A Classification of One-Dimensional Cellular Automata Using Infinite Computations

Louis D'Alotto
Department of Mathematics and Computer Science
York College/City University of New York
Jamaica, New York 11451
and
The Doctoral Program in Computer Science
CUNY Graduate Center
ldalotto@gc.cuny.edu

Abstract

This paper proposes an application of the Infinite Unit Axiom and *grossone*, introduced by Yaroslav Sergeyev (see [15] - [19]), to classify one-dimensional cellular automata whereby each class corresponds to a different and distinct dynamical behavior. The forward dynamics of a cellular automaton map are studied via defined classes. Using these classes, along with the Infinite Unit Axiom and grossone, the number of configurations that equal those of a given configuration, in some finite central window, under an automaton map can now be computed. Hence a classification scheme for one-dimensional cellular automata is developed, whereby determination in a particular class is dependent on the number of elements in their respective forward iteration classes.

Keywords: Cellular automata, Infinite Unit Axiom, grossone, metric, dynamical systems.

1. Introduction

Cellular automata, originally developed by von Neuman and Ulam in the 1940's to model biological systems, are discrete dynamical systems that are known for their strong modeling and self-organizational properties (for examples of some modeling properties see [3], [5], [22], [23], [24], and [26]). Cellular automata are defined on an infinite lattice and can be defined for all dimensions. In the one-dimensional case the integer lattice $\mathbb{Z}$ is used. In the two-dimensional case, $\mathbb{Z} \times \mathbb{Z}$. An example of a two-dimensional cellular automata is John Conway's ever popular "Game of Life"[1]. Probably the most interesting aspect about

[1] For a complete description (including some of the more interesting structures that emerge) of "The Game of Life" see [1] Chapter 25.

cellular automata is that which seems to conflict our physical systems. While physical systems tend to maximal entropy, even starting with complete disorder, forward evolution of cellular automata can generate highly organized structure.

As with all dynamical systems, it is important and interesting to understand their long term or evolutionary behavior. Hence it makes sense to develop a classification of a system based on its dynamical behavior. The concept of classifying cellular automata was initiated by Stephen Wolfram in the early 1980's, see [25] and [26]. Through numerous computer simulations, Wolfram noticed that if an initial configuration (sequence) was chosen at random the probability is high that a cellular automaton rule will fall within one of four classes.

The examples to follow are referred to by a rule numbering system developed by Wolfram, see [25] and [27]. In [27], one-dimensional cellular automata are partitioned into four classes depending on their dynamical behavior, see figure 1 (Totalistic Rule 36) for an example of a Wolfram class 1 cellular automaton. Class 1 are the least chaotic, indeed Wolfram labeled these as automata that evolve to a uniform state. Figure 2 (Totalistic Rule 24) is an example of a Wolfram class 2 cellular automaton. Wolfram described the evolution of automata of this class as leading to simple stable or periodic structures. Figure 3 (Totalistic Rule 12) is an example of a Wolfram class 3 cellular automaton. In these automata the dynamical behavior is more complicated, however triangles and other small structures are seen to emerge in the form of a chaotic pattern. Figure 4 (Totalistic Rule 20) is an example of a class 4 cellular automaton. Wolfram labeled class 4 the most chaotic whereby localized complex structures emerge. In these figures it can be seen that a cellular automaton map starts with a given (random) initial configuration and evolves in a downward direction upon forward iterations (evolution) of the cellular automaton rule. It is interesting to note the two persisting structures that emerge in the figure 4 automaton. The structure on the left evolves straight down, while the structure on the right evolves on a diagonal. Eventually the one on the right will 'crash' into the structure on the left and they will either annihilate each other or produce another persisting structure.

A later and more rigorous classification scheme for one-dimensional cellular automata, see [7], was developed by Robert Gilman. Here a probabilistic/measure theoretic classification scheme was developed based on the probability of choosing a configuration that will stay arbitrarily close to a given initial configuration under forward iteration (evolution) [2]. Gilman's classification partitions the cellular automata rules into three classes. Class A is the class of equicontinuous automata, whereby there is an open disk of configurations that stay arbitrarily close to the given initial configuration. Automata in class B conform to a stochastic analog of equicontinuity. Automata in this class have the property that the probability is positive that one can find (at random) another

[2] Gilman's classification is based on choosing an infinite product probability measure on the space of cellular automata.

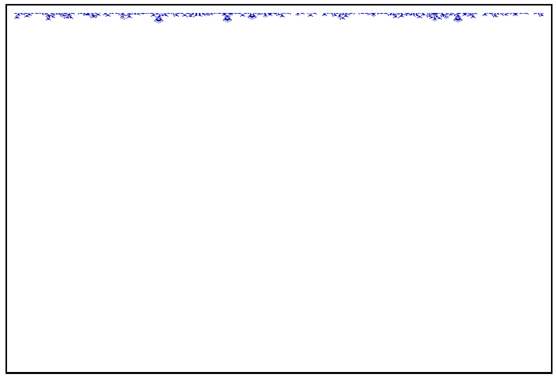

Figure 1: Wolfram Class 1 Cellular Automaton (Totalistic Rule 36)

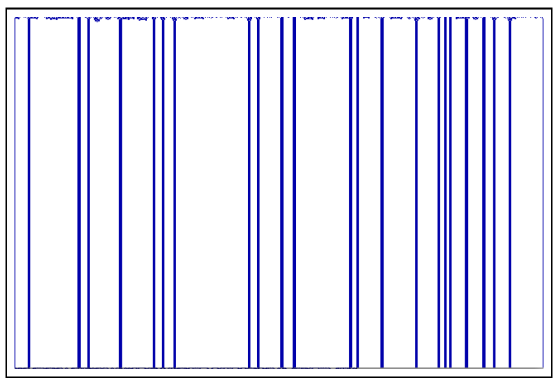

Figure 2: Wolfram Class 2 Cellular Automaton (Totalistic Rule 24)

configuration that can stay arbitrary close to an initial under forward evolution. Automata in class C have the property that the probability of finding another configuration that stays arbitrarily close to the initial under forward iterations is 0. Owing to the fact that the lens of measure theory does not distinguish between countably infinite and uncountably infinite (in the Cantor sense) sets, it is noted that automata in all the classes have some indistinguishable dynamic similarities. For instance, in both Gilman classes A and B there are an infinite amount of configurations that can stay arbitrarily close to a given initial configuration under forward evolution. In the classification presented herein, using the infinite unit axiom of Sergeyev, see [15] - [19], this similarity is overcome by actually having a numeric representation for the number of configurations in each class that equal (or match) an initial configuration under forward evolution. Thus making the classes more distinguishable.

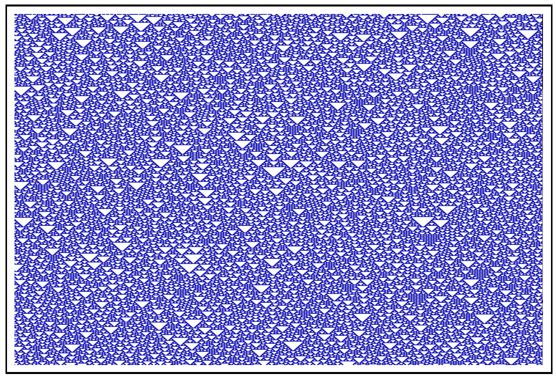

Figure 3: Wolfram Class 3 Cellular Automaton (Totalistic Rule 12)

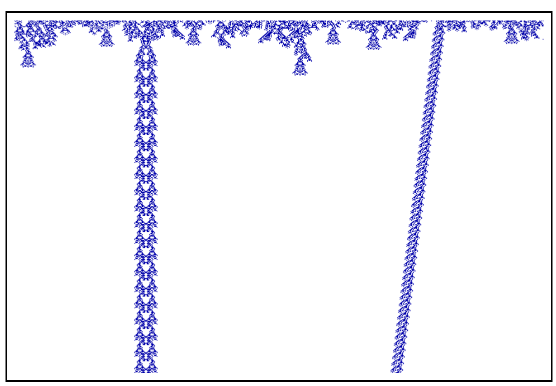

Figure 4: Wolfram Class 4 Cellular Automaton (Totalistic Rule 20)

2. The Infinite Unit Axiom

The new methodology of computation, initiated by Sergeyev (see [15] - [18]), provides a new way of computing with infinities and infinitesimals. Indeed, Sergeyev uses concepts and observations from physics (and other sciences) to set the basis for this new methodology. This basis is philosophically founded on three postulates:

Postulate 1. *"We postulate the existence of infinite and infinitesimal objects but accept that human beings and machines are able to execute only a finite number of operations."*

Postulate 2. *"We shall not tell what are the mathematical objects we deal with. Instead, we shall construct more powerful tools that will allow us to improve our capacities to observe and to describe properties of mathematical objects."*

Postulate 3. *"We adopt the principle: 'The part is less than the whole', and apply it to all numbers, be they finite, infinite, or infinitesimal, and to all sets and processes, finite or infinite."*[3]

These postulates set the basis for a new way of looking at and measuring mathematical objects. The postulates are actually important philosophical realizations that we live in a finite world (i.e. that we, and machines, are incapable of infinite or infinitesimal computations). All the postulates are important in the application presented herein, however Postulate 1 has a ready illustration. In this paper we will deal with counting and hence representing infinite quantities and measuring (by way of a metric) extremely small or infinitesimal quantities. Postulate 2 also has a ready consequence herein. In the classification presented in this paper, more powerful numeral representations will be constructed that actually improve our capacity to observe, and describe, mathematical objects and quantities. Postulate 3 culminates in the actual classification scheme presented in this paper. Indeed, the cellular automata classification presented here is developed by partitioning the entire space into three classes. It is quite interesting to notice that the order of Postulates 1 - 3 seem to dictate the exposition and order of results of this paper. It must be noted that the Postulates should not be conceived as axioms in this new axiomatic system but rather set the methodological basis for the new system[4].

The *Infinite Unit Axiom* is formally stated in three parts below and are assumed throughout this paper. This axiom involves the idea of an infinite unit from finite to infinite. The infinite unit of measure is expressed by the numeral ①, called *grossone*, and represents the number of elements in the set $\mathbb{N}$ of natural numbers.

1. *Infinity*: For any finite natural number n, it follows that $n <$ ①.
2. *Identity*: The following involve the identity elements 0 and 1

[3] See [18], section 2, for a complete discussion.
[4] In [11], G. Lolli gives a clear distinction and discussion of the Postulates and Axioms.

(a) $0 \cdot ① = ① \cdot 0 = 0$

(b) $① - ① = 0$

(c) $\frac{①}{①} = 1$

(d) $①^0 = 1$

(e) $1^{①} = 1$

3. *Divisibility*: For any finite natural number n, the numbers

$$①, \frac{①}{2}, \frac{①}{3}, ..., \frac{①}{n}, ...$$

are the number of elements of the n^{th} part of $\mathbb{N}$[5].

An important aspect of ① that will be used extensively in this paper is the numeric representation of $①^{-i}$ for $i > 0$ (note that i can be infinite as well). These numbers are called *infinitesimals*. The simplest infinitesimal is $①^{-1} = \frac{1}{①}$. It is noted that $①^{-1}$ is the multiplicative inverse element for ①. That is, $①^{-1} \cdot ① = ① \cdot ①^{-1} = 1$. It is also important (and essential in this paper) to note that all infinitesimals are not equal to 0. In particular, $\frac{1}{①} > 0$[6]

As noted above, the set of natural numbers is represented by

$$\mathbb{N} = \{1, 2, 3, ..., ① - 2, ① - 1, ①\}$$

and the set of integers, with the new grossone methodology, is represented by

$$\mathbb{Z} = \{-①, -① + 1, -① + 2, ..., -3, -2, -1, 0, 1, 2, 3, ..., ① - 2, ① - 1, ①\}$$

However, since we will be working with the set $S^{\mathbb{Z}}$ as the domain of definition for cellular automata maps, we will need to make use of the set of extended natural numbers by applying the arithmetical operations to ①

$$\hat{\mathbb{N}} = \{1, 2, 3, ..., ① - 2, ① - 1, ①, ① + 1, ..., ①^n, ..., 2^{①}, ..., ①^{①}, ...\}$$

---

[5]In [18], Sergeyev formally presents the divisibility axiom as saying for any finite natural number n sets $\mathbb{N}_{k,n},\ 1 \leq k \leq n$, being the nth parts of the set $\mathbb{N}$, have the same number of elements indicated by the numeral $\frac{①}{n}$ where

$$\mathbb{N}_{k,n} = \{k, k+n, k+2n, k+3n, ...\},\ 1 \leq k \leq n,\ \bigcup_{k=1}^{n} \mathbb{N}_{k,n} = \mathbb{N}$$

and illustrates this with examples of the odd and even natural numbers.

[6]In [15] and [17] this is also shown as a limiting process. That is,

$$\lim_{n \to ①} \frac{1}{n} = \frac{1}{①} \neq 0.$$

Where

$$1 < 2 < 3 < ... < ① - 1 < ① < ① + 1 <$$
$$... < ①^{10} < ... < 2^{①} < ... < ①^{①} < ...$$

and hence the infinitesimals

$$0 < ... < \frac{1}{①^{①}} < ... < \frac{1}{2^{①}} < ... < \frac{1}{①^{10}} < ... < \frac{1}{①} < ...$$

The extended natural numbers will be used to represent the number of elements in a set and their reciprocals used for infinitesimal quantities. The sequence of forward iterates of an automaton map will only go up to ①, as the maximum number of elements in a sequence cannot be more than grossone[7]. Cellular automata are important models of computation, namely parallel computation. However, the theory of grossone has already been successfully applied to studying other models of computation, see [20] and [21].

3. Cellular Automata

Let S is an alphabet of size $s = |S|$ such that $s \geq 2$ and let $X = S^{\mathbb{Z}}$, i.e. the set of all maps from the lattice $\mathbb{Z}$ to the set S. That is, for $x \in X$, $x : \mathbb{Z} \to S$. One-Dimensional cellular automata (hence just called cellular automata from now on) are induced by arbitrary local maps:

$$F : S^{(2r+1)} \longrightarrow S$$

These are usually called local rules or block maps in the literature, see [7] and [8]. The value $r \in \mathbb{N}_0$ is called the range of the map. The automaton map f induced by F is defined by $f(x) = y$ with

$$y(i) = F[x(i-r), ..., x(i+r)]$$

To illustrate the importance of discrete time steps in the forward evolution of the automaton, we will use the following formula where t represents time.

$$y(i)_{t+1} = F[x(i-r)_t, ..., x(i+r)_t]$$

The study of dynamical systems, in this case discrete dynamical systems, endeavors to understand the forward evolution (or forward iterations) of the system map and in this case the automaton rule. In this paper $\mathbb{N}_0$ is used to represent the set of $\mathbb{N} \cup \{0\}$. $f^t(x)$ is used to represent the t^{th} iterate of the automaton map f. That is,

$$f^t(x) = f \circ f \circ f \cdots \circ f(x)$$

[7] In [18], Theorem 5.1, Sergeyev shows, using the new methodology, that the number of elements of any infinite sequence is less or equal to ①. It is also mentioned in this reference, that a subsequence, being a sequence from which some of the elements have been removed, can be an infinite sequence having its number of terms less than grossone.

where $0 \leq t \leq ①$.

The restriction of $x \in X$ to a non-empty interval $[i, j]$ of $\mathbb{Z}$, where $-① \leq i \leq j \leq ①$ is called a *word*. Words are written $x[i, j]$. The length of a word $w = x[i, j]$ is $|w| = j - i + 1$. It is important to note that, using ①, words (or the length of a word) can be infinite, however cannot have an endpoint greater than ① (nor less than $-①$). Also, for any $a \in S$, define $x_a \in X$ by $x(i) = a$, for $i \in \mathbb{Z}$. Words of infinite length are mentioned in [7] but not much is done with them as the metric used cannot handle infinite computations. The introduction of the Infinite Unit Axiom and grossone make infinite, and infinitesimal, computations possible. As stated in the abstract and introduction, this paper is concerned with the development of a classification of one-dimensional cellular automata. However, before a classification scheme can be presented, it is useful to first define a metric on the space of cellular automata.

Definition 3.1. *Let*

$$x \wedge y = \begin{cases} x & if \;\; x = y \\ * & if \;\; x(0) \neq y(0) \; or \; x(0) = * \\ x(-n)...x(0)...x(n) & if \;\; x(i) = y(i) \; \forall i \in [-n, n] \;\; and \; * \, outside \end{cases}$$

With the introduction of the Infinite Unit Axiom, $-n$ can be infinite and equal $-① + k$ for some finite integer $k < 0$, similarly n can equal $① - k$ for some finite integer $k > 0$ (note that if $k = 0$, then $x = y$). Hence computations on infinite configurations are allowed. Thus, $x \wedge y$ is the place where two sequences agree on the largest symmetric window around 0 and is $*$ valued outside. Let λ be an arbitrary real-valued function defined on the alphabet S and taking values in the open interval $(0, 1)$, i.e. $\lambda : S \to (0, 1)$ where $\lambda_i = \lambda(x(i))$, hence $0 < \lambda_i < 1$. Keeping in line with the metric development in [4], we need the following definition[8].

Definition 3.2.

$$F(x \wedge y) = \begin{cases} 1 & if \;\; x \wedge y = * \\ \prod_{-n}^{n} \lambda_i & if \;\; x \wedge y = ... * * * \, x(-n)...x(0)...x(n) * * * ... \end{cases}$$

We now form the following metric on the space of bi-infinite sequences:

Definition 3.3.

$$d(x, y) = \begin{cases} 0 & if \;\; x = y \\ F(x \wedge y) & otherwise \end{cases}$$

It is a simple exercise to check that this is indeed a metric and satisfies the nonarchimedian or ultra metric inequality

$$d(x, y) \leq \max\{d(x, z), d(z, y)\}.$$

[8]The function $F : X \to (0, 1]$ is called an evaluation function, see [4] for a more formal development of a general nonarchimedean metric and [14] for a complete reference on nonarchimedean analysis.

The great advantage of this metric and grossone lies in the fact that we can now use configurations (and do computations) that agree, and hence are infinitesimally close to each other, on infinite intervals, as the next example shows.

Example 3.1. *Given* $S = \{0, 1\}$*, let*

$$x = \overbrace{1}^{-①} ...111\langle 1\rangle 111... \overbrace{1}^{①}$$

and

$$y = \overbrace{0}^{-①} \overbrace{0}^{-①+1} \overbrace{0}^{-①+2} 11...111\langle 1\rangle 111...1 \overbrace{1}^{①}$$

That is, x is the sequence of all 1's and y is the sequence of all 1's except for 3 zeros at the negative infinite indicated positions. In our examples, when not explicitly denoted, we will use the symbol $\langle\ \rangle$ to denote the zeroth place on the integer lattice. The sequences x and y agree completely on the right hand side, and don't agree at integral values $-①, -① + 1,$ and $-① + 2$. Hence,

$$x \wedge y = x(-① + 3), x(-① + 4), ..., x(-1), x(0), x(1), ..., x(① - 4), x(① - 3).$$

and the evaluation function becomes

$$F(x \wedge y) = \lambda_1^{(2① - 6)}.$$

If $\lambda_1 = \frac{1}{2}$, for example, then the distance becomes

$$d(x, y) = \frac{1}{2^{2① - 6}}\ .$$

Note that the value of λ_0 was not needed since these configurations have only 1's where they agree, however the value should still be given and used when needed. Hence, the distance between the two points x and y is infinitesimal. Of course, as the following example shows, the above construction easily covers the finite distance case.

Example 3.2. *Again, using the binary alphabet* $S = \{0, 1\}$*, let*

$$x = ...1110\langle 1\rangle 0011...$$

and

$$y = ...1110\langle 1\rangle 0101...$$

Then

$$x \wedge y = ... * * * \, x(-1)x(0)x(1) * * * ...$$

That is, the sequences differ in the 2nd integral position and hence is $*$ valued outside the central places where they agree. Therefore, depending on the values of λ_0 and λ_1,

$$d(x, y) = \lambda_0 \cdot \lambda_1 \cdot \lambda_0$$

Note that, to compute their distance, we do not need to know the rest of these sequences past the 2nd and the −2nd integral positions, hence '...' is used to mean the sequences continue to the −① position on the left and the ① position on the right.

Under the usual product topology, a *cylinder* is a set $C(i, j, w) = \{x \in X | x[i, j] = w\}$, where $|w| = j - i + 1$. We define the open disk of radius ε around x to be $C_{[-n,n]}(x) = C(-n, n, x[-n, n])$. Here, it is important to note, $\varepsilon > 0$ and that ε can be infinitesimal. It should be clarified that ε must be computed with respect to the metric defined above but first with the respective values of λ chosen. As the following example illustrates.

Example 3.3. *Given the alphabet $S = \{0, 1\}$ and $\lambda_0 = \lambda_1 = 1/2$, then the disk centered at x and of radius $\varepsilon = 1/8$ is denoted by $C_{[-1,1]}(x)$*[9].

It should also be noted that being a nonarchimedean metric space, given any two open disks, either one contains the other or they intersect trivially, see [14] for a complete introduction to nonarchimedean spaces. In connecting the new methodology with traditional topological dynamics, the following definitions are classical in the theory of cellular automata and should be stated.

Definition 3.4. *For $\varepsilon > 0$ (note that ε can be infinitesimal) and $x \in S^{\mathbb{Z}}$, let $D(x, \varepsilon) = \{y | \ d(f^i(y), f^i(x)) < \varepsilon, \forall \ i \in \mathbb{N}_0\}$.*

Definition 3.5. *f is* **expansive** *if there is an $\varepsilon > 0$ such that for all x, $D(x, \varepsilon) = \{x\}$.*

Expansive cellular automata demonstrate a very strong form of sensitivity to initial conditions. Indeed, a cellular automata is called expansive if there exists a finite window such that, for all configurations, any differences will eventually propagate inside this window. This means that, for all configurations, say x and y with $x \neq y$, there will be a forward iterate n such that $f^n(x) \neq f^n(y)$. The left (or right) shift map is an example of an expansive automaton. Indeed, for all configurations sufficiently 'close' to each other, a small difference (no matter how far away) will eventually propagate inside the finite window. The important item to notice here is that $D(x, \varepsilon)$ is defined using the metric and hence requires a symmetric window around the zeroth place.

The following is a simple, but important, example of a cellular automaton of range $r = 1$. The evolutionary behavior of this automaton is clearly exhibited.

Example 3.4. *Let $S = \{0, 1\}$ and let f be the automaton induced by the local rule $F : S^3 \to S$ by $F(1, 1, 1) = 1$ and $F(a, b, c) = 0$ otherwise. If we apply forward iterations of the induced automaton map f, all sequences eventually go to the quiescent state of x_0, except for the initial sequence x_1 which remains constant*[10].

---

[9] We can also take the convention, once the λ values are fixed, to denote $C_{1/8}(x)$ as the disk of radius 1/8.

[10] This is an example of a Gilman class A automaton. For this and other examples, including Gilman class B automata, see [7]

In the previous example, given any finite word $x[i,j]$ with at least one element in the word not equal to 1, every configuration will eventually evolve, under forward iterations, to the quiescent state of x_0. Moreover, if we choose an open disk $C_{[-n,n]}(x)$ around that point, every configuration in the open disk will eventually evolve to x_0. Hence it is important to determine how many elements are in these open disks. Using Theorem 5.2 of [18][11], this was answered in [4]. Theorem 3.1 and its corollaries below show the answer.

Theorem 3.1. *Given the space $S^{\mathbb{Z}}$ of bi-infinite sequences, the number of elements $x \in S^{\mathbb{Z}}$ is equal to $|S|^{2①+1}$.*

Proof. See [4] for the proof. □

Corollary 3.1. *The open disk $C_{[-n,n]}(x)$ around x contains $|S|^{2(①-n)}$ elements.*

Proof. Follows directly from Theorem 3.1, see [4]. □

Corollary 3.2. *If there are $|S|^{2①-2n}$ elements in an open disk $C_{[-n,n]}(x)$ of X, then there are $|S|^{2①-2n} \cdot (|S|^{2n+1} - 1)$ elements in the complement of $C_{[-n,n]}(x)$*[12].

Proof. An open disk, $C_{[-n,n]}(x)$, must be centered and symmetric around the zeroth element. Hence, by Corollary 3.1, $C_{[-n,n]}(x)$ contains $|S|^{2(①-n)}$ elements. Since an open disk must be centered around a word of at least length one (the zeroth place), there are $|S|^{2n+1}-1$ other disks. The space is nonarchimedean and hence the open disks are either disjoint or one contains the other. Therefore there are $|S|^{2n+1} - 1$ other $C_{[-n,n]}(y)$ disks that are disjoint from $C_{[-n,n]}(x)$ and each one of these disks contain $|S|^{2①-2n}$ elements, any other disk will be contained in one of these. Being that there are $|S|^{2①+1}$ total elements in the space X, the number of elements in the complement of $C_{[-n,n]}(x)$ is (as the computation goes):

$$|S|^{2①+1} - |S|^{2①-2n} = \overbrace{|S|^{2①-2n} + ... + |S|^{2①-2n}}^{|S|^{2n+1}-1} = |S|^{2①-2n} \cdot (|S|^{2n+1} - 1).$$

□

Given the definitions and the previous corollaries, it is allowable to define an open disk of infinitesimal radius. A disk of infinitesimal radius is an open disk around a word of infinite length. For example, the disk $C_{[-①+2,①-2]}(x)$ has exactly $|S|^{2①-(①-2+①-2)} = |S|^4$ elements. As is seen, open disks of infinitesimal radius can have very few elements. It should be noted that Corollaries 3.1 and 3.2 also apply to disks of infinitesimal radius.

[11] In this reference it is shown the number of elements of the set $\mathbb{Z}$ is equal to 2①1. This notation represents $2① + 1$ in the new positional number system.

[12] M. Margenstern proves a general result, see [12] Proposition 1.

4. Classes of One-Dimensional Automata

To understand the dynamics of cellular automata it is necessary to study the forward iterates of configurations that equal or match those of a given configuration, call it "x", on a given interval of $\mathbb{Z}$. Here the relation $x \sim y$ iff $\forall i \in \mathbb{N}_0$, $(f^i(y))[m,n] = (f^i(x))[m,n]$ forms an equivalence relation with equivalence classes denoted by $B_{m,n}(x)$. That is,

$$B_{m,n}(x) = \{y \mid (f^i(y))[m,n] = (f^i(x))[m,n] \ \ \forall i \in \mathbb{N}_0\}.$$

$B_{m,n}(x)$ is the set of y for which $(f^i(y))[m,n] = (f^i(x))[m,n]$, for $m \leq 0 \leq n$, under forward iterations of the cellular automaton function. That is, $\forall i \in \mathbb{N}_0$. Recall, $(f^i(y))[m,n]$ represents words and that the cellular automaton function, f is first applied to the entire configuration x (or y), and then restricted to the interval $[m,n]$. Note that m can equal $-① + k$ and n can equal $① - k$, for some finite integer $k \geq 0$. In those cases the words are left-sided, right-sided or both sided infinite. Hence elements in the $B_{m,n}(x)$ classes will agree with, and so will their forward iterations, $x[m,n]$ and all forward iterations of $x[m,n]$ under the automaton map f. This will form the effect of an infinite vertical strip (column), not necessarily symmetric, around the central window.

The dynamical analysis of cellular automata presented herein is based on counting the number of elements in the entire domain space, X. Hence, in this section we will use ① to count the number of elements in the class $B_{m,n}(x)$ whose forward iterates match those of x in the central window and develop a simple classification of one-dimensional cellular automata based on this count. One-dimensional cellular automata rules are thus partitioned into three classes.

Definition 4.1. *Define the classes of one dimensional cellular automata, f, as follows:*

1. *$f \in \mathcal{A}$ if there is a $B_{m,n}(x)$ that contains at least $|S|^{2①-k}$ elements, for some finite integer $k \geq 0$.*
2. *$f \in \mathcal{B}$ if there is a $B_{m,n}(x)$ that contains at least $|S|^{\alpha①-k}$ elements, for some finite integer $k \geq 0$, $0 < \alpha < 2$ and α not infinitesimal, but f does not belong to class $\mathcal{A}$.*
3. *$f \in \mathcal{C}$ otherwise.*

Class $\mathcal{C}$ is the most chaotic class of automata. Indeed, in this class there may only be finitely many elements or simple infinitely many elements in any $B_{m,n}(x)$ class. Hence, beginning with an initial configuration, most other configurations will diverge away from the initial configuration. Automata in class $\mathcal{A}$ are the least chaotic and most elements will equal an initial configuration upon repeated applications (iterations) of the automata rule on the infinite strip. The following theorems show the relationship between an open disk and the number of configurations in a $B_{m,n}(x)$ class.

Theorem 4.1. *If there exists a $B_{m,n}(x)$, for cellular automaton f, that contains an open disk of non-infinitesimal radius, then $f \in \mathcal{A}$.*

Proof. If there is a $B_{m,n}(x)$, for cellular automaton f that contains an open disk $C_{[-n,n]}(x)$ of non-infinitesimal radius, then $C_{[-n,n]}(x)$ contains $|S|^{2(①-n)}$ elements. Therefore $B_{m,n}(x)$ contains at least $|S|^{2①-2n}$ elements. Take finite $k = 2n$ and by Definition 4.1 the theorem is proved. □

Theorem 4.2. *If $f \in \mathcal{A}$ then there exists a $B_{m,n}(x)$ class that contains an open disk.*

Proof. If $B_{m,n}(x)$ contains points y, then they must equal $x[m,n]$, and so must all forward iterations, on the central window $[m,n]$ that contains the 0^{th} place. Given $f \in \mathcal{A}$, there is a $B_{m,n}(x)$ that contains $|S|^{2①-k}$ elements, for some $k \geq 0$. Recalling $m \leq 0 \leq n$, if $|m| \geq |n|$, take $k = 2|m|$ and choose $C_{[m,-m]}(x)$. If $|m| < |n|$, take $k = 2|n|$ and choose $C_{[-n,n]}(x)$, and the theorem is proved. □

Hence, the dynamical behavior of automata in class $\mathcal{A}$ is determined by a finite amount of information on the initial configuration x. This is not true for the other classes. For example, the forward (or backwards) dynamics of the right or left shift map cannot be determined by a finite amount of information.

The remainder of this section is dedicated to some examples.

Example 4.1. *By Theorem 4.1, the automaton defined in Example 3.4 above is an example of a class $\mathcal{A}$ automaton.*

Example 4.2. *The left shift map, σ, is a cellular automaton of range* 1*, defined by $\sigma(x_i) = x_{i+1}$. i.e. the map that shifts all symbols of a configuration to the left, as illustrated below:*

$$\begin{array}{rcl} x & = & ...0\,1\,1\,1\,0\,0\,1\,1\,0\,1\,1\,\langle 1\rangle\,0\,1\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1... \\ \sigma(x) & = & ...1\,1\,1\,0\,0\,1\,1\,0\,1\,1\,1\,\langle 0\rangle\,1\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1\,... \\ \sigma^2(x) & = & ...1\,1\,0\,0\,1\,1\,0\,1\,1\,1\,0\,\langle 1\rangle\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1\;\;... \\ . & & \\ . & & \\ . & & \end{array}$$

Obviously all configurations $y \in B_{m,n}(x)$ would have to agree with x to the right, out to ① and at the zeroth place. Therefore there are at least $|S|^{①-k}$ elements, for some finite $k > 0$ and at most $|S|^{①}$ elements in $B_{m,n}(x)$, hence $\sigma \in \mathcal{B}$

It should be noted that, by definitions 3.4 and 3.5, the left (or right) shift automaton is expansive. However it must be noted that there are also an infinite, but representable quantity in the new methodology, amount of configurations, under the left or right shift, that will not propagate inside a finite window. These are precisely those that differ to the left of a finite window for the left shift and those that differ to the right of a finite window for a right shift. Hence it is shown that class $\mathcal{B}$ contains, in the classical sense, expansive automata.

5. Probabilities

In [25] Wolfram based his classification on the observation that if a configuration is chosen at random then the probability is high that the cellular automaton rule will lie in one of four classes. In [7] Gilman based the classification on the probability of finding another configuration that will stay arbitrarily close to the initial configuration. In this section we show and compute some of the probabilities, for the classification presented herein, of finding another configuration that stays arbitrarily close to an initial configuration. Here again the advantage of using the Infinite Unit Axiom is demonstrated. Grossone gives us the ability to actually compute real probabilities, with a higher degree of accuracy than shown in [7]. Here we assume the equiprobable probability distribution. That is, given a finite alphabet S, the probability of each element occurring is

$$\frac{1}{|S|}$$

For $f \in \mathcal{A}$ there is a $B_{m,n}(x)$ that contains at least $|S|^{2①-k}$ elements, for some finite $k \geq 0$. Hence the probability, $P(y)$, of randomly finding another configuration y that equals x on the central vertical strip (stays close to x), under forward iteration of the cellular automaton, is

$$\frac{1}{|S|^{k+1}} \leq P(y) \leq \frac{1}{|S|} .$$

Under the definitions, it is possible for the probability $P(y)$ to equal $\frac{1}{|S|}$, for example the automaton map that takes everything to the quiescent state of 0's. Hence everything gets mapped to 0 and the probability is $\frac{1}{|S|}$ that another randomly chosen configuration will equal the initial in the 0^{th} place and then equal the initial under forward iterations.

For $f \in \mathcal{B}$ there is a $B_{m,n}(x)$ that contains at least $|S|^{\alpha①-k}$ elements, for some finite $k \geq 0$, $0 < \alpha < 2$ and α not infinitesimal, but $f \notin \mathcal{A}$. Hence the probability of finding another configuration y that stays arbitrarily close to x under forward iteration, is at most

$$\frac{1}{|S|^{(2-\alpha)①+k+1}} .$$

A simple computation will show this. For $f \in \mathcal{B}$, the probability of randomly finding another configuration, that will equal an initial configuration x in a central window upon forward iteration, is infinitesimal and hence highly improbable. However it is not impossible. Indeed, there are still a lot of configurations available. For a specific example see Example 4.2, the left (or right) shift map would have to have all other configurations agreeing with the initial on an infinite word out to the right (or left, respectively) and these values would have to be fixed without choice. In particular, choosing $B_{-1,1}(x)$ for the left shift map

σ, the probability of finding another configuration that would equal the given x in the finite window $x[-1,1]$[13] upon forward iterations of σ would equal

$$\frac{1}{|S|^{①+2}}\,.$$

It is obvious that this probability is infinitesimal however, as the following illustration shows, there are still an infinite number of configurations in the $B_{-1,1}(x)$ class.

$$\begin{array}{rcl} & & \qquad\qquad\qquad\qquad\quad \overbrace{\hphantom{1\,\langle 1\rangle\,0\,1\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1...}}^{①+2\ places\ fixed\ without\ choice} \\ x & = & ...*\ *\ *\ *\ *\ *\ *\ *\ 1\ \langle 1\rangle\ 0\,1\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1... \\ \sigma(x) & = & ...*\ *\ *\ *\ *\ *\ *\ 1\,1\ \langle 0\rangle\ 1\,0\,0\,1\,0\,1\,0\,0\,0\,1\,1\ ... \\ \sigma^2(x) & = & ...*\ *\ *\ *\ *\ *\ \ 1\,1\,0\ \langle 1\rangle\ 0\,0\,1\,0\,1\,0\,0\,0\,1\,1\ \ ... \\ . & & \\ . & & \\ . & & \end{array}$$

Here the $*$ means a wildcard choice for an element of alphabet S. Hence each element choice, on the fixed side, has a probability of occurrence of $\frac{1}{|S|}$.

For $f \in \mathcal{C}$ the probability of finding another configuration that equals a given initial configuration in a central window, under forward iterations, is much smaller and can possibly be 0.

6. Discussion and Conclusions

In this paper a classification scheme for one-dimensional (linear) cellular automata, based on the Infinite unit axion and grossone, has been presented. The entire domain space of one-dimensional automata, $X = S^{\mathbb{Z}}$, contains $|S|^{2①+1}$ configurations. This puts an upper bound representation on the number of elements in the entire space, hence we sub-divide the space into three components and use this to build a classification on the number of configurations whose forward evolution, under a cellular automaton, equal those (on a central window) of a given initial configuration.

This classification is in line with that of Wolfram and that of Gilman, however it is based on a numeric representation of counting elements in a set. Automata in class $\mathcal{A}$ are the least chaotic, having a very large number of configurations equaling those of a given configuration, on some central window[14], upon forward iterations of the automaton map. Automata in class $\mathcal{B}$, such as the left shift automaton, are more chaotic than those in class $\mathcal{A}$. However, it seems that

[13] The reader is reminded that, according to the definitions, this does not have to be a symmetric central window but has to include the zeroth place.

[14] Given the definition of the metric, it is allowable to say "staying close together" upon forward iterations.

they can still be described without too much complexity. Automata in class $\mathcal{C}$ are more difficult to find and are the most chaotic in the respect that there are relatively very few other configurations that will follow and stay close to a given. Indeed, the number of configurations that stay close to a given initial configuration, upon forward iterations, is much less than the other classes and may be simple infinite (either ①, or $①^2$, ..., or $①^n$, or some part thereof), finite or a single configuration.

Wolfram class 1 and 2 (see figure 1 and figure 2) seem to correspond to class $\mathcal{A}$ automata. By Theorem 4.1, Gilman's class A automata corresponds to the class $\mathcal{A}$ automata presented in this paper, however there may be some overlap with Gilman's class B automata. Being expansive, the left (or right) shift map belongs in Gilman's class C automata, while they both belong to class $\mathcal{B}$ presented herein. This shows the classifications have some differences. Automata in Wolfram class 4, for example rule 20 as seen in figure 4, seem to be similar to the shift map and hence are conjectured to fall into class $\mathcal{B}$. Wolfram class 3, as seen in rule 12 (see figure 3), exhibit aperiodic behavior and seem to correspond to the most chaotic, class $\mathcal{C}$. However some of the Wolfram totalistic rules are conjectured to, and some were proven to, be capable of universal computation, see [26]. Due to the nature of universal computation, some of these automata can fall into class $\mathcal{C}$. It is left as an open problem to show these. On the whole, the presented classification would be stronger if there was an algorithm to determine membership in the different classes and we pose this as an open problem.

References

[1] Berlekamp, E., R., Conway, J., H., Guy, R., K., *Winning Ways for Your Mathematical Plays*, Volume 4, 2nd Edition, A. K. Peters, Wellesley, Massachusettes, 2004.

[2] Calidonna, C.R., Naddeo, A., Trunfio, G.A., Di Gregorio, S., From classical infinite space-time CA to a hybrid CA model for natural sciences modeling, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 8137-8150.

[3] Chopard, B., Droz, M., *Cellular Automata Modeling of Physical Systems*, Cambridge University Press, 1998.

[4] D'Alotto, L., Cellular Automata Using Infinite Computations, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 8077 - 8082.

[5] D'Ambrosio, D., Filippone, G., Marocco, D., Rongo, R., Spataro, W., Efficient application of GPGPU for lava flow hazard mapping, *The Journal of Supercomputing*, 65(2): 630-644 (2013).

[6] De Cosmis, S., De Leone, R., The Use of Grossone in Mathematical Programming and Operations Research, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 8029 - 8038.

[7] Gilman, R., Classes of Linear Automata, *Ergodic Theory and Dynamical Systems*, 7, (1987) 105-118.

[8] Hedlund, G.A., Edomorphisms and Automorphisms of the Shift Dynamical System, Math. Sys. Theory **3** (1969), 51-59.

[9] Iudin, D. I., Sergeyev, Ya., D., Hayakawa, M., Interpretation of Percolation in Terms of Infinity Computations, Applied Mathematics and Computation, Vol. 218 Issue 16, (2012) 8099-8111.

[10] Lolli, G., Infinitesimals and Infinities in the History of Mathematics: A brief survey, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 7979 - 7988.

[11] Lolli, G., Metamathematical Investigations on the Theory of Grossone, Preprint, submitted and accepted for publication in *Applied Mathematics and Computation*, Elsevier.

[12] Margenstern, M., Using Grossone to count the number of elements of infinite sets and the connection with bijections, *p-Adic Numbers, Ultrametric Analysis and Applications*, 3(3), (2011), 196-204.

[13] Margenstern, M., An application of grossone to the Study of a Family of Tilings of the Hyperbolic Plane, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 8005-8018.

[14] Narici, L., Beckenstein, E., and Bachman, G., *Functional Analysis and Valuation Theory,* Marcel Dekker, Inc., NY, 1971.

[15] Sergeyev, Ya. D., *Arithmetic of Infinity*, Edizioni Orizzonti Meridionali, Italy, 2003.

[16] Sergeyev, Ya. D., Numerical Point of View on Calculus for Functions Assuming Finite, Infinite, and Infinitesimal Values Over Finite, Infinite, and Infinitesimal Domains, *Nonlinear Analysis Series A: Theory, Methods & Applications*, 71(12), (2009) e1688-e1707.

[17] Sergeyev, Ya. D., Numerical Computations with Infinite and Infinitesimal Numbers: Theory and Applications, "Dynamics of Information Systems: Algorithmic Approaches" edited by Sorokin, A., Pardalos, P. M., Springer, New York, 1 - 66, 2013.

[18] Sergeyev, Ya. D., A New Applied Approach for Executing Computations with Infinite and Infinitesimal Quantities, *Informatica*, 19(4), (2008) 567-596.

[19] Sergeyev, Ya. D., Measuring fractals by infinite and infinitesimal numbers, *Mathematical Methods, Physical Methods & Simulation Science and Technology*, Vol. 1(1), (2008) 217-237.

[20] Sergeyev, Ya., D., Garro, A., Observability of Turing Machines: A Refinement of the Theory of Computation, Informatica, Vol: 21 Issue: 3, (2010) 425-454.

[21] Sergeyev, Ya., D., Garro, A., Single-tape and multi-tape Turing machines through the Lens of Grossone methodology, The Journal of Supercomputing, Vol. 65, Issue 2, (2013) 645-663.

[22] Sirakoulis, G. Ch., Krafyllidis, I., Spataro, W., A Computational Intelligent Oxidation Process Model and its VLSI Implementation, International Conference on Scientific Computing Proceedings, (2009) 329-335.

[23] Trunfio, G. A., Predicting Wildfire Spreading Through a Hexagonal Cellular Automata Model, In: P.M.A. Sloot, B. Chopard, and A.G. Hoekstra, Editors, ACRI 2004, LNCS 3305, Spring, Berlin, (2004) pp. 725-734.

[24] Trunfio, G. A., D'Ambrosio, D., Rongo, R., Spataro, W., Di Gregorio, S., A New Algorithm for Simulating Wilfire Spread Through Cellular Automata, ACM Transactions on Modeling and Computer Simulation, Vol. 22, (2011), 1-26.

[25] Wolfram, S., Statistical Mechanics of Cellular Automata, *Reviews of Modern Physics*, Vol. 55, No. 3, (1983) 601-644.

[26] Wolfram, S., *A New Kind of Science*, Wolfram Media, Inc., IL, 2002.

[27] Wolfram, S., Universality and Complexity in Cellular Automata, *Physica D*, Vol. 10 (1984) 1-35.

[28] Zhigljavsky, A., Computing Sums of conditionally convergent and divergent series using the concept of grossone, *Applied Mathematics and Computation*, Vol. 218, Issue 16, (2012) 8064 - 8076.