A PROBABILISTIC METHOD FOR PREDICTING STOCHASTIC BEHAVIOR IN CONWAY'S GAME OF LIFE

CALEB KOCH

Mathematics
Extended Essay
May 2016
Galileo Magnet High School
Word Count: 3934

Acknowledgements

I would like to thank Mr. Bill Lawrence, my adviser, who took time out of his busy, significant career to read the research article of an insignificant teenager. I would also like to acknowledge Mrs. Kathy Shanks who mails all of the extended essays and is arguably the most overworked and underpaid teacher in the school. Basically, if it weren't for Mrs. Shanks, the IB program probably would not exist at our school and if it weren't for Mr. Lawrence, neither would my adviser.

Abstract. This paper documents a study on the predictability of cellular automaton systems in Conway's Game of Life (GoL). It investigates the question: *are cellular automaton systems in GoL predictable and if so, how predictable are they?* The research involves the development of a probabilistic model which relies on limited information to predict the immediate, general behavior of regions within a finite automaton. A simulation program is developed to test the accuracy of the model. The program implements the probabilistic model alongside a random guessing algorithm and tracks the progress of both in predicting the behavior of randomly generated automata. The results from the simulation are analyzed using the chi square test of independence. With a p value less than 0.05, the model is established as statistically effective. As such, the results of the research suggest that the behavior of a dynamical system is not as unpredictable as previously regarded. The limitations and future work for this study are detailed and explained.

Contents

1. Introduction

Cellular automata (CAs) were invented in the 1940s by John von Neumann [1] and have typically been defined as consisting of "cells on a grid...that evolves through a number of discrete time steps according to a set of rules based on the states of neighboring cells [2]." As such, every CA consists of a certain type of cellular space and a transition rule. The cellular space can be described in terms of any d-dimensional regular lattice of cells with finite boundary conditions. Each cell has k number of states where k is most commonly used as a positive integer ($k : k \in \mathbb{Z}^+$) [3]. The set of states is denoted Σ, making $k = |\Sigma|$. The state of cell C located at index i and at time t is denoted C_i^t such that $C_i^t \in \Sigma$. The neighborhood of C consists of the state of the adjacent cells or $C_{i\pm1}^t$ and is used to determine the state of C at time $t+1$. The function φ, known as the transition rule, is used to compute this state. Thus, $C_i^{t+1} = \varphi(C_{i-1}^t, C_i^t, C_{i+1}^t)$. It should be noted that different CAs use different neighborhoods. Namely, a 2-dimensional CA uses either the Moore neighborhood or the von Neumann neighborhood shown in Figure 1.

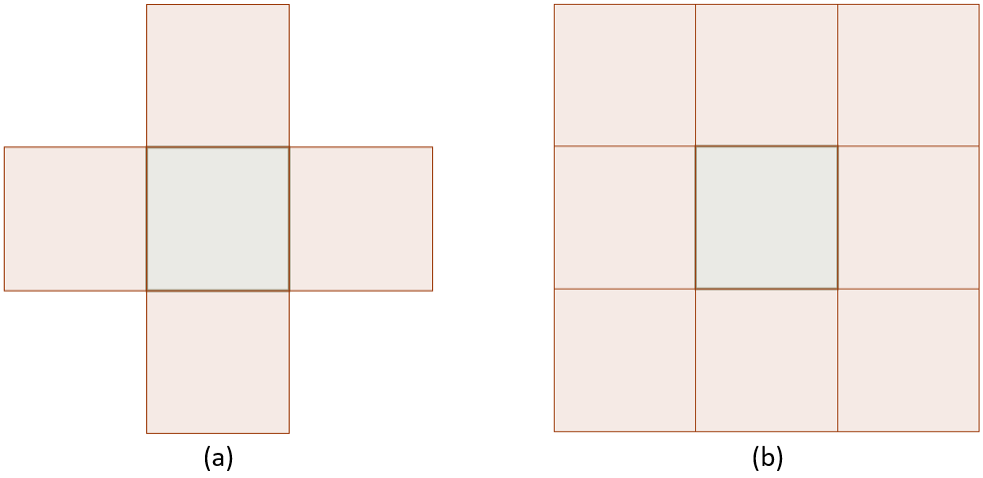


Figure 1. *(a) gives a depiction of the von Neumann neighborhood with the gray cell in the center being updated. while (b) represents the Moore neighborhood, with the center gray cell being updated.*

1.1. Game of Life.

In the 1970s, Cambridge Professor John Conway invented a 2-dimensional CA consisting of a Moore neighborhood that he called "life" [4]. The automaton was said to model populations and included these rules [3]:

- $\Sigma = \{0, 1\}$ where a state of 0 represents a "dead" member of the population and 1 represents an "alive" member of the population
- The transition rule consists of either the death of a member (going from $1 \to 0$), birth of a member ($0 \to 1$), or no change ($0 \to 0$ or $1 \to 1$)
- With zero or one bordering member alive in a cell's neighborhood, death occurs from loneliness
- With two or three members alive in the neighborhood, the state of an alive member remains unchanged

- If a dead member's neighborhood consists of three live members, a new member is "born" $(0 \to 1)$
- If a live member's neighborhood consists of more than three live members, it will die from overpopulation

To propose a formal definition of φ, let the state of a cell C at position (m, n) in the 2-dimensional Cartesian grid (i.e. $\mathbb{Z}^2$ integer lattice) such that $m, n \in \mathbb{Z}$ at time t be given as $C^t_{m,n}$. Then, S or the number of live members in the neighborhood N of $C^t_{m,n}$ is

$$S = \sum C^t_{g,h} : g \in \{m-1, m, m+1\}, h \in \{n-1, n, n+1\} \wedge (g \neq m \vee h \neq n)$$

The transition rule can then be summarized as piecewise function:

$$C^{t+1}_{m,n} = \varphi(N) = \begin{cases} 1 & : S = 3 \wedge C^t_{m,n} = 0 \\ 0 & : S \neq 3 \wedge C^t_{m,n} = 0 \\ 1 & : (S = 2 \vee S = 3) \wedge C^t_{m,n} = 1 \\ 0 & : (S < 2 \vee S > 3) \wedge C^t_{m,n} = 1 \end{cases}$$

Despite the simplicity of this function, $\varphi(N)$ can produce complex behavior and can lead to functional behavior. Hence, many computer scientists and mathematicians have taken interest in these particular CAs not only for their growing popularity but also their versatile utility. The increase in research has led to some interesting discoveries. For example, GoL has been shown to be Turing complete, indicating that a Turing machine can be implemented in it [5, 6]. However, few studies have studied large-scale patterns in GoL [7]. One research study [8] presents a large-scale pattern involving the evolution of squares implemented as start configurations in GoL, but it lacks rigorous mathematical proof to show that this pattern continues indefinitely.

Since behavior in GoL is so complex, a mathematical formula to give the outcome of an initial start state is nonexistent. Thus, proving such patterns from [8] would be difficult and would require a novel approach. It would require some type of equation or model which could take limited information regarding the start state of an automaton and return the probability that some type of behavior is exhibited. The solution to this problem begins to be developed in this study. Namely, the aforementioned model is developed and tested for accuracy. Section 2 highlights the mathematical development of this model and outlines the simulation program used to test it.

2. Developing the Model

2.1. **Probabilistic View on Cellular Automata Behavior.** In order to develop a model to predict the stochastic behavior of CAs, several notations need to be established, and parameters need to be outlined.

Definition 2.1. Since $C^t_{m,n}$ gives the state of cell C at time t and position m, n, let P denote the maximum value of m such that $C^t_{m,n} = 1$ and let Q denote the maximum value of n such that $C^t_{m,n} = 1$. Similarly, let p denote the minimum value of m such that $C^t_{m,n} = 1$ and let q denote the minimum value of n such that $C^t_{m,n} = 1$.

Definition 2.2. For any given t, the CA is represented as a matrix, $\mathbf{A}_t$, such that

$$\mathbf{A}_t = \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ 0 & C^t_{p,Q} & \cdots & C^t_{P,Q} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & C^t_{p,q} & \cdots & C^t_{P,q} & 0 \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix}$$

The buffer of dead cells around the matrix exists because such cells are subject to the transition rule at the given time t. It is important that they are included in future calculations.

Definition 2.3. The variable s denotes the sidelength of $\mathbf{A}_t$ (it includes the buffer layer).

The automaton is divided into four main regions: outer, corner, border, and inner.

Definition 2.4. The buffer layer of dead cells in $\mathbf{A}_t$ is referred to as the *outer layer*. It includes two subregions: S_0 and regular outer cells. At time t there are no live cells in this region.

Definition 2.5. The four cells $C^t_{p,Q}, C^t_{P,Q}, C^t_{p,q}, C^t_{P,q}$ are termed *corner cells*. The number of live corners cells at time t is denoted L_c. The probability density, γ_c, gives the probability that a randomly selected corner cell will be alive at time t and is expressed as

$$\gamma_c = \frac{L_c}{4}$$

Definition 2.6. The third group of cells consists of those located in the same layer as the corner cells. These cells are referred to as *border cells* and are situated between the corner cells, the outer cells, and the inner cells. The border cells are divided into the subregions S_1 and regular border cells. The number of live border cells at time t is given by L_b. The probability density, γ_b is

$$\gamma_b = \frac{L_b}{2(P-p-1)+2(Q-q-1)}$$

Definition 2.7. The last group of cells is designated *inner cells*. These cells remain within the border cells. This region consists of three subregions: S_2, S_3, and regular inner cells. At time t, there are L_i inner cells. The probability density, γ_i is expressed as

$$\gamma_i = \frac{L_i}{(P-p-1)(Q-q-1)}$$

A graphic is provided in Figure 2 to illustrate the general structure of the regions and their place in a CA.

2.2. **Analyzing Immediate States.** The analysis begins at trying to understand the probable state of a given cell at time $t+1$. To do so, the probable state of a given cell based on its inclusion in a specific group is determined. Thus, each region is analyzed independently to determine the probability that a randomly selected cell is alive at time $t+1$. The goal for this analysis is to provide an expression for

$$P(C^{t+1}_{m,n} = 1) \mid m \in [p-1, P+1] \wedge n \in [q-1, Q+1] \tag{2.8}$$

X	X	S_0	0	0	S_0	X	X
X	C	S_1	B	B	S_1	C	X
S_0	S_1	S_2	S_3	S_3	S_2	S_1	S_0
0	B	S_3	I	I	S_3	B	0
0	B	S_3	I	I	S_3	B	0
S_0	S_1	S_2	S_3	S_3	S_2	S_1	S_0
X	C	S_1	B	B	S_1	C	X
X	X	S_0	0	0	S_0	X	X

FIGURE 2. *Xs denote cells that will not change states. S denotes cells within their respective subregion. B, C, and I denote regular border, corner, and inner cells respectively (regions are demarcated by shading)*

2.3. **Inner Cell Behavior.** The inner region is further divided into three subregions. Each subregion is considered separately. It is important to note that for a CA with side lengths greater than eight cells, there would be more subregions. In general, there are $s - 6$ layers in the inner region. The innermost layer would consist of "regular cells" (considered in Section 2.3.3) while the other layers would each consist of cells whose neighborhoods and probabilities are similar to that of those in the S_2 and S_3 subregions. Although the addition of these layers is not explicitly addressed, it would be easy to add them to the model because the calculations would be the same.

2.3.1. *S_2 Subregion.* This subregion is analogous to the corner cells in the second to last layer of cells. S_2 cells are those located in the corners of the third to last layer of cells. Their neighborhood resembles this general structure:

$$\begin{pmatrix} \gamma_c & \gamma_b & \gamma_b \\ \gamma_b & \gamma_i & \gamma_i \\ \gamma_b & \gamma_i & \gamma_i \end{pmatrix}$$

where the S_2 cell is located in the center. Given that the cell is initially alive, there are two general cases that need to be considered. In the first case, there are two live neighbors and seven dead neighbors. The total combination of cases can be enumerated in Table 1 (note that when a probability density is grouped under "Alive," it signifies that a cell neighboring the S_2 cell is alive, and that this cell's chance of being alive is described by the aforementioned probability density). Note that some combinations are excluded because they would be repeats of the ones already listed. Also, the addition symbols are included because the total probability for Case 1 can be given as the sum of these individual probabilities times the probability that the S_2 cell is alive. This working can be roughly verified by observing that there are $\binom{8}{2}$ or 28 different probabilities. Thus, the sum of the probability coefficients should equal 28, which it does.

Alive	Dead	Probability
$\gamma_c\gamma_b$	$\gamma_b\gamma_b\gamma_i\gamma_b\gamma_i\gamma_i$	$4\gamma_c\gamma_b(1-\gamma_b)^3(1-\gamma_i)^3$ +
$\gamma_i\gamma_c$	$\gamma_b\gamma_b\gamma_b\gamma_i\gamma_b\gamma_i$	$3\gamma_i\gamma_c(1-\gamma_i)^2(1-\gamma_b)^4$ +
$\gamma_i\gamma_i$	$\gamma_c\gamma_b\gamma_b\gamma_b\gamma_i\gamma_b$	$3(\gamma_i)^2(1-\gamma_c)(1-\gamma_b)^4(1-\gamma_i)$ +
$\gamma_b\gamma_i$	$\gamma_i\gamma_c\gamma_b\gamma_b\gamma_b\gamma_i$	$12\gamma_b\gamma_i(1-\gamma_i)^2(1-\gamma_b)^3(1-\gamma_c)$ +
$\gamma_b\gamma_b$	$\gamma_i\gamma_b\gamma_i\gamma_i\gamma_c\gamma_b$	$6(\gamma_b)^2(1-\gamma_b)^2(1-\gamma_i)^3(1-\gamma_c)$

TABLE 1. S_2 Case 1

In Case 2, there are three live neighbors and five dead ones. The total number of possibilities should equal $\binom{8}{3}$ or 56. The table below enumerates these possibilities.

Alive	Dead	Probability
$\gamma_c\gamma_b\gamma_b$	$\gamma_b\gamma_i\gamma_b\gamma_i\gamma_i$	$6\gamma_c(\gamma_b)^2(1-\gamma_b)^2(1-\gamma_i)^3$ +
$\gamma_i\gamma_c\gamma_b$	$\gamma_b\gamma_b\gamma_i\gamma_b\gamma_i$	$12\gamma_i\gamma_c\gamma_b(1-\gamma_b)^3(1-\gamma_i)^2$ +
$\gamma_i\gamma_i\gamma_c$	$\gamma_b\gamma_b\gamma_b\gamma_i\gamma_b$	$3(\gamma_i)^2\gamma_c(1-\gamma_b)^4(1-\gamma_i)$ +
$\gamma_b\gamma_i\gamma_i$	$\gamma_c\gamma_b\gamma_b\gamma_b\gamma_i$	$12(\gamma_i)^2\gamma_b(1-\gamma_c)(1-\gamma_b)^3(1-\gamma_i)$ +
$\gamma_b\gamma_i\gamma_b$	$\gamma_i\gamma_i\gamma_c\gamma_b\gamma_b$	$18\gamma_i(\gamma_b)^2(1-\gamma_i)(1-\gamma_b)^2(1-\gamma_i)^2$ +
$\gamma_b\gamma_b\gamma_b$	$\gamma_i\gamma_b\gamma_i\gamma_i\gamma_c$	$4(\gamma_b)^3(1-\gamma_b)(1-\gamma_i)^3(1-\gamma_c)$ +
$\gamma_i\gamma_i\gamma_i$	$\gamma_b\gamma_b\gamma_b\gamma_b\gamma_c$	$(\gamma_i)^3(1-\gamma_b)^4(1-\gamma_c)$

TABLE 2. S_2 Case 2

Note that the sum denoted in Table 1 will be symbolized T_1 while the sum in Table 2 will be T_2. There must also be consideration for the case where the central cell is initially dead. In such cases, three live neighbors will result in the life of the center cell. The probability of three live neighbors was already calculated in Table 2, so the total probability that a cell in the S_2 region is alive at time $t+1$ can now be calculated.

$$\begin{aligned} P(S_2) &= P(Case1) + P(Case2) + P(Other) \\ &= \gamma_i(\mathrm{T}_1) + \gamma_i(\mathrm{T}_2) + (1-\gamma_i)(\mathrm{T}_2) = \mathrm{T}_2 + \gamma_i(\mathrm{T}_1) \end{aligned} \tag{2.9}$$

2.3.2. *S_3 Subregion.* This subregion consists of cells that are located in the outer layer of the inner region in between S_2 cells. The process of identifying the probability that a cell in this region is alive at time $t+1$ is identical to that of the S_2 subregion. First, it is important to identify the general neighborhood in terms of probability densities. A given cell in this subregion would have a neighborhood resembling:

$$\begin{pmatrix} \gamma_b & \gamma_b & \gamma_b \\ \gamma_i & \gamma_i & \gamma_i \\ \gamma_i & \gamma_i & \gamma_i \end{pmatrix}$$

If the cell is initially alive, there are two cases in which that same cell is alive in the subsequent generation. Case 1 is depicted in Table 3.

Alive	Dead	Probability
$\gamma_i\gamma_i$	$\gamma_i\gamma_i\gamma_i\gamma_b\gamma_b\gamma_b$	$10(\gamma_i)^2(1-\gamma_i)^3(1-\gamma_b)^3$ +
$\gamma_i\gamma_b$	$\gamma_i\gamma_i\gamma_i\gamma_i\gamma_b\gamma_b$	$15\gamma_i\gamma_b(1-\gamma_i)^3(1-\gamma_b)^2$ +
$\gamma_b\gamma_b$	$\gamma_i\gamma_i\gamma_i\gamma_i\gamma_i\gamma_b$	$3(\gamma_b)^2(1-\gamma_i)^5(1-\gamma_b)$

TABLE 3. S_3 Case 1

Again, these probabilities can be verified by noting the sum of the coefficients. In this case, the sum is $10 + 15 + 3$ or 28, which is expected (reasons are detailed in Section 2.3.1). Case 2 consists of a different set of probabilities which are depicted in Table 4.

Alive	Dead	Probability
$\gamma_i\gamma_i\gamma_i$	$\gamma_i\gamma_i\gamma_b\gamma_b\gamma_b$	$10(\gamma_i)^2(1-\gamma_i)^2(1-\gamma_b)^3$ +
$\gamma_b\gamma_b\gamma_b$	$\gamma_i\gamma_i\gamma_i\gamma_i\gamma_i$	$(\gamma_b)^3(1-\gamma_i)^5$ +
$\gamma_i\gamma_i\gamma_b$	$\gamma_i\gamma_i\gamma_i\gamma_b\gamma_b$	$30(\gamma_i)^2\gamma_b(1-\gamma_i)^3(1-\gamma_b)^2$ +
$\gamma_b\gamma_b\gamma_i$	$\gamma_b\gamma_i\gamma_i\gamma_i\gamma_i$	$15\gamma_i(\gamma_b)^2(1-\gamma_b)(1-\gamma_i)^4$

TABLE 4. S_3 Case 2

As with the previous region, the sum of probabilities in Table 3 is represented T_3 while that of Table 4 is T_4. Hence the total probability can be expressed as

$$P(S_3) = \gamma_i(\mathrm{T}_3) + \gamma_i(\mathrm{T}_4) + (1-\gamma_i)(\mathrm{T}_3) = \mathrm{T}_4 + \gamma_i(\mathrm{T}_3) \tag{2.10}$$

2.3.3. *Regular Inner Cells.* This subregion consists of all cells not a part of the other two subregions. The general neighborhood of these cells is made solely of cells with probability γ_i of being alive. Thus, it is not necessary to enumerate probabilities as with the other regions. In the first case, the main cell is alive and there are exactly two live neighbors. Hence, the probability would be given as $28(\gamma_i)^2(1-\gamma)^6$. Similarly, for Case 2, there are three live neighbors. This probability would be given as $56(\gamma_i)^3(1-\gamma)^5$. Combining these results in the same way as the previous subregions yields the following

$$\begin{aligned} P(Regular) &= 28(\gamma_i)^3(1-\gamma)^6 + 56(\gamma_i)^4(1-\gamma)^5 + 56(\gamma_i)^3(1-\gamma)^6 \\ &= 28(\gamma_i)^3(1-\gamma_i)^5(3-\gamma_i) \end{aligned} \tag{2.11}$$

2.3.4. *Combining Results for Inner Region.* In order to provide an overall probability that a cell in this region will be alive at time $t+1$, one must first weight the previously defined probabilities. To do so, it is first necessary to note that there are a total of $(s-4)^2$ inner cells. Moreover, there are four S_2 cells, $4(s-6)$ S_3 cells, and $(s-6)^2$ regular cells. Thus, the total probability is

$$P(Inner) = \frac{4P(S_2)}{(s-4)^2} + \frac{4(s-6)P(S_3)}{(s-4)^2} + \frac{(s-6)^2P(Regular)}{(s-4)^2} \tag{2.12}$$

Substituting the probabilities from equations 2.9, 2.10, and 2.11

$$\frac{4(\mathrm{T}_2 + \gamma_i\mathrm{T}_1) + 4(s-6)(\mathrm{T}_4 + \gamma_i\mathrm{T}_3) + (s-6)^2(28(\gamma_i)^3(1-\gamma_i)^5(3-\gamma_i))}{(s-4)^2} \tag{2.13}$$

This final expression denotes $P(Inner)$ which is equivalent to $P(C_{m,n}^{t+1} \mid C \in \text{Inner Region})$.

2.4. **Border Cell Behavior.** This region consists of the subregion S_1 and the regular border cells. The process of determining the probability that a cell in this region is alive at time $t+1$ is identical to that of Section 2.3, but results in a different final expression.

2.4.1. *S_1 Subregion.* This subregion consists of cells that are in the second layer and are adjacent to the corner cells. The general neighborhood of these cells resembles

$$\begin{pmatrix} 0 & 0 & \text{X} \\ \gamma_b & \gamma_b & \gamma_c \\ \gamma_i & \gamma_i & \gamma_b \end{pmatrix}$$

where the S_1 cell of interest is in the center and X denotes a cell whose state is fixed at both time t and $t+1$. If the cell is initially alive, the first case leading to an alive state at time $t+1$ can be given by

Alive	Dead	Probability
$\gamma_c\gamma_i$	$\gamma_i\gamma_b\gamma_b$	$2\gamma_c\gamma_i(1-\gamma_i)(1-\gamma_b)^2 +$
$\gamma_b\gamma_c$	$\gamma_i\gamma_i\gamma_b$	$2\gamma_b\gamma_c(1-\gamma_i)^2(1-\gamma_b) +$
$\gamma_b\gamma_b$	$\gamma_i\gamma_i\gamma_c$	$(\gamma_b)^2(1-\gamma_c)(1-\gamma_i)^2 +$
$\gamma_i\gamma_b$	$\gamma_b\gamma_c\gamma_i$	$4\gamma_i\gamma_b(1-\gamma_b)(1-\gamma_c)(1-\gamma_i) +$
$\gamma_i\gamma_i$	$\gamma_b\gamma_b\gamma_c$	$(\gamma_i)^2(1-\gamma_b)^2(1-\gamma_c)$

TABLE 5. S_1 Case 1

The sum should equal $\binom{5}{2}$ which it does $(2+2+1+4+1)$. Note that in this case, the state of three neighbors is fixed. Case 2 is depicted in Table 6.

Alive	Dead	Probability
$\gamma_c\gamma_i\gamma_i$	$\gamma_b\gamma_b$	$\gamma_c(\gamma_i)^2(1-\gamma_b)^2 +$
$\gamma_b\gamma_c\gamma_i$	$\gamma_i\gamma_b$	$4\gamma_b\gamma_c\gamma_i(1-\gamma_i)(1-\gamma_b) +$
$\gamma_b\gamma_b\gamma_c$	$\gamma_i\gamma_i$	$(\gamma_b)^2\gamma_c(1-\gamma_i)^2 +$
$\gamma_i\gamma_b\gamma_b$	$\gamma_c\gamma_i$	$2\gamma_i(\gamma_b)^2(1-\gamma_c)(1-\gamma_i) +$
$\gamma_i\gamma_i\gamma_b$	$\gamma_b\gamma_c$	$2(\gamma_i)^2\gamma_b(1-\gamma_b)(1-\gamma_c)$

TABLE 6. S_1 Case 2

If T_5 denotes the sum of probabilities in Table 5 and T_6, the sum in Table 6, then the total probability that a cell in this region is alive in the subsequent generation can be expressed as (derived from previous equations)

$$P(S_1) = \text{T}_6 + \gamma_b(\text{T}_5) \tag{2.14}$$

2.4.2. *Regular Border Cells.* These cells are those that are located within the border region but are not within the S_1 subregion. The general neighborhood is

$$\begin{pmatrix} 0 & 0 & 0 \\ \gamma_b & \gamma_b & \gamma_b \\ \gamma_i & \gamma_i & \gamma_i \end{pmatrix}$$

If the cell is initially alive, there must be two or three live neighbors. In Case 1, there are two live neighbors. The probability that a given cell is alive at $t+1$ due to the conditions for Case 1 is detailed in Table 7.

Alive	Dead	Probability
$\gamma_b\gamma_b$	$\gamma_i\gamma_i\gamma_i$	$(\gamma_b)^2(1-\gamma_i)^3 +$
$\gamma_i\gamma_b$	$\gamma_i\gamma_b\gamma_i$	$6\gamma_i\gamma_b(1-\gamma_b)(1-\gamma_i)^2 +$
$\gamma_i\gamma_i$	$\gamma_i\gamma_b\gamma_b$	$3(\gamma_i)^2(1-\gamma_i)(1-\gamma_b)^2$

TABLE 7. Regular Border Case 1

Note that the sum of the coefficients is 10, as expected. The same applies to Case 2 which is depicted in Table 8.

Alive	Dead	Probability
$\gamma_b\gamma_b\gamma_i$	$\gamma_i\gamma_i$	$3(\gamma_b)^2\gamma_i(1-\gamma_i)^2 +$
$\gamma_b\gamma_i\gamma_i$	$\gamma_b\gamma_i$	$6\gamma_b(\gamma_i)^2(1-\gamma_b)(1-\gamma_i) +$
$\gamma_i\gamma_i\gamma_i$	$\gamma_b\gamma_b$	$(\gamma_i)^3(1-\gamma_b)^2$

TABLE 8. Regular Border Case 2

If T_7 denotes the sum of probabilities in Table 7 and T_8, the sum in Table 8, then the total probability that a cell in this region is alive in the subsequent generation can be expressed as (derived from previous equations)

$$P(Regular) = \mathrm{T}_8 + \gamma_b(\mathrm{T}_7) \tag{2.15}$$

2.4.3. *Combining Results for Border Region.* Again, the probabilities must first be weighted before addition. Note that there are a total of $4(s-4)$ border cells. Moreover, there are eight cells in the S_1 subregion and hence $4(s-4)-8$ or $4s-24$ regular border cells. If $P(Border)$ is used to represent $P(C_{m,n}^{t+1} \,|C \in$ Border Region) then

$$P(Border) = \frac{8P(S_1)}{4(s-4)} + \frac{(4s-24)P(Regular)}{4(s-4)} \tag{2.16}$$

Substituting appropriate expressions for $P(S_1)$ and $P(Regular)$

$$P(Border) = \frac{8(\mathrm{T}_6 + \gamma_b\mathrm{T}_5) + (4s-24)(\mathrm{T}_8 + \gamma_b\mathrm{T}_7)}{4(s-4)} \tag{2.17}$$

2.5. **Corner Cell Behavior.** The corner cells are those that are located in the corners of the second outermost layer. The general neighborhood of a cell in the corner region is

$$\begin{pmatrix} 0 & \text{X} & \text{X} \\ \gamma_b & \gamma_c & \text{X} \\ \gamma_i & \gamma_b & 0 \end{pmatrix}$$

In the case that the cell is initially dead, there must be three live neighbors to change the cell's state. The probability for this circumstance is

$$(1-\gamma_c)(\gamma_b)^2\gamma_i \tag{2.18}$$

In the case that the cell is alive, it can have either two or three live neighbors to keep it in that state. The probability that it has three live neighbors can be expressed as

$$\gamma_c(\gamma_b)^2\gamma_i \tag{2.19}$$

while the probability that it has two live neighbors is expressed as

$$2\gamma_c(1-\gamma_b)\gamma_b\gamma_i + \gamma_c(\gamma_b)^2(1-\gamma_i) \tag{2.20}$$

Hence, the probability that a randomly chosen corner cell will be alive at $t+1$ can be expressed as a sum of the probabilities of each previous case's occurrence. If $P(Corner)$ denotes $P(C_{m,n}^{t+1} \mid C \in$ Corner Region), then

$$\begin{aligned} P(Corner) = {} & (1-\gamma_c)(\gamma_b)^2\gamma_i + \\ & \gamma_c(\gamma_b)^2\gamma_i + 2\gamma_c(1-\gamma_b)\gamma_b\gamma_i + \gamma_c(\gamma_b)^2(1-\gamma_i) \end{aligned} \tag{2.21}$$

2.6. **Outer Cell Behavior.** The outer cells are those that are located in the outermost layer. These cells are dead at time t but have the potential to change states. They can be divided into cells that have a corner cell in their neighborhood (S_0 subregion) and those that do not.

2.6.1. *S_0 Subregion.* The general structure of a cell in this subregion resembles the following

$$\begin{pmatrix} \text{X} & \text{X} & \text{X} \\ 0 & 0 & 0 \\ \gamma_c & \gamma_b & \gamma_b \end{pmatrix}$$

Because the center cell is already in the dead state, it must have precisely three live neighbors to become alive in the subsequent generation. The probability of this occurring can be given as

$$P(S_0) = \gamma_c(\gamma_b)^2 \tag{2.22}$$

2.6.2. *Regular Outer Cells.* These cells differ in that they do not have a corner cell in their neighborhood. Thus, the probability that one of these cells will be alive at $t+1$ can be expressed as

$$P(Regular) = (\gamma_b)^3 \tag{2.23}$$

2.6.3. *Combining Results for Outer Cells.* For any given automaton, there are eight cells within the S_0 subregion (two for every corner cell) and a total of $s^2 - 12$ outer cells. Hence the following

weightings would be applied to yield the probability that a cell in the outer layer is alive at time $t+1$

$$P(Outer) = \frac{8\gamma_c(\gamma_b)^2 + (s^2-20)(\gamma_b)^3}{s^2-12} \tag{2.24}$$

2.7. **Extending Probabilities.** Thus far, the calculations have concerned themselves with finding the probability that a randomly chosen cell within a certain region will become alive or remain alive at $t+1$. However, such probabilities do not allow one to predict the behavior of a group of cells within a region. First, it is necessary to distinguish the individual probabilities to avoid future confusion.

Definition 2.25. Each probability is assigned a variable $\alpha_{c,b,i,o}$ based on whether the variable refers to the probability regarding the corner, border, inner, or outer region. In other words,

- $P(Inner) = \alpha_i$
- $P(Corner) = \alpha_c$
- $P(Border) = \alpha_b$
- $P(Outer) = \alpha_o$

The use of α without a subscript can denote any one of these four variables.

Also it is important to distinguish between the number of live cells at time t versus time $t+1$.

Definition 2.26. A superscript t denotes the start generation of the CA. For example, the number of initial live cells in the corner region is denoted L_c^t.

This section concerns itself with finding $P(L^{t+1} < L^t)$ which is the probability that the number of live cells decreases in a given region over one generation. In general,

$$P(L^{t+1} < L^t) = P(L^{t+1} = L^t - 1) + P(L^{t+1} = L^t - 2) + ... + P(L^{t+1} = 0) \tag{2.27}$$

To find the probability of one of these terms, the α variable is applied to each cell in the region. This probability is binomial because each cell is either dead or alive and there is a fixed number of successes. If η represents the number of cells in a region, then the following holds

$$P(L^{t+1} = L^t - 1) = \binom{\eta}{L^t - 1}\alpha^{L^t-1}(1-\alpha)^{\eta - L^t+1} \tag{2.28}$$

This probability can be applied to the expression in equation 2.27. If the number 1 in the equation above is replaced with a general variable which takes on all values from 1 to L^t (note that the total number of terms in 2.27 equals L^t), then the summation can be accurately described. Hence,

$$P(L^{t+1} < L^t) = \sum_{n=1}^{L^t} \binom{\eta}{L^t - n}\alpha^{L^t-n}(1-\alpha)^{\eta - L^t+n} \tag{2.29}$$

This sum is applied to each region individually. Such sums are critical to the simulations conducted in Section 3; however, since each sum is essentially in the same format as equation 2.29, each is not calculated explicitly.

3. Simulation

A simulation program was written to determine the accuracy of the probabilistic model. The main part of the program looped in order to provide the desired amount of data. Probabilities were calculated for each individual region. Then, the program predicted whether the number of live cells would decrease in the next generation. For comparative purposes, the program also made pseudorandom guesses. The program would output both the success rate of the pseudorandom guessing and of the model. The code for the simulation program has been stored in a GitHub repository along with the data from the simulation and the statistical testing of the data.[1] More qualitative details regarding this program are located in Appendix A.

3.1. **Results.** The results from the simulations are depicted in Figure 3. In all regions, the probabilistic model outperformed the random guessing. The probabilistic model for the border region performed the worst out of the group. This variation can be explained by noting that the border region has neighbors consisting of all regions (disregarding subregions). This observation coupled with the size of the border region suggest that it may be slightly harder to predict. However, the variation between the border, inner, corner, and outer layers is relatively small.

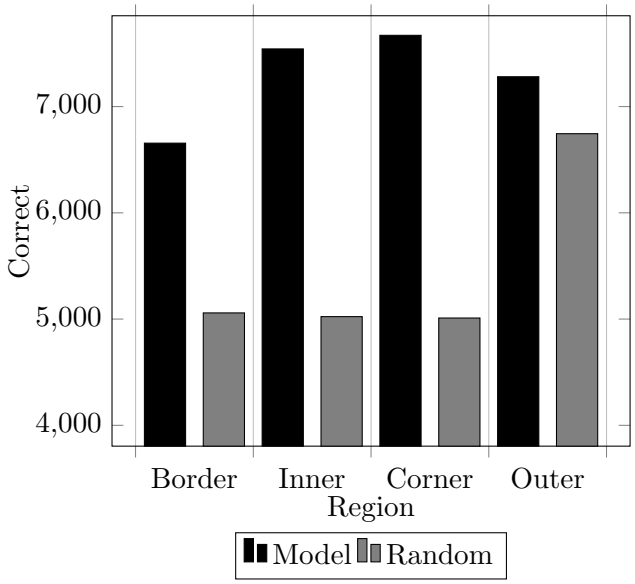


Figure 3. *A graph depicting results. This figure is referenced in the results section. The y-axis denotes the number of correct guesses. "Model" indicates bars that denote the number of correct guesses made by the model (same for "random")*

Also, the random guessing for the outer region outperformed the random guessing for the other regions. This observation originates from the nature of the outer layer. Since the probabilistic model predicts the probability that the states of the cells in the outer layer remain constant, it accounts for less possibilities than the other regions. Hence, the random guessing will perform better since there is less fluctuation in the states of outer layer cells from time t to $t + 1$. The corner region likely performed well because it is the smallest region and the inner region because it

[1]This can be found at `https://github.com/cakoch10/Patterns_in_Game_of_Life`

is more isolated than the others. The average success rate of the probabilistic model in relation to the CAs as a whole was 73%, further suggesting that the model did in fact predict the behavior of the system. However, to confidently demonstrate this notion, it is necessary to demonstrate that the random data is statistically different from the model data. Such statistical difference can be identified by the chi square test.

The observed and expected values are summarized in Table 9. The essential question then regards whether these two sets of data are statistically different. If the answer to this question is no, then it can be concluded that the model did not succeed in predicting more accurate results than that of random guessing. However, if the answer is yes, then the model did turn out to be successful. The question can be answered confidently by performing a chi square test. To do so, a set of expected values are calculated.

Table 9. Summations: Observed and Expected

	Observed		Expected	
	Model	Random	Model	Random
Border	6656	5058	6697.29	5016.70
Inner	7543	5023	7184.41	5381.59
Corner	7671	5010	7250.16	5430.84
Outer	7281	6745	8019.14	6006.86

The chi square test is conducted in Excel from these two sets of data using the *chitest()* function. This function compares two data sets and uses the chi square distribution to output a p value. Specifically, this function performs the chi square test for independence and returns the probability that the variation between the sets is a result of chance. The p value after conducting this test is 1.17×10^{-55} or approximately 0. Because this value is significantly less than 0.05 (the standard basis for comparison), it can be concluded with confidence that the differences between the model and random guessing are a deliberate result of probabilistic prediction.

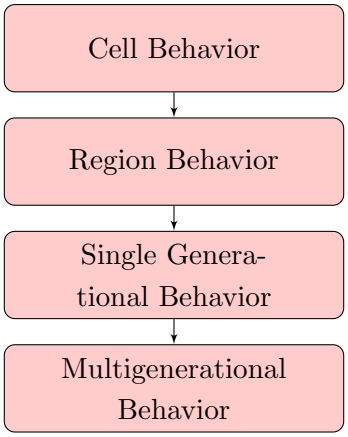
4. Discussion

Previous research studies have investigated the predictability of cellular automata in one dimension [9, 10] but few have investigated such in two dimensions, and none have done so with the Game of Life. And though there are many ways in which these results can be extended, it can be argued that the results thus presented are significant in relation to certain past studies. The Game of Life has been described as exhibiting Class IV behavior [7] which is characterized by Wolfram [11] as involving "a mixture of order and randomness: localized structures are produced which on their own are fairly simple, but these structures move around and interact with each other in very complicated ways." As such, many have withheld the consensus that GoL is largely unpredictable. As Solomon [3] described it "[Game of] Life cannot be decided." However, on the basis that GoL is partially random, one cannot argue that it is wholly unpredictable. In fact, this paper's results have suggested the contrary. If the automata were unpredictable, it would seem that the results from the probabilistic model would not differ as much from the random guessing — even though these results are collected from a relatively small automata over one generation.

Hence, these results can be used to form the basis for future work in establishing a probabilistic model to predict the extended behavior of an automaton.

The study is limited in some aspects, and it is important to address these limitations and how they relate to the conclusions. The scope of the study is limited by the size of the automata and the length of time over which an automaton is observed (i.e. one generation). Thus, the study can be improved by extending the model over multiple generations and perhaps testing larger systems. Nonetheless, the immediate effects of the limited scope of the simulation are reduced by the application of the model to the automata in their entirety. In other words, the model does not only apply to "localized structures" but to the whole system. Also, the model produced results that differed considerably from the random guessing, suggesting that if the model were applied to multiple generations it would still be useful in predicting behavior.

This flow chart below demonstrates the complexity of cellular automata behavior from least to greatest. At the most fundamental level, a CA contains individual cells which behave themselves according to the transition function. The cells often form distinctive structures or regions (much in the way human cells form tissue) which can then behave as a singular unit. These units in turn behave on two separate levels. The simplest level consists of the behavior of these cellular groups over one transition or generation. Then, as the groups of cells change over subsequent generations, the automaton exhibits multigenerational behavior.



This paper has focused on the first three levels. Studying the fourth level would require a revision of the model and an updated simulation program. Although the model loses accuracy after progressing from one level to the next, it seems that this accuracy loss is not significant (based on the simulation results from studying the first three levels). Indeed, much accuracy would be lost in moving to multigenerational behavior, but this loss may not prevent statistically significant predictions.

5. Conclusion

To formally study the 2-dimensional CA known as Conway's Game of Life, a formal definition of the transition rule had to be provided. In doing so, one could establish several conventions which were utilized later in the paper. After describing the GoL, the paper shifted into developing the model to predict CA behavior. The methods section primarily involved defining the information

on which the model relied and using this information to calculate the probability of every scenario which could lead to a desired outcome. This approach can be applied to practically any CA, including those in higher dimensions. It is also important to note that the model would not be as effective with larger automata since doing so would result in a decrease of information. The loss of information can be compensated for by allowing the regions to possess dynamic characteristics. For example, each layer (other than the outermost and two innermost layers) could have its own border region and corner region. Then, the model could account for the size of the automata.

In short, the project answered the research question and challenged previously withheld definitions. The statistical testing demonstrated the CAs in GoL are predictable with a relatively high degree of accuracy. Moreover, the simulation program can be easily modified for testing future models and has been made open source via GitHub. Ultimately, it is through the answering of these questions and problems that one can better grasp the underlying nature of CAs and similar discrete models of computation.

References

[1] Toffoli, T., & Margolus, N. (1987). Cellular automata machines: a new environment for modeling. MIT press.

[2] Weisstein, Eric W. (2004). Cellular Automaton. 2014. Web site. Accessed: September, 2015. Wolfram MathWorld. http://mathworld.wolfram.com/CellularAutomaton.html.

[3] Solomon, R. (2008). The Little Book of Mathematical Principles, Theories, & Things. Metro Books.

[4] Sarkar, P. (2000). A brief history of cellular automata. ACM Computing Surveys (CSUR), 32(1), 80-107.

[5] Mitchell, M. (1996). Computation in cellular automata: A selected review.Nonstandard Computation, 95-140.

[6] Martnez, G. J., Seck-Tuoh-Mora, J. C., & Zenil, H. (2013). Wolframs classification and computation in cellular automata Classes III and IV. In Irreducibility and Computational Equivalence (pp. 237-259). Springer Berlin Heidelberg.

[7] Rennard, J. P. (2002). Implementation of logical functions in the Game of Life. In Collision-based computing (pp. 491-512). Springer London.

[8] Koch, C. (2015). Regularity within Conway's Game of Life. doi: 10.13140 /RG.2.1.3940.5286

[9] Agapie, A., Andreica, A., Chira, C., & Giuclea, M. (2014). Predictability in Cellular Automata.

[10] Agapie, A., Hns, R., & Mhlenbein, H. (2004). Markov chain analysis for one-dimensional asynchronous cellular automata. Methodology and Computing in Applied Probability, 6(2), 181-201.

[11] Wolfram, S. (2002). A new kind of science (Vol. 5). Champaign: Wolfram media.

Appendix A. Qualitative Description of the Simulation Program

The simulation program was developed by the author and written in C++. During each iteration of the primary loop, an 8×8 cellular automaton was constructed and each cell was randomly assigned an on/off state (this process excluded the outermost layer). A larger grid was not tested because the computation time required for a larger grid was exceptionally high. This growth can be attributed to the time taken to compute combinations which increases factorially.

The fixed trials of pseudorandom guessing which alternated between true and false guesses (in other words, the computer would guess that the number of live cells decreases for each region every other iteration) ensured an even distribution of positive versus negative guesses. The total number of correct guesses for each guessing technique is tallied as the simulation progresses. After 100 iterations, the program outputs the desired values to a file and the simulations start again.

Each iteration of the program outputs eight values. Four values represent the number of correct guesses from the probabilistic model while the other four represent that from the random guesses. These values are copied into an Excel file in which data synthesis can take place. Overall, the program is executed 100 times resulting in a total of 10,000 simulations. The collection of a large amount of data allowed for more confidence in the results when comparing the random and probabilistic guessing methods.