022. Since 2023 he is a M.Sc in Signal and Communication Theory student at Universidad Politécnica de Madrid (UPM), Spain.

#### REFERENCES

- [1] M. Ghaemi and A. Shahrokhi, "Combination of the cellular potts model and lattice gas cellular automata for simulating the avascular cancer growth," *Lecture Notes in Computer Science*, vol. 4173, 2006. DOI: [10.1007/11861201\\_35](https://doi.org/10.1007/11861201_35).
- [2] K. Libbrecht, "Toward a comprehensive model of snow crystal growth: Measurements of changes in attachment kinetics from background gas interactions at -5 c," Nov. 2012.
- [3] W. Lv, J. Wang, Z. Fang, and D. Mao, "Simulation method of urban evacuation based on mesoscopic cellular automata," *Acta Physica Sinica*, 2023. DOI: [10.7498](https://doi.org/10.7498).
- [4] J. C. López-González and A. Rueda-Toicen, "Search of complex binary cellular automata using behavioral metrics," *Complex Systems*, vol. 24(1), pp. 1–25, Mar. 2015. DOI: [10.25088](https://doi.org/10.25088).
- [5] M. Gardner, *The fantastic combinations of john conway's new solitaire game "life"*, Last accessed 20 May 2024, 1970. [Online]. Available: <https://web.stanford.edu/class/sts145/Library/life.pdf>.
- [6] Kaggle, *Conway's reverse game of life*, Last accessed 20 May 2024, 2014. [Online]. Available: <https://www.kaggle.com/c/conway-s-reverse-game-of-life>.
- [7] P. Tyshevskyi, *Reversing game of life using genetic algorithms*, Last accessed 20 May 2024. [Online]. Available: <https://medium.com/@ptyshevs/rgol-ga-1cafc67db6c7>.
- [8] I. P. Payo and V. G. García, *Bioinspired learning project*, Last accessed 20 May 2024. [Online]. Available: [https://github.com/isapri/bioinspired%5C\\_learning](https://github.com/isapri/bioinspired%5C_learning).



**Isabel Prieto Payo** was born in Huelva, Spain. She received the B.Sc. degrees in Telecommunications Engineering from the Universidad Politécnica de Madrid (UPM), Spain, in 2022. Since 2022 she is a M.Sc in Signal and Communication Theory student at Universidad Politécnica de Madrid (UPM), Spain.