4 Chapter 4: A First Look at the Space of Cellular Automata

In the previous chapters, we have seen examples of very carefully constructed cellular automata rules and initial configurations. In this and the following chapters, we change perspective. Rather than designing cellular automata to exhibit a prescribed phenomenon, we ask what can be said about a cellular automaton once it is given to us: which dynamical and computational properties can we observe or prove, and how common such properties are in the space of cellular automata?

The aim of this chapter is to introduce the first tools for such a study. We begin with the role of shifts and explain why finite cyclic dynamics should be understood as the restriction of the original CA to canonical periodic subspaces. After that, we introduce the radius as a natural parametrization of CA spaces, discuss elementary cellular automata as a first example of a CA family, present Wolfram's four classes as a heuristic guide to identifying typical CA behaviour, and finally describe two standard ways of reducing CA spaces: by restricting the form of the local rule and by quotienting the space by simple dynamical equivalences.

Throughout this chapter, we use boldface letters such as $\mathbf{v}, \mathbf{i}, \mathbf{n}$ to denote vectors in $\mathbb{Z}^d$, while ordinary letters denote scalar quantities.

4.1 Cellular Automata and their Global Dynamics

We begin by recalling the definition of a cellular automaton from Section **??**. Throughout this chapter, we will primarily view a cellular automaton as a dynamical system $(S^{\mathbb{Z}^d}, F)$, where the global map F is induced by a finite local rule applied uniformly across the lattice.

Definition 4.1 (Cellular Automaton)**.** *A d-dimensional cellular automaton (CA) $\mathcal{A}$ is specified by a finite set of states S, a neighbourhood $N = (\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_k) \in (\mathbb{Z}^d)^k$, and a local update rule $f : S^k \to S$. These determine the CA's global update rule $F : S^{\mathbb{Z}^d} \to S^{\mathbb{Z}^d}$ defined, for each configuration $c \in S^{\mathbb{Z}^d}$, by*

$$F(c)(\mathbf{v}) = f\big(c(\mathbf{v} + \mathbf{n}_1), \ldots, c(\mathbf{v} + \mathbf{n}_k)\big) \quad \text{for each cell } \mathbf{v} \in \mathbb{Z}^d.$$

The global rule updates the state of each cell by applying the same local rule to the states in its relative neighbourhood. This update is performed synchronously for all cells in parallel. We often write $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$.

For each $c \in S^{\mathbb{Z}^d}$ we call the sequence

$$c, F(c), F^2(c), F^3(c), \ldots$$

the *trajectory with initial configuration c*. If this trajectory is eventually periodic, we call its finite preperiod the *transient* and its periodic part the *limit cycle*. We call the matrix whose rows are the successive configurations in such a trajectory a *space-time diagram*.

4.2 Shifts, Periodic Configurations, and Finite Cyclic Dynamics

A fundamental structural property of every cellular automaton is translation invariance. This is captured by the family of shift maps.

Definition 4.2 (Shift map)**.** *Let $\mathbf{v} \in \mathbb{Z}^d$. We define the shift operator $\sigma_{\mathbf{v}} : S^{\mathbb{Z}^d} \to S^{\mathbb{Z}^d}$ by $(\sigma_{\mathbf{v}}(c))(\mathbf{w}) = c(\mathbf{v} + \mathbf{w})$ for each $\mathbf{w} \in \mathbb{Z}^d$.*

Observation 4.3 (Cellular automata commute with shifts)**.** *Let $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ be a cellular automaton and $\mathbf{v} \in \mathbb{Z}^d$. Show that $\sigma_{\mathbf{v}} \circ F = F \circ \sigma_{\mathbf{v}}$.*

This fact makes periodic configurations particularly natural. Since a CA commutes with shifts, it preserves every periodicity relation satisfied by a configuration. In particular, the set of totally periodic configurations that we define below is invariant under the CA dynamics.

Definition 4.4 (Totally periodic configuration)**.** *We say that a configuration $c \in S^{\mathbb{Z}^d}$ is totally periodic if there exist d linearly independent vectors $\mathbf{v}_1, \ldots, \mathbf{v}_d \in \mathbb{Z}^d$ such that $\sigma_{\mathbf{v}_i}(c) = c$ for all $i \in \{1, \ldots, d\}$.*

Indeed, if $c \in S^{\mathbb{Z}^d}$ satisfies $\sigma_{\mathbf{v}}(c) = c$, then

$$\sigma_{\mathbf{v}}(F(c)) = F(\sigma_{\mathbf{v}}(c)) = F(c),$$

so $F(c)$ is also invariant under $\sigma_{\mathbf{v}}$. Hence, every CA preserves total periodicity. If we denote by $\mathrm{Per}_{\mathrm{tot}} \subset S^{\mathbb{Z}^d}$ the set of all totally periodic configurations, then F maps $\mathrm{Per}_{\mathrm{tot}}$ into itself.

This explains why finite cyclic models arise naturally when observing CA dynamics. When computing CA trajectories on a computer, one usually considers the grid to be finite cyclic of size $\mathbf{m} = (m_1, \ldots, m_d) \in \mathbb{N}^d$. We define this mode of operation below.

Definition 4.5 (Finite cyclic global map)**.** *Let $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ be a cellular automaton with neighbourhood $N = (\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_k) \in (\mathbb{Z}^d)^k$, and local rule f. Fix $\mathbf{m} = (m_1, m_2, \ldots, m_d) \in \mathbb{N}^d$. We define the global map of $\mathcal{A}$ operating on a cyclic grid of size $\mathbf{m}$ to be the map*

$$F_{\mathbf{m}} : S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}} \to S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}}$$

defined as follows. For $\mathbf{i} = (i_1, i_2, \ldots, i_d) \in \mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}$ and $c \in S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}}$, we put

$$F_{\mathbf{m}}(c)(\mathbf{i}) = f\big(c((\mathbf{i} + \mathbf{n}_1) \bmod \mathbf{m}), c((\mathbf{i} + \mathbf{n}_2) \bmod \mathbf{m}), \ldots, c((\mathbf{i} + \mathbf{n}_k) \bmod \mathbf{m})\big).$$

A priori, this may look like a different dynamical system from the original one on $\mathbb{Z}^d$. The point is that this is not the case: finite cyclic dynamics can be identified with the restriction of the original CA to a canonical finite invariant subspace of periodic configurations. To formulate this precisely, we introduce the corresponding notion of isomorphic dynamical systems.

Definition 4.6 (Isomorphism of dynamical systems)**.** *Let $\mathcal{A} = (X, F)$ and $\mathcal{B} = (Y, G)$ be (arbitrary) dynamical systems. We say that $\mathcal{A}$ and $\mathcal{B}$ are isomorphic if there exists a bijection $\varphi : X \to Y$ such that*

$$\varphi \circ F = G \circ \varphi.$$

Remark 4.7. *In later chapters, we will strengthen this notion by imposing additional topological or computational requirements on the map φ.*

If $\mathcal{A} = (X, F)$ and $\mathcal{B} = (Y, G)$ are isomorphic via $\varphi : X \to Y$, then each trajectory of $\mathcal{A}$

$$x, F(x), F^2(x), F^3(x), \ldots$$

corresponds to a trajectory of $\mathcal{B}$

$$\varphi(x), G(\varphi(x)), G^2(\varphi(x)), G^3(\varphi(x)), \ldots;$$

we simply use φ to map the former onto the latter trajectory. Thus, isomorphic dynamical systems have the same trajectories and space-time diagrams up to relabelling by φ. Thus, from the dynamical point of view, they should be regarded as the same system (modulo the bijection φ).

Every finite cyclic configuration can be unwrapped into a totally periodic configuration on $\mathbb{Z}^d$. This identifies the finite cyclic dynamics with the restriction of the original CA to a canonical finite invariant subspace, as summarized in the next exercise.

Exercise 4.8 (Finite cyclic configurations and periodic subspaces)**.** *Let* $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ *be a cellular automaton and* $\mathbf{m} = (m_1, m_2, \ldots, m_d) \in \mathbb{N}^d$*. We can "unwrap" each finite cyclic configuration* $\tilde{c} \in S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}}$ *to obtain a totally periodic configuration* $c \in S^{\mathbb{Z}^d}$ *where for each* $\mathbf{i} = (i_1, i_2, \ldots, i_d) \in \mathbb{Z}^d$ *we define*

$$c(\mathbf{i}) = \tilde{c}(i_1 \bmod m_1, i_2 \bmod m_2, \ldots, i_d \bmod m_d).$$

Let us define the subspace of all periodic configurations obtained in this way by $C_{\mathbf{m}}$*. Show that the two dynamical systems are isomorphic:*

$$F\big|_{C_{\mathbf{m}}} : C_{\mathbf{m}} \to C_{\mathbf{m}} \quad \textit{and} \quad F_{\mathbf{m}} : S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}} \to S^{\mathbb{Z}_{m_1} \times \mathbb{Z}_{m_2} \times \cdots \times \mathbb{Z}_{m_d}}.$$

Is it also the case that every totally periodic configuration can be expressed as an unwrapped version of some finite cyclic configuration?

Thus, computer-generated space-time diagrams on finite tori should be interpreted as exact information about the original CA, restricted to periodic subsets of its configuration space.

4.3 Radius and Parametrized CA Spaces

If we want to study not just one CA, but an entire family of them, we need a convenient way to parametrize CA spaces. The usual choice is to fix the dimension, the state set, and a symmetric neighbourhood determined by a single parameter: the radius.

A basic issue is that the neighbourhood of a CA is not uniquely determined.

Observation 4.9. *Consider* $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ *a* d*-dimensional cellular automaton with neighbourhood* $N \subset (\mathbb{Z}^d)^k$ *and local rule* f *and let* $N' \subset \mathbb{Z}^d$ *be a finite set such that* $N \subseteq N'$*. Then we can interpret* $\mathcal{A}$ *as a CA with neighbourhood* N' *simply by extending* f *on the additional inputs in such a way, that it does not depend on them.*

To obtain a canonical parametrization, one typically works either with the minimal neighbourhood, or a symmetric one.

Definition 4.10 (CA radius)**.** *Let* $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ *be a* d*-dimensional cellular automaton and let* $r \in \mathbb{N}$*. We say* $\mathcal{A}$ *has radius* r*, if its neighbourhood is of the form* $N = \{\mathbf{v} \in \mathbb{Z}^d \mid \|\mathbf{v}\| \leq r\}$*. In such a case,* $|N| = (2r+1)^d$ *and the local rule* f *is a function* $f : S^{(2r+1)^d} \to S$*.*

We will typically interpret each CA as a CA with radius r for the smallest possible $r \in \mathbb{N}$. This is always possible: if a CA has minimal neighbourhood $N = (\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_k) \in (\mathbb{Z}^d)^k$, we simply put

$$r = \max\{\|\mathbf{v}\| \mid \mathbf{v} \in N\}.$$

This allows us to speak about the CA space determined by the triple (d, S, r). Once these parameters are fixed, every CA in the space is obtained by choosing a suitable local rule.

4.4 Elementary Cellular Automata

The smallest nontrivial CA space of this kind is obtained by fixing $d = 1$, $S = \mathbf{2} = \{0, 1\}$, and $r = 1$. These are the Elementary Cellular Automata (ECAs).

Each local rule $f : \mathbf{2}^3 \to \mathbf{2}$ is uniquely described by its *Wolfram number* given by:

$$2^0 f(0,0,0) + 2^1 f(0,0,1) + \cdots + 2^6 f(1,1,0) + 2^7 f(1,1,1).$$

We will refer to each ECA by the corresponding Wolfram number of its local rule. In Figure 1, we show a space-time diagram of Rule 90. There are only 256 ECAs, which makes this class small enough to study exhaustively. Despite its size, it already exhibits a wide range of behaviours, and some ECAs are even known to be computationally universal [1]. For this reason, ECAs serve as a first tractable laboratory for exploring CA spaces.

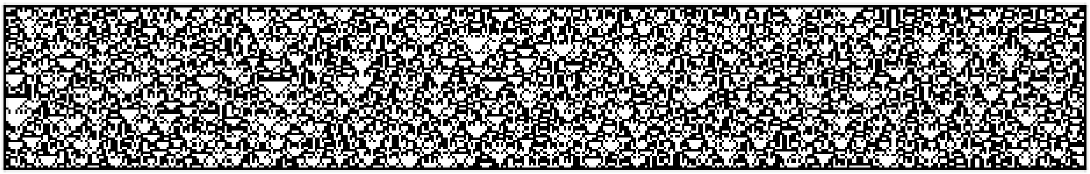

Figure 1: Space-time diagram of Rule 90; with local rule $f(x, y, z) = (x + z) \bmod 2$; operating on a cyclic configuration of size 250. Time flows downwards.

To build intuition, one can inspect space-time diagrams arising from simple families of initial configurations, for instance random configurations or a single cell in state 1 surrounded by 0s. This was done in particular detail by Stephen Wolfram, who proposed a heuristic way to classify CA dynamics, as discussed next.

4.5 Wolfram Classification

A particularly influential attempt to organize such observations was proposed by Stephen Wolfram in [3]. Based on empirical inspection of cellular automata evolutions, he suggested the following four qualitative classes of behaviour:

1) Evolution leads to a homogeneous state.

2) Evolution leads to a set of separated simple stable or periodic structures.

3) Evolution leads to a chaotic pattern.

4) Evolution leads to complex localized structures, sometimes long-lived.

Figure 2 illustrates the four classes on representative examples of elementary cellular automata.

This classification provides a convenient first vocabulary for discussing CA behaviour visible in space-time diagrams. Its boundaries, however, are not sharp: class membership is often difficult to justify rigorously and may depend on the chosen family of initial configurations. In this chapter, we use it mainly to organize examples and build intuition.

With this caveat in mind, Classes 1 and 2 are usually associated with ordered behaviour: trajectories quickly settle into short cycles, typically with homogeneous, stable, or periodic patterns, local perturbations tend to disappear quickly, and the resulting space-time diagrams are highly compressible.

On the other hand, Class 3 is usually interpreted as the chaotic class. Although there is no single agreed-upon definition of chaos for discrete dynamical systems, Class 3 examples are typically associated with sensitive dependence on local perturbations, apparent randomness, and low compressibility of their space-time diagrams. Class 4 is the most mysterious one: it is characterized by the presence of localized structures that persist for long periods and interact in complicated ways.

Wolfram further speculated that Class 4 captures the regime in which computationally rich behaviour appears. This idea is suggestive and historically important, but it remains heuristic. In

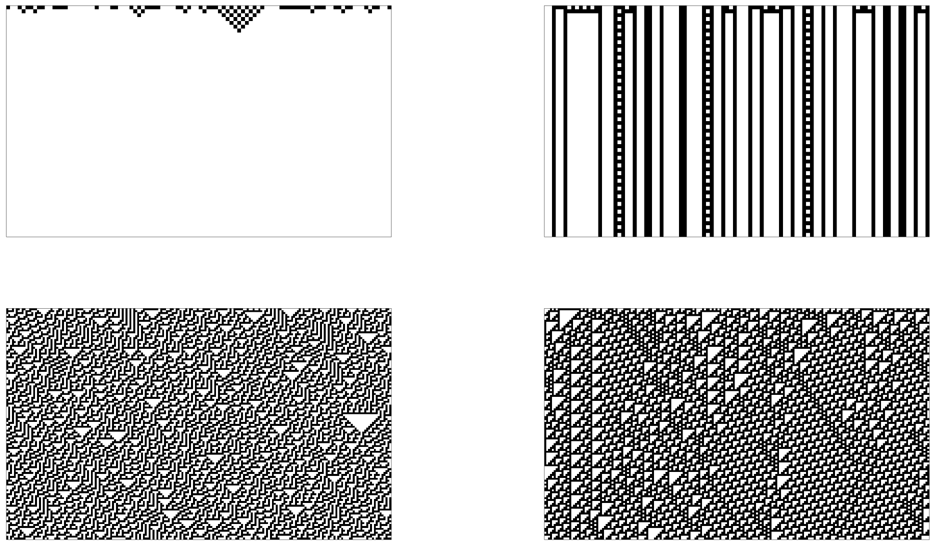

Figure 2: Space-time diagrams of ECAs from each of Wolfram's classes. *(Top left)* Class 1 Rule 32. *(Top right)* Class 2 Rule 108. *(Bottom left)* Class 3 Rule 30. *(Bottom right)* Class 4 Rule 110.

particular, it is still an open problem to formulate satisfactory formal characterizations of Class 3 and Class 4, and to determine which dynamical properties are necessary or sufficient for universal computation. We will discuss several results in this direction in later chapters.

We now mention a few standard examples whose space-time diagrams are commonly associated with Wolfram's classes. The purpose of these examples is again mainly intuitive: they provide concrete pictures that one should have in mind when hearing terms such as "chaotic" or "complex" in the CA context.

Rule 30 Rule 30 is one of the canonical examples associated with Class 3. Starting from a simple initial configuration, its space-time diagrams display a highly irregular pattern that appears random, despite being generated by a completely deterministic rule; see Figure 3. For this reason, Rule 30 became one of the standard examples of apparent chaos in elementary cellular automata.

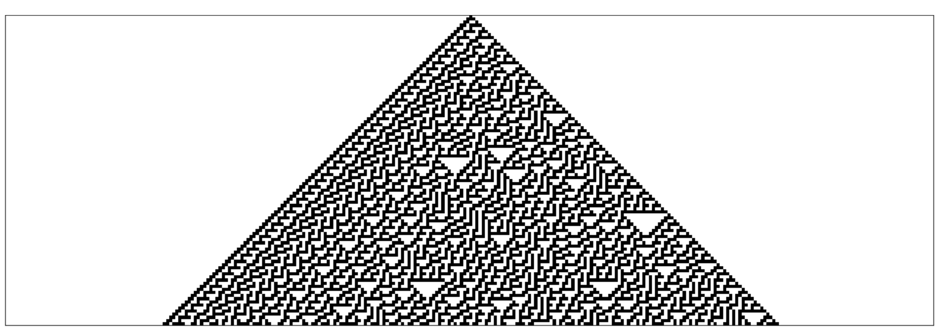

Figure 3: Rule 30 iterated from an initial configuration with a single cell in state 1.

Rule 110 Rule 110 is a standard example associated with Class 4. Its space-time diagrams, for suitable initial configurations, display localized structures propagating on a nontrivial background.

These structures interact in ways that are difficult to predict more efficiently than by simulating the automaton itself. Wolfram conjectured that such interactions could support universal computation, and this was later established in [1].

Game of Life Perhaps the most famous cellular automaton is Conway's Game of Life. It is a two-dimensional outer totalistic CA with state set $S = \{0, 1\}$ and radius $r = 1$. The local rule is specified in terms of the number of live neighbours of a cell: a live (1) cell survives if it has two or three live neighbours, while a dead (0) cell becomes alive if it has exactly three live neighbours.

Its dynamics exhibits a rich variety of persistent and mobile structures, including oscillators, spaceships, and complicated collisions between localized patterns. For this reason, it is one of the canonical examples usually associated with Class 4 behaviour.

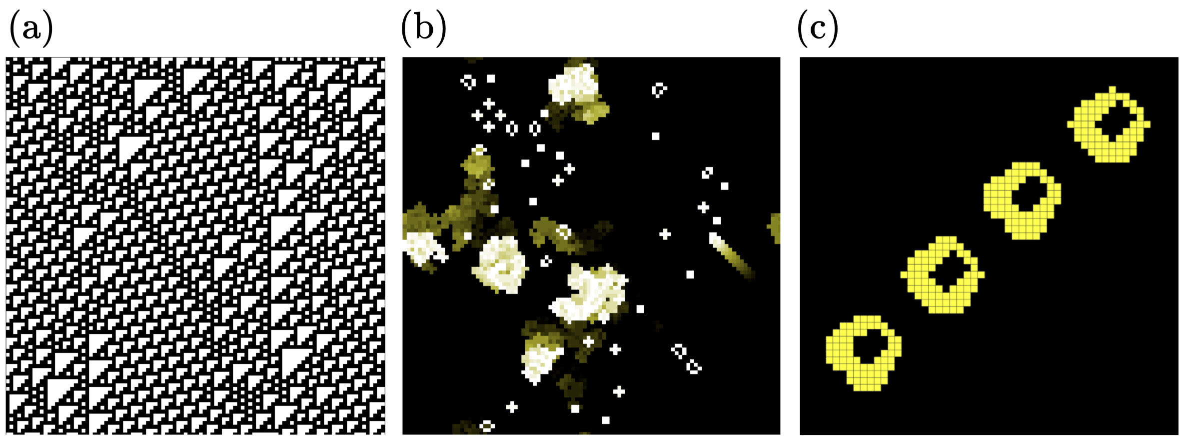

Figure 4: Examples commonly associated with Wolfram's Class 4. (a) 1D ECA Rule 110. (b) 2D CA Game of Life, past states shown with shades of yellow. (c) Illustrative phases of a glider in a 2D CA Larger than Life rule $(5, 34, 45, 34, 58)$.

Larger than Life The family of cellular automata known as *Larger than Life* generalizes the Game of Life by increasing the neighbourhood radius while preserving a similar birth–survival format. These automata operate on a two–dimensional grid with radius $r > 1$, so that each cell interacts with $(2r + 1)^2$ neighbours (including itself). Instead of specifying exact neighbour counts as in Game of Life, the rule is typically given by intervals: a dead cell becomes alive if the number of live neighbours lies within a specified birth interval, and a live cell survives if the number lies within a survival interval.

More concretely, following the notation from [2], each Larger than Life local rule is specified by a tuple $(r, \beta_1, \beta_2, \delta_1, \delta_2)$ where:

- r is the CA radius
- a dead cell will become alive if and only if the total number of its alive neighbours k satisfies $\beta_1 \leq k \leq \beta_2$.
- an alive cell stays alive if and only if the total number of its alive neighbours k satisfies $\delta_1 \leq k \leq \delta_2$.

An illustration of a glider in a Larger than Life rule $(5, 34, 45, 34, 58)$ is shown in Figure 4 (c).

Exploration of this rule space revealed many examples with oscillators, moving patterns, and long-lived localized structures. In this sense, Larger than Life provides evidence that behaviour reminiscent of Life is not confined to one isolated rule, but appears in a broader region of CA space. We will return to this family later in the course.

4.6 Restricting CA Spaces

If we want to study the distribution of cellular automata across Wolfram's classes beyond the elementary case, we quickly run into the problem regarding the vastness of such spaces. Indeed, simply increasing the radius of 1D CAs with two states to $r = 2$ already results in $2^{2^5} = 2^{32}$ local update rules. One standard way to reduce the CA space size is to restrict the form of the local rule.

4.6.1 Totalistic and Outer Totalistic Cellular Automata

Definition 4.11 (Totalistic cellular automaton). *A CA $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ with neighbourhood $N = (\mathbf{n}_1, \ldots, \mathbf{n}_k)$ and local rule f is totalistic if f only depends on the number of cells in each state, rather than on their exact position in the neighbourhood. More formally, f is invariant under every permutation of its inputs; let $\pi \in S_k$ be a permutation of the inputs, then $f(s_{\pi(1)}, s_{\pi(2)}, \ldots, s_{\pi(k)}) = f(s_1, s_2, \ldots, s_k)$ for all $s_1, s_2, \ldots, s_k \in S$.*

Exercise 4.12 (Totalistic ECAs). *Which elementary cellular automata are totalistic?*

Definition 4.13 (Outer totalistic cellular automaton). *Let $\mathcal{A} = (S^{\mathbb{Z}^d}, F)$ be a CA with local rule f and with neighbourhood $N = (\mathbf{n}_1, \ldots, \mathbf{n}_k)$, where $\mathbf{n}_1 = 0$; thus f depends on the central cell. A is outer totalistic if f only depends on the number of cells in each state, rather than on their exact position in the neighbourhood, except for the central cell. More formally, f is invariant under every permutation of its inputs which preserves the first input's location; let $\pi \in S_k$ be a permutation of the inputs such that $\pi(1) = 1$, then $f(s_1, s_{\pi(2)}, \ldots, s_{\pi(k)}) = f(s_1, s_2, \ldots, s_k)$ for all $s_1, s_2, \ldots, s_k \in S$.*

Examples of outer totalistic CAs include Game of Life and Larger than Life automata. Considering only totalistic or outer totalistic CAs allows us to exhaustively study CA classes with larger neighbourhoods or number of states.

Exercise 4.14 (Counting totalistic CAs). *How many outer totalistic CAs are there in two dimensions, with radius $r = 1$ and two states?*

4.6.2 Cellular Automata up to Equivalent Dynamics

A second classical way to reduce CA spaces is to identify rules whose dynamics is the same up to a given class of isomorphisms. In practice, this means that two rules are regarded as equivalent whenever their trajectories differ only by a systematic relabelling of configurations. Some examples of isomorphic CAs can be quite nontrivial; here, however, we focus only on two simple ways of generating isomorphic rules and illustrate them on the class of ECAs.

4.6.3 Equivalent ECA Classes

For ECAs, there are two standard symmetries that generate equivalent dynamics:

- reflection of the neighbourhood,

- exchange of the states 0 and 1.

Formally, let **ECA** denote the set of all local rules of elementary cellular automata. We have $|\mathbf{ECA}| = 256$. We define bijective transformations of the local rules $\pi : \mathbf{ECA} \to \mathbf{ECA}$ and $\sigma : \mathbf{ECA} \to \mathbf{ECA}$ as follows:

$$\begin{aligned} \pi(f)(a,b,c) &= f(c,b,a) && \forall a,b,c \in \mathbf{2}, \\ \sigma(f)(a,b,c) &= 1 - f(1-a, 1-b, 1-c) && \forall a,b,c \in \mathbf{2}. \end{aligned}$$

Rule $\pi(f)$ is obtained from f by changing the role of the “left and right neighbour”. Similarly, rule $\sigma(f)$ is obtained from f by changing the role of states 0 and 1. It is not difficult to see that f and $\pi(f)$ have equivalent dynamics via an isomorphism that “reflects each configuration along the cell with index 0”. Concretely, the isomorphism $\Pi : \mathbf{2}^{\mathbb{Z}} \to \mathbf{2}^{\mathbb{Z}}$ is defined as $\Pi(c)_i = c_{-i}$ for all $i \in \mathbb{Z}$ and $c \in \mathbf{2}^{\mathbb{Z}}$. Similarly, f and $\sigma(f)$ have equivalent dynamics via an isomorphism that exchanges the state 0 and 1 in each configuration; formally defined as $\Sigma : \mathbf{2}^{\mathbb{Z}} \to \mathbf{2}^{\mathbb{Z}}$ with $\Sigma(c)_i = 1 - c_i$ for all $i \in \mathbb{Z}$ and $c \in \mathbf{2}^{\mathbb{Z}}$. We illustrate the two cases in Figures 5 and 6.

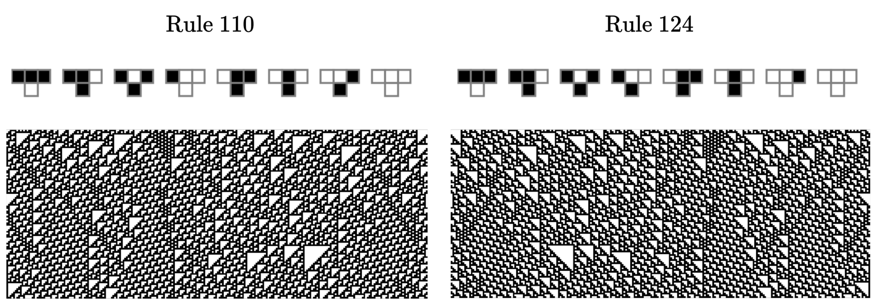

Figure 5: An example of Rule 110, and Rule 124 obtained from it by exchanging the role of the left and right neighbour, as can be verified from the two rule tables. We show the space-time diagram of Rule 110 with a finite cyclic initial configuration $c \in \mathbf{2}^{250}$. For Rule 124 we show the space-time diagram with a reflected initial configuration $\Pi(c)$. The two space-time diagrams are simply reflections of each other.

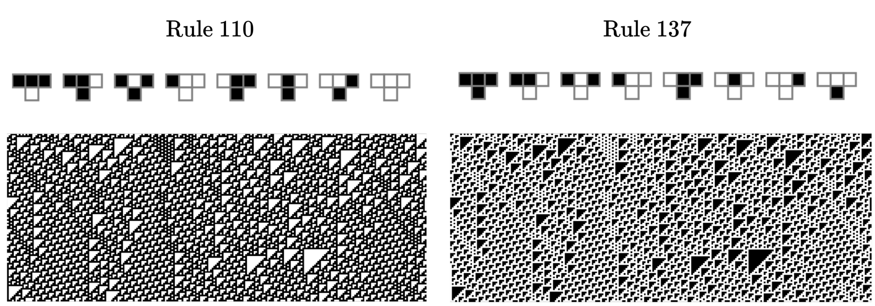

Figure 6: An example Rule 110, and Rule 137 obtained from it by exchanging the role of state 0 and state 1, as can be verified from the two rule tables. We show the space-time diagram of Rule 110 with a finite cyclic initial configuration $c \in \mathbf{2}^{250}$. For Rule 137 we show the space-time diagram with an initial configuration $\Sigma(c)$ where the values 0 and 1 are exchanged. The two space-time diagrams show the same image, just with the two colours exchanged.

Exercise 4.15 (The fourth twin)**.** *Can you construct one more ECA, apart from Rule 124 and Rule 137, whose dynamics is isomorphic to Rule 110? How can you do that using the two examples above?*

These symmetries generate an action of a finite group on the set of ECA rules. The orbits of this action are exactly the standard equivalence classes of ECAs. This gives a first concrete example of quotienting a CA space by dynamical equivalence.

Concretely, the group G that acts on the set **ECA** is generated by π and σ. Let $\rho \in G$, $f \in \mathbf{ECA}$ and $g = \rho(f)$. Then, it is clear that f and g are local rules of ECA that are isomorphic, and thus have equivalent dynamics (since ρ is a particular composition of π and σ, the witnessing conjugation is the corresponding composition of Π and Σ). It is not difficult to verify that

$$\pi^2 = \mathrm{id}, \quad \sigma^2 = \mathrm{id}, \quad \pi\sigma = \sigma\pi.$$

And as a result, it follows that $G = \{\,\mathrm{id}, \pi, \sigma, \pi\sigma\}$.

The action of G induces an equivalence relation $\sim_G$ on **ECA**, defined by $f \sim_G g$ if there exists $x \in G$ such that $f = x(g)$.

Now, to compute the number of unique ECA rules up to this equivalence is a simple application of the Burnside's lemma.

Lemma 4.16 (Burnside's Lemma)**.** *Let G be a finite group acting on a set X. For each $g \in G$, let*

$$X^g = \{x \in X \mid g(x) = x\}$$

denote the set of elements of X that are fixed by g and let

$$\mathcal{O}_x = \{y \in X \mid \text{ there exists } g \in G \text{ such that } g(x) = y\}$$

denote the orbit of $x \in X$. Then the number of unique orbits, denoted $|X/G|$, is given by

$$|X/G| = \frac{1}{|G|}\sum_{g \in G} |X^g|.$$

In our case, we precisely aim to count $|\mathbf{ECA}/G|$. Therefore, for each $g \in G$ we have to count the number of its fixed points.

- **Fixed points of $\boldsymbol{\pi}$**: We need to count local rules f such that $f(a,b,c) = f(c,b,a)$ for all $a,b,c \in \mathbf{2}^3$. There are $2^6 = 64$ of those.
- **Fixed points of $\boldsymbol{\sigma}$**: We are free to specify the values $f(1,1,1)$, $f(1,1,0)$, $f(1,0,1)$, $f(1,0,0)$, and the remainder is fully determined. There is $2^4 = 16$ of those.
- **Fixed points of $\boldsymbol{\pi\sigma}$**: We are free to specify the values $f(1,1,1)$, $f(1,1,0)$, $f(1,0,1)$, $f(0,1,1)$, and the remainder is fully determined. There is $2^4 = 16$ of those.

Altogether, we have:

$$|\mathbf{ECA}/G| = \frac{1}{4}\big(X_{id} + X_\pi + X_\sigma + X_{\pi\sigma}\big) = \frac{1}{4}\big(256 + 64 + 16 + 16\big) = 88.$$

Exercise 4.17 (Equivalences of 2D CAs)**.** *Consider the class of two-dimensional CAs with von Neumann neighbourhood and two states. How many CAs are in this class? What are the natural symmetries of their local rules to consider? How many unique CAs up to the equivalence does it induce?*

Remark 4.18. *We note that by no means is the group G defined above describing all the possible ways to generate two ECAs with equivalent dynamics. Some equivalences can come from for less trivial symmetries. We will discuss such examples later in the course.*

Chapter Summary

In this chapter, we introduced the global dynamical viewpoint on cellular automata, explained the role of shifts and periodic subspaces, and saw how finite cyclic dynamics fits naturally into the infinite CA framework. We then introduced radius as a way to parametrize CA spaces, used elementary cellular automata as a first laboratory for exploration, discussed Wolfram's classes as a heuristic guide to classifying CA behaviour, and presented two basic methods for reducing CA spaces: restriction of local rules and quotienting by natural symmetries.

In the next chapter, we begin the systematic study of CA dynamics using tools from topological dynamics.

References

[1] Cook, M. Universality in elementary cellular automata. *Complex Systems 15*, 1 (2004), 1–40.

[2] Evans, K. M. Larger than life: Digital creatures in a family of two-dimensional cellular automata. In *Discrete Models: Combinatorics, Computation, and Geometry (DM-CCG 2001)* (2001), vol. AA of *Discrete Mathematics and Theoretical Computer Science Proceedings*, pp. 177–192.

[3] Wolfram, S. Universality and complexity in cellular automata. *Physica D: Nonlinear Phenomena 10*, 1-2 (1984), 1–35.