A Group-Theoretic Interpretation of Margolus Neighborhood Cellular Automata

Austin Jay Warner

Senior Honors Thesis
Department of Mathematics
Oklahoma State University
Spring 2016
with Dr. Henry Segerman

Contents

Abstract

A set-theoretic structure of Margolus neighborhood cellular automata is developed to accommodate a group structure in an intuitive way. It is proven that pairs of reversible Margolus rule-global maps generate a group of bijections on a finite $2n \times 2m$ grid of binary cells with function composition. This group can further be understood as a group action on the grid. We focus on the subgroup that consists of pairs of reversible, conservative rules and, in particular, the action of this group on the set of all possible "lonely universes" (grids with one living cell). We examine the permutation representation of this action and compute the sizes of the subgroups that are the isomorphic copies of the group under the permutation representation map.

1 Introduction

1.1 Background

A *cellular automaton* can be thought of simply as a grid of cells that evolves in discrete time according to locally-defined rules that are applied homogeneously on the *grid*. The grid may have a finite number of dimensions and each cell may assume one of a finite number of values. Canonically, the grid extends to infinitely in every dimension. The spirit of cellular automata (CA) is the relationship between the behavior of the grid over time and the locally-defined *rules* that produce this evolution. These locally defined rules are functions whose domain is some subset of the grid called a neighborhood.

These neighborhoods can assume many forms. In the case of one-dimensional cellular automata, the rule is often defined such that the value of a cell in the subsequent time-step will consider only the current state of the cell and its adjacent "neighbors." This rule is a special case of the radial neighborhood which considers a p-neighborhood about a cell. For example, say we want to compute the value of the *ith* cell in the next time step. We must then consider the values of the cells $i-p, i-(p+1), ..., i+p$. So the aforementioned rule would be the case of the radial neighborhood with $p = 1$.

Example 1.1. Consider the 1-dimensional cellular automaton defined by the 1-radial rule

where the $3-$*tuple* on the first line maps to the singlet below (also represented by

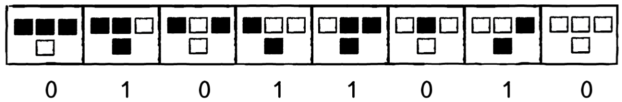
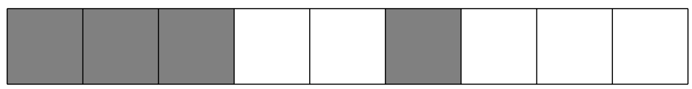


Figure 1: [13] 1-dimensional Local Rule

a 0 or 1). Applying this rule to an *arrangement*, that is, a grid on which each cell has assumed a value, on a 1-dimensional grid on a torus,



i = 0 1 2 3 4 5 6 7 8

we get

i = 0 1 2 3 4 5 6 7 8 .

Breaking this down further, say we start from the left-hand side and thus take the $3-$*tuple*

i = 0 1 2 .

Referring to the rule above, we see that this $3-$*tuple* maps to a non-living (0 or white) cell. Therefore, in the next time-step, the cell of the same index as the central cell here will be white. That is, for $i = 1$, the cell will be white, and so forth.

This idea extends to two and three-dimensional CA as well. Along with the radial neighborhood, well studied neighborhoods include the Moore neighborhood (from Conway's Game of Life [1]) and the von-Neumann neighborhood [2] (Figure 1). Note that it is the state of the neighborhood, the gray cells (and sometimes the central cell), that determines the state of the central cell in the next time-step.

In this paper we analyze the Margolus neighborhood [3, pages 119-138]. These cellular automata are conducive to reversibility which in turn realizes a group structure. Said structure occupies a central theme in this paper. With the Margolus

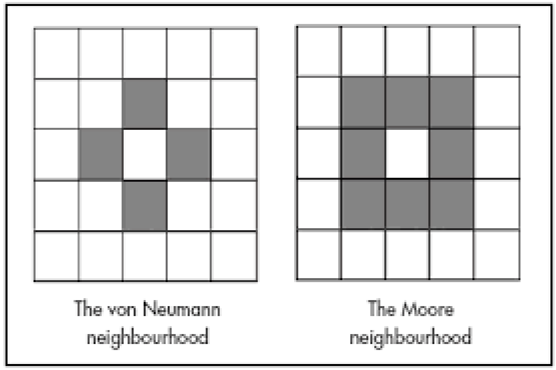
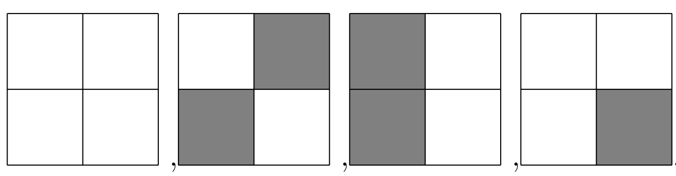


Figure 2: [10] Common CA Neighborhoods

neighborhood construction, a two-dimensional grid is partitioned into two-by-two blocks of cells, usually on an infinite grid whose cells can assume a binary value (0 or 1, black or white, alive or dead). In this paper we specifically discuss finite grids. We will refer to a two-by-two block of cells as a Margolus neighborhood (MN). Likewise we will refer to a rule whose domain and codomain are the set of all possible configurations of the Margolus neighborhood as a Margolus rule. Examples of Margolus neighborhood configurations include



The $2n \times 2m$ grid is partitioned into two-by-two blocks so that this rule in turn extends to the entire grid by being applied to every MN (Figure 2). Thus Margolus rules act on Margolus neighborhoods, assigning each possible arrangement of cells in an MN to a corresponding arrangement MN via a formal mathematical function. One such interpretation would be a function $r : \{0,1\}^4 \to \{0,1\}^4$ where each cell in a block is designated by a position in the $4-tuple$.[1] Note the distinction between Margolus neighborhoods and arrangements of Margolus neighborhoods. An MN is a two-by-two block of cells whereas an arrangement is a $4-tuple$ that specifies living

[1]Representing the examples of Margolus neighborhood configurations just given, we have $(0,0,0,0), (0,1,1,0), (1,0,1,0)$, and $(0,0,0,1)$, respectively, although we could arbitrarily orient the neighborhood in an alternative way.

and dead cells within an MN. We will nonetheless sometimes refer to a $4-$*tuple* arrangement as an MN, but the context will disambiguate.

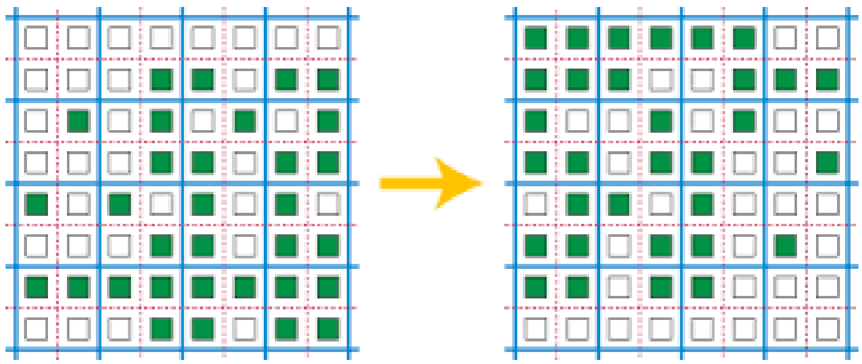


Figure 3: [12] A Margolus rule extending to the grid

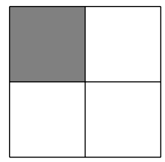
Once a rule has been applied to each MN (as in Figure 2), the grid is then re-partitioned into new two-by-two blocks. Figure 2 shows the two partitions: one outlined in blue, the other in red. After the grid has been re-partitioned, a Margolus rule has been entirely executed and one time-step is complete. Traditionally, the same Margolus rule is applied again iteratively. In this paper, we do not stipulate that it must be the same rule applied at each time-step. We allow different Margolus rules to be applied in sequence.

1.1.1 Reversible and conservative rules

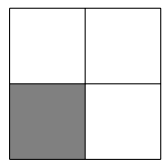
We see (Figure 2) that each arrangement in an MN is reassigned according to some Margolus rule. For example, the upper-left most MN of four dead cells maps to the MN of four living cells, etc. $(r : (0, 0, 0, 0) \mapsto (1, 1, 1, 1))$. This is, in fact, a special type of Margolus rule that is referred to as "reversible."[8]

Definition 1. A Margolus rule is *reversible* [8] if each possible arrangement of the neighborhood has a unique time-step predecessor.

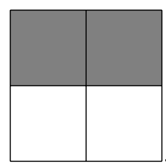
Example 1.2. Consider an MN cellular automata on a 2×2 torus with a rule such that



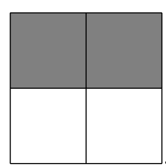
and



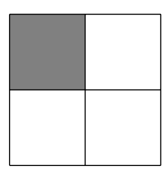
both map to

.

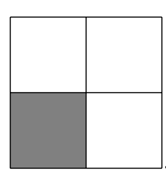
This rule is *not* reversible since, given

,

we cannot determine whether or not the inverse map would map to



or

.

Consequently, reversible Margolus rules are bijections on Margolus neighborhoods (that is, $r : \{0,1\}^4 \to \{0,1\}^4$ is a bijection). In fact, r^{-1} is also a (reversible)

Margolus rule, hence the appellation "reversible." The proof follows directly from the bijectivity of r.

Lemma 1. *The inverse of a reversible cellular automaton is also a reversible cellular automaton [4].*

Much has been written on the relationships between Margolus rules and the resulting behavior on the grid.[1][2][3][6][8] For example, we know that a Margolus CA is reversible on the entire grid if and only if the local rule is reversible.[9] This will be proven in Section 2 along with further explanation. From this we establish one more essential result.

Lemma 2. *"A cellular automaton is reversible if and only if it is bijective."[8]*

There is one more category of rules that we need to define: conservative rules.

Definition 2. A Margolus rule is said to be *conservative* [5] if the map preserves the number of living cells. In the $4-$*tuple* style that we have been using, let $x = (x_1, x_2, x_3, x_4)$ for $x_i \in \{0, 1\}$ and r a Margolus rule such that $r(x) = (y_1, y_2, y_3, y_4)$, then r is a conservative rule if and only if

$$\sum_{i=1}^{4} x_i = \sum_{i=1}^{4} y_i$$

for all Margolus neighborhood arrangements x.

1.2 Notation

Throughout this paper we consider a $2n \times 2m$ grid on a torus ($n, m \in \mathbb{N}$). With this construction we face none of the undesirable ramifications of edges or uneven Margolus neighborhoods, but we do begin to accrue a weighty amount of definitions and denotations. Let R be the set of reversible Margolus rules and from this point on, r is an element of R.

1.2.1 The Multiverse

We are interested in the collection of all possible arrangements of this grid. Firstly, we have reversible Margolus rules, which are bijections $r : \{0, 1\}^4 \to \{0, 1\}^4$. Secondly, we have bijections on the set of all possible configurations of the grid of a given size ($\rho : \{0,1\}^{2n\times 2m} \to \{0,1\}^{2n\times 2m}$). We claim that when r is applied to every 2×2 block on the grid, the resulting map on the entire grid $\rho : \{0,1\}^{2n\times 2m} \to \{0,1\}^{2n\times 2m}$ is a bijection. We prove this in Section 2. Later we will consider bijections on subsets of the set of all possible configurations. We need more definitions to deal with this in more depth.

Definition 3. A *universe* is a $2n \times 2m$ partitioned grid on a torus with some arrangement of living and dead cells.

Example 1.3. Consider Figure 3 (below) on a torus with either the blue-line partition or the red-line partition. This is an example of a universe.

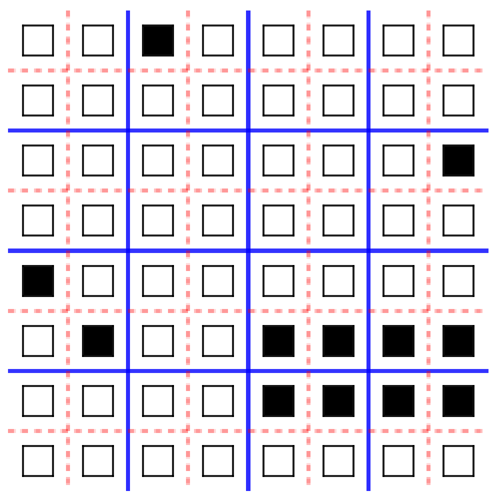
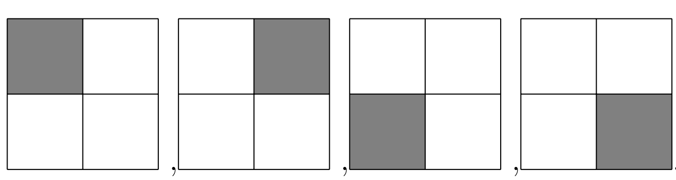


Figure 4: [11] An 8×8 universe (on a torus) partitioned by the blue lines or red lines

Definition 4. We refer to a universe with only one living cell as a *lonely universe*.

Definition 5. A *multiverse* is a collection of universes.

Example 1.4. Often we use qualifiers such as "the multiverse of all $2n \times 2m$ grids" or "the multiverse of 2×4 universes with two living cells." The multiverse of 2×2 lonely universes is the set



Bijections on multiverses form the basis for this paper.

2 A Set-theoretic Margolus Rule Interpretation

Let $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ be the multiverse of universes of size $2n\times 2m$ with an arbitrary 2×2 block partition and let $\mathfrak{M}({}_\beta T_{2n\times 2m})$ be the collection of universes with the alternate partition. Figure 3 is an example of the grid on a torus where ${}_\alpha T_{2n\times 2m} \in \mathfrak{M}({}_\alpha T_{2n\times 2m})$ is partitioned via the blue lines and ${}_\beta T_{2n\times 2m} \in \mathfrak{M}({}_\beta T_{2n\times 2m})$ is partitioned via the red lines.

Let us now describe an entire time-step of a reversible Margolus rule-based cellular automaton. We have a multiverse $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ and a multiverse $\mathfrak{M}({}_\beta T_{2n\times 2m})$ and some process that constitutes a bijection between these sets that somehow results from a rule r that we described in the introduction. We need mathematical definitions and notation for this process. There are two essential components: the map $r \in R$ which extends to the bijection $\rho : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$ (or $\rho : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$) and the re-partitioning of the grid which is, as we will prove, a bijection $\phi : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$. We have therefore two bijections that compose the time-step, namely ρ and ϕ. Thus we have the time step $\phi \circ \rho : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$. A subsequent time-step would be composed of a map $\rho_2 : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$ extending from a Margolus rule r_2, not necessarily equal to r, and $\phi^{-1} : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$. We therefore write the second time-step as $\phi^{-1} \circ \rho_2 : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$. In this section we describe ϕ and prove that is a bijection. Furthermore we show how a Margolus rule extends to a bijection on a multiverse of $2n \times 2m$ grids.

2.1 Re-partitioning a Universe

Let's consider what the re-partitioning process looks like mathematically. We have a collection of universes, $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ with an arbitrary 2×2 block partition. We can write $T \in \mathfrak{M}({}_\alpha T_{2n\times 2m})$ as a matrix

$$T := \left[\begin{array}{cccc} \begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & \begin{bmatrix} x_{2,0} & x_{3,0} \\ x_{2,1} & x_{3,1} \end{bmatrix} \dots & & \begin{bmatrix} x_{2n-2,0} & x_{2n-1,0} \\ x_{2n-2,1} & x_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} x_{0,2} & x_{1,2} \\ x_{0,3} & x_{1,3} \end{bmatrix} & \begin{bmatrix} x_{2,2} & x_{3,2} \\ x_{2,3} & x_{3,3} \end{bmatrix} \dots & & \begin{bmatrix} x_{2n-2,2} & x_{2n-1,2} \\ x_{2n-2,3} & x_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} x_{0,2m-2} & x_{1,2m-2} \\ x_{0,2m-1} & x_{1,2m-1} \end{bmatrix} & \begin{bmatrix} x_{2,2m-2} & x_{3,2m-2} \\ x_{2,2m-1} & x_{3,2m-1} \end{bmatrix} & \dots & \begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix} \end{array}\right], x_{i,j} \in \{0,1\}$$

where each sub-matrix represents a Margolus neighborhood of the grid. Throughout this paper, where the partitioning is not relevant, we may exclude the sub-matrix notation from our calculations. We want to capture the idea of the shift of the grid mathematically. That is, we want to construct a map that, while holding the partition sub-matrices in place, shifts the cells on the grid in order to convey an algebraic description of the visual provided in the introduction (see Figure 2, 3).

Consider

$$\phi := \begin{pmatrix} 0 & 1 & 0 & 0 & 0\dots & 0 \\ 0 & 0 & 1 & 0 & 0\dots & 0 \\ 0 & 0 & 0 & 1 & 0\dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0\dots & 1 \\ 1 & 0 & 0 & 0 & 0\dots & 0 \end{pmatrix}$$

such that

$$\phi(x) = \begin{pmatrix} 0 & 1 & 0\dots & 0 \\ 0 & 0 & 1\dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0\dots & 0 \end{pmatrix} \begin{pmatrix} x_{0,0} \\ x_{1,0} \\ \vdots \\ x_{2n-1,2m-1} \end{pmatrix}$$

We can present this map element-wise as

$$\phi : x_{i,j} \mapsto x_{i+1(\text{mod 2n}),j+1(\text{mod 2m})}.$$

Note that although ϕ can be written as a map on the elements of a universe, this in turn is a map on matrices (universes) which constitutes, as we will prove, a bijection $\phi : \mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$. What we really want to do with ϕ is shift the sub-matrices themselves. That is, to be true to the idea of re-partitioning, we would want to define ϕ such that the sub-matrices are shifted without moving the elements. Instead we shift the elements. As we are working with grids on tori, this is equivalent.[2]

We therefore leave $\mathfrak{M}({}_{\beta}T_{2n\times 2m})$ to be defined as the image of $\mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ under ϕ.

Theorem 2.1. *The re-partitioning map ϕ is a bijection $\mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$.*

Proof. Consider ${}_{\alpha}T_{2n\times 2m} \in \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ as a matrix of values,

$$\begin{bmatrix}
\begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & \begin{bmatrix} x_{2,0} & x_{3,0} \\ x_{2,1} & x_{3,1} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,0} & x_{2n-1,0} \\ x_{2n-2,1} & x_{2n-1,1} \end{bmatrix} \\
\begin{bmatrix} x_{0,2} & x_{1,2} \\ x_{0,3} & x_{1,3} \end{bmatrix} & \begin{bmatrix} x_{2,2} & x_{3,2} \\ x_{2,3} & x_{3,3} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,2} & x_{2n-1,2} \\ x_{2n-2,3} & x_{2n-1,3} \end{bmatrix} \\
\vdots & \vdots & \ddots & \vdots \\
\begin{bmatrix} x_{0,2m-2} & x_{1,2m-2} \\ x_{0,2m-1} & x_{1,2m-1} \end{bmatrix} & \begin{bmatrix} x_{2,2m-2} & x_{3,2m-2} \\ x_{2,2m-1} & x_{3,2m-1} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix}
\end{bmatrix}$$

where $x_{i,j}, 1 \leq i < 2n, j < 2m$ takes a value either 0 or 1.

Note that ϕ is independent of the partitioning scheme.[3] For this reason we are going to represent universes without the MN partition drawn-in in order to simplify computation but the reader should note that an arbitrary partition exists.

Take $T_j, T_k \in \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$. We want to show that ϕ is well-defined. Since $T_j, T_k \in \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$, we can write

---

[2] For example, we would want to take a grid $\begin{bmatrix} \begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & \ddots & \cdots \\ \vdots & & \begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix} \end{bmatrix}$ and apply a map so that $\begin{bmatrix} x_{1,1} & x_{2,1} \\ x_{1,2} & x_{2,2} \end{bmatrix}$ is a sub-matrix in the new partitioning scheme. Ideally the brackets themselves would move, but since, again, we are on a torus, shifting the cells is equivalent to shifting the brackets.

[3] Indeed ϕ is independent of the partitioning scheme because ϕ extends from ϕ_o which acts on cells, not on Margolus neighborhoods. That is, since the values of the cells themselves are not affected by the partition, ϕ_o is independent of the partitioning scheme.

$$T_j := \begin{bmatrix} j_{0,0} & j_{1,0} & j_{2,0} & \cdots & j_{2n-1,0} \\ j_{0,1} & j_{1,1} & j_{2,1} & \cdots & j_{2n-1,1} \\ j_{0,2} & j_{1,2} & j_{2,2} & \cdots & j_{2n-1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ j_{0,2m-1} & j_{1,2m-1} & j_{2,2m-1} & \cdots & j_{2n-1,2m-1} \end{bmatrix}, j_{p,q} \in \{0,1\}$$

and

$$T_k := \begin{bmatrix} k_{0,0} & k_{1,0} & k_{2,0} & \cdots & k_{2n-1,0} \\ k_{0,1} & k_{1,1} & k_{2,1} & \cdots & k_{2n-1,1} \\ k_{0,2} & k_{1,2} & k_{2,2} & \cdots & k_{2n-1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{0,2m-1} & k_{1,2m-1} & k_{2,2m-1} & \cdots & k_{2n-1,2m-1} \end{bmatrix}, k_{p,q} \in \{0,1\}.$$

Consider

$$\phi(T_j) = \phi \begin{bmatrix} j_{0,0} & j_{1,0} & j_{2,0} & \cdots & j_{2n-1,0} \\ j_{0,1} & j_{1,1} & j_{2,1} & \cdots & j_{2n-1,1} \\ j_{0,2} & j_{1,2} & j_{2,2} & \cdots & j_{2n-1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ j_{0,2m-1} & j_{1,2m-1} & j_{2,2m-1} & \cdots & j_{2n-1,2m-1} \end{bmatrix} = \begin{bmatrix} j_{1,1} & j_{2,1} & j_{3,1} & \cdots & j_{0,1} \\ j_{1,2} & j_{2,2} & j_{3,2} & \cdots & j_{0,2} \\ j_{1,3} & j_{2,3} & j_{3,3} & \cdots & j_{0,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ j_{1,0} & j_{2,0} & j_{3,0} & \cdots & j_{0,0} \end{bmatrix}$$

and

$$\phi(T_k) = \phi \begin{bmatrix} k_{0,0} & k_{1,0} & k_{2,0} & \cdots & k_{2n-1,0} \\ k_{0,1} & k_{1,1} & k_{2,1} & \cdots & k_{2n-1,1} \\ k_{0,2} & k_{1,2} & k_{2,2} & \cdots & k_{2n-1,2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{0,2m-1} & k_{1,2m-1} & k_{2,2m-1} & \cdots & k_{2n-1,2m-1} \end{bmatrix} = \begin{bmatrix} k_{1,1} & k_{2,1} & k_{3,1} & \cdots & k_{0,1} \\ k_{1,2} & k_{2,2} & k_{3,2} & \cdots & k_{0,2} \\ k_{1,3} & k_{2,3} & k_{3,3} & \cdots & k_{0,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{1,0} & k_{2,0} & k_{3,0} & \cdots & k_{0,0} \end{bmatrix}.$$

We see that $\phi(T_k) = \phi(T_j)$ if and only if $T_j = T_k$. Suppose $T_j = T_k$, then $k_{a,b} = j_{a,b}$ for all $a \in \{0,..,2n\}, b \in \{0,...,2m\}$. It follows that $k_{a+1(mod 2n),b+1(mod 2m)} = j_{a+1(\text{mod } 2\text{n}),b+1(\text{mod } 2\text{m})}$ for all a, b and therefore $\phi(T_j) = \phi(T_k)$. This argument reverses to show that if $\phi(T_j) = \phi(T_k)$, then $T_j = T_k$. This shows that ϕ is well-defined and injective. Now we must only show that ϕ is surjective. Since ϕ is an injective function from one finite set to another and these two sets are of the same size, we conclude that ϕ must also be a surjection by the pigeonhole principle[7][14, page 2]. We conclude that ϕ is a bijection.

□

2.2 Extending Reversible Rules to Bijections on a Multiverse

We know that if a Margolus rule is reversible, then the rule extends to a reversible cellular automaton. That is, the behavior of the entire grid, $\rho : \mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$, is reversible.

Theorem 2.2. *A Margolus CA is reversible on the entire grid if and only if the local rule is reversible [6].*

We will nonetheless prove this in order to better exhibit the complete Margolus time-step. Now we have the necessary structure to prove, in the context of our set-theoretic interpretation, that $r \in R$ extends to a bijection on the multiverse $\mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ and further more we want to show that the entire time-step composes a bijection $\mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$.

We represent $T \in \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ by the matrix

$$\left[\begin{array}{cccc} \begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & \begin{bmatrix} x_{2,0} & x_{3,0} \\ x_{2,1} & x_{3,1} \end{bmatrix} \cdots & & \begin{bmatrix} x_{2n-2,0} & x_{2n-1,0} \\ x_{2n-2,1} & x_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} x_{0,2} & x_{1,2} \\ x_{0,3} & x_{1,3} \end{bmatrix} & \begin{bmatrix} x_{2,2} & x_{3,2} \\ x_{2,3} & x_{3,3} \end{bmatrix} \cdots & & \begin{bmatrix} x_{2n-2,2} & x_{2n-1,2} \\ x_{2n-2,3} & x_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} x_{0,2m-2} & x_{1,2m-2} \\ x_{0,2m-1} & x_{1,2m-1} \end{bmatrix} & \begin{bmatrix} x_{2,2m-2} & x_{3,2m-2} \\ x_{2,2m-1} & x_{3,2m-1} \end{bmatrix} \cdots & & \begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix} \end{array}\right], x_{i,j} \in \{0,1\}$$

where each sub-matrix is a Margolus neighborhood and therefore compute the image of T under ρ to be

$$\left[\begin{array}{cccc} r\begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & r\begin{bmatrix} x_{2,0} & x_{3,0} \\ x_{2,1} & x_{3,1} \end{bmatrix} \cdots & & r\begin{bmatrix} x_{2n-2,0} & x_{2n-1,0} \\ x_{2n-2,1} & x_{2n-1,1} \end{bmatrix} \\ r\begin{bmatrix} x_{0,2} & x_{1,2} \\ x_{0,3} & x_{1,3} \end{bmatrix} & r\begin{bmatrix} x_{2,2} & x_{3,2} \\ x_{2,3} & x_{3,3} \end{bmatrix} \cdots & & r\begin{bmatrix} x_{2n-2,2} & x_{2n-1,2} \\ x_{2n-2,3} & x_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ r\begin{bmatrix} x_{0,2m-2} & x_{1,2m-2} \\ x_{0,2m-1} & x_{1,2m-1} \end{bmatrix} & r\begin{bmatrix} x_{2,2m-2} & x_{3,2m-2} \\ x_{2,2m-1} & x_{3,2m-1} \end{bmatrix} \cdots & & r\begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix} \end{array}\right].$$

Recall that $r : \{0,1\}^4 \to \{0,1\}^4$ and take

$$r\begin{bmatrix} x_{i,j} & x_{i+1,j} \\ x_{i,j+1} & x_{i+1,j+1} \end{bmatrix} = \begin{bmatrix} y_{i,j} & y_{i+1,j} \\ y_{i,j+1} & y_{i+1,j+1} \end{bmatrix}$$

for $i \in \{0, 2, 4, ..., 2n-2\}$ and $j \in \{0, 2, 4, ..., 2m-2\}$. Therefore r applied to the neighborhoods of T gives us

$$\rho(T) = \left[\begin{array}{cccc} \begin{bmatrix} y_{0,0} & y_{1,0} \\ y_{0,1} & y_{1,1} \end{bmatrix} & \begin{bmatrix} y_{2,0} & y_{3,0} \\ y_{2,1} & y_{3,1} \end{bmatrix} \cdots & & \begin{bmatrix} y_{2n-2,0} & y_{2n-1,0} \\ y_{2n-2,1} & y_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} y_{0,2} & y_{1,2} \\ y_{0,3} & y_{1,3} \end{bmatrix} & \begin{bmatrix} y_{2,2} & y_{3,2} \\ y_{2,3} & y_{3,3} \end{bmatrix} \cdots & & \begin{bmatrix} y_{2n-2,2} & y_{2n-1,2} \\ y_{2n-2,3} & y_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} y_{0,2m-2} & y_{1,2m-2} \\ y_{0,2m-1} & y_{1,2m-1} \end{bmatrix} & \begin{bmatrix} y_{2,2m-2} & y_{3,2m-2} \\ y_{2,2m-1} & y_{3,2m-1} \end{bmatrix} \cdots & & \begin{bmatrix} y_{2n-2,2m-2} & y_{2n-1,2m-2} \\ y_{2n-2,2m-1} & y_{2n-1,2m-1} \end{bmatrix} \end{array}\right].$$

We claim that in this way, r is a bijection $\rho : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$.

Theorem 2.3. *For all reversible Margolus rules r, the map r, when applied to a multiverse $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ of $2n\times 2m$ partitioned grids on tori in the manner heretofore defined, is a bijection $\rho : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$.*

Proof. Let $R \ni r : \{0,1\}^4 \to \{0,1\}^4$ and let $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ be the collection of all possible configurations of a $2n \times 2m$ partitioned grid on a torus. Take $T_i, T_j \in \mathfrak{M}({}_\alpha T_{2n\times 2m})$ and take ρ to be the map $\mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$ induced by r. As $T_i, T_j \in \mathfrak{M}({}_\alpha T_{2n\times 2m})$, let

$$T_i := \left[\begin{array}{cccc} \begin{bmatrix} i_{0,0} & i_{1,0} \\ i_{0,1} & i_{1,1} \end{bmatrix} & \begin{bmatrix} i_{2,0} & i_{3,0} \\ i_{2,1} & i_{3,1} \end{bmatrix} \cdots & & \begin{bmatrix} i_{2n-2,0} & i_{2n-1,0} \\ i_{2n-2,1} & i_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} i_{0,2} & i_{1,2} \\ i_{0,3} & i_{1,3} \end{bmatrix} & \begin{bmatrix} i_{2,2} & i_{3,2} \\ i_{2,3} & i_{3,3} \end{bmatrix} \cdots & & \begin{bmatrix} i_{2n-2,2} & i_{2n-1,2} \\ i_{2n-2,3} & i_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} i_{0,2m-2} & i_{1,2m-2} \\ i_{0,2m-1} & i_{1,2m-1} \end{bmatrix} & \begin{bmatrix} i_{2,2m-2} & i_{3,2m-2} \\ i_{2,2m-1} & i_{3,2m-1} \end{bmatrix} \cdots & & \begin{bmatrix} i_{2n-2,2m-2} & i_{2n-1,2m-2} \\ i_{2n-2,2m-1} & i_{2n-1,2m-1} \end{bmatrix} \end{array}\right], i_{a,b} \in \{0,1\}$$

and

$$T_j := \left[\begin{array}{cccc} \begin{bmatrix} j_{0,0} & j_{1,0} \\ j_{0,1} & j_{1,1} \end{bmatrix} & \begin{bmatrix} j_{2,0} & j_{3,0} \\ j_{2,1} & j_{3,1} \end{bmatrix} \cdots & & \begin{bmatrix} j_{2n-2,0} & j_{2n-1,0} \\ j_{2n-2,1} & j_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} j_{0,2} & j_{1,2} \\ j_{0,3} & j_{1,3} \end{bmatrix} & \begin{bmatrix} j_{2,2} & j_{3,2} \\ j_{2,3} & j_{3,3} \end{bmatrix} \cdots & & r\begin{bmatrix} j_{2n-2,2} & j_{2n-1,2} \\ j_{2n-2,3} & j_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} j_{0,2m-2} & j_{1,2m-2} \\ j_{0,2m-1} & j_{1,2m-1} \end{bmatrix} & \begin{bmatrix} j_{2,2m-2} & j_{3,2m-2} \\ j_{2,2m-1} & j_{3,2m-1} \end{bmatrix} \cdots & & \begin{bmatrix} j_{2n-2,2m-2} & j_{2n-1,2m-2} \\ j_{2n-2,2m-1} & j_{2n-1,2m-1} \end{bmatrix} \end{array}\right], j_{a,b} \in \{0,1\}.$$

Then we have

$$\rho(T_i) = \begin{bmatrix} r\begin{bmatrix} i_{0,0} & i_{1,0} \\ i_{0,1} & i_{1,1} \end{bmatrix} & r\begin{bmatrix} i_{2,0} & i_{3,0} \\ i_{2,1} & i_{3,1} \end{bmatrix} & \cdots & r\begin{bmatrix} i_{2n-2,0} & i_{2n-1,0} \\ i_{2n-2,1} & i_{2n-1,1} \end{bmatrix} \\ r\begin{bmatrix} i_{0,2} & i_{1,2} \\ i_{0,3} & i_{1,3} \end{bmatrix} & r\begin{bmatrix} i_{2,2} & i_{3,2} \\ i_{2,3} & i_{3,3} \end{bmatrix} & \cdots & r\begin{bmatrix} i_{2n-2,2} & i_{2n-1,2} \\ i_{2n-2,3} & i_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ r\begin{bmatrix} i_{0,2m-2} & i_{1,2m-2} \\ i_{0,2m-1} & i_{1,2m-1} \end{bmatrix} & r\begin{bmatrix} i_{2,2m-2} & i_{3,2m-2} \\ i_{2,2m-1} & i_{3,2m-1} \end{bmatrix} & \cdots & r\begin{bmatrix} i_{2n-2,2m-2} & i_{2n-1,2m-2} \\ i_{2n-2,2m-1} & i_{2n-1,2m-1} \end{bmatrix} \end{bmatrix}$$

and

$$\rho(T_j) = \begin{bmatrix} r\begin{bmatrix} j_{0,0} & j_{1,0} \\ j_{0,1} & j_{1,1} \end{bmatrix} & r\begin{bmatrix} j_{2,0} & j_{3,0} \\ j_{2,1} & j_{3,1} \end{bmatrix} & \cdots & r\begin{bmatrix} j_{2n-2,0} & j_{2n-1,0} \\ j_{2n-2,1} & j_{2n-1,1} \end{bmatrix} \\ r\begin{bmatrix} j_{0,2} & j_{1,2} \\ j_{0,3} & j_{1,3} \end{bmatrix} & r\begin{bmatrix} j_{2,2} & j_{3,2} \\ j_{2,3} & j_{3,3} \end{bmatrix} & \cdots & r\begin{bmatrix} j_{2n-2,2} & j_{2n-1,2} \\ j_{2n-2,3} & j_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ r\begin{bmatrix} j_{0,2m-2} & j_{1,2m-2} \\ j_{0,2m-1} & j_{1,2m-1} \end{bmatrix} & r\begin{bmatrix} j_{2,2m-2} & j_{3,2m-2} \\ j_{2,2m-1} & j_{3,2m-1} \end{bmatrix} & \cdots & r\begin{bmatrix} j_{2n-2,2m-2} & j_{2n-1,2m-2} \\ j_{2n-2,2m-1} & j_{2n-1,2m-1} \end{bmatrix} \end{bmatrix}.$$

Recalling that r is a bijection, we see that $T_i = T_j$ if and only if $\rho(T_i) = \rho(T_j)$. That is, $\rho(T_i) = \rho(T_j)$ if and only if every sub-matrix of $\rho(T_i)$ is equal to that of $\rho(T_j)$. Since r is a bijection, every sub-matrix of $\rho(T_j)$ and $\rho(T_i)$ will be equal if and only if the pre-image of each sub-matrix, the sub-matrices of T_j and T_i, are equal. Finally, $T_i = T_j$ if and only if their sub-matrices are equal, thus $T_j = T_i$ if and only if $\rho(T_i) = \rho(T_j)$. We must still prove surjectivity of ρ. Let $T_k \in \mathfrak{M}({}_\alpha T_{2n\times 2m})$, so

$$T_k := \begin{bmatrix} \begin{bmatrix} k_{0,0} & k_{1,0} \\ k_{0,1} & k_{1,1} \end{bmatrix} & \begin{bmatrix} k_{2,0} & k_{3,0} \\ k_{2,1} & k_{3,1} \end{bmatrix} & \cdots & \begin{bmatrix} k_{2n-2,0} & k_{2n-1,0} \\ k_{2n-2,1} & k_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} k_{0,2} & k_{1,2} \\ k_{0,3} & k_{1,3} \end{bmatrix} & \begin{bmatrix} k_{2,2} & k_{3,2} \\ k_{2,3} & k_{3,3} \end{bmatrix} & \cdots & \begin{bmatrix} k_{2n-2,2} & k_{2n-1,2} \\ k_{2n-2,3} & k_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} k_{0,2m-2} & k_{1,2m-2} \\ k_{0,2m-1} & k_{1,2m-1} \end{bmatrix} & \begin{bmatrix} k_{2,2m-2} & k_{3,2m-2} \\ k_{2,2m-1} & k_{3,2m-1} \end{bmatrix} & \cdots & \begin{bmatrix} k_{2n-2,2m-2} & k_{2n-1,2m-2} \\ k_{2n-2,2m-1} & k_{2n-1,2m-1} \end{bmatrix} \end{bmatrix}, k_{a,b} \in \{0,1\}.$$

Let $r \in R$. Since r is a bijection $\{0,1\}^4 \to \{0,1\}^4$, we know that r^{-1} is also a bijection $\{0,1\}^4 \to \{0,1\}^4$. We claim that

$$T'_k := \begin{bmatrix} r^{-1}\begin{bmatrix} k_{0,0} & k_{1,0} \\ k_{0,1} & k_{1,1} \end{bmatrix} & r^{-1}\begin{bmatrix} k_{2,0} & k_{3,0} \\ k_{2,1} & k_{3,1} \end{bmatrix} & \cdots & r^{-1}\begin{bmatrix} k_{2n-2,0} & k_{2n-1,0} \\ k_{2n-2,1} & k_{2n-1,1} \end{bmatrix} \\ r^{-1}\begin{bmatrix} k_{0,2} & k_{1,2} \\ k_{0,3} & k_{1,3} \end{bmatrix} & r^{-1}\begin{bmatrix} k_{2,2} & k_{3,2} \\ k_{2,3} & k_{3,3} \end{bmatrix} & \cdots & r^{-1}\begin{bmatrix} k_{2n-2,2} & k_{2n-1,2} \\ k_{2n-2,3} & k_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ r^{-1}\begin{bmatrix} k_{0,2m-2} & k_{1,2m-2} \\ k_{0,2m-1} & k_{1,2m-1} \end{bmatrix} & r^{-1}\begin{bmatrix} k_{2,2m-2} & k_{3,2m-2} \\ k_{2,2m-1} & k_{3,2m-1} \end{bmatrix} & \cdots & r^{-1}\begin{bmatrix} k_{2n-2,2m-2} & k_{2n-1,2m-2} \\ k_{2n-2,2m-1} & k_{2n-1,2m-1} \end{bmatrix} \end{bmatrix}$$

is an element of $\mathfrak{M}({}_\alpha T_{2n\times 2m})$. Well, since r^{-1} is a bijection, let

$$r^{-1}\begin{bmatrix} k_{i,j} & k_{i+1,j} \\ k_{i,j+1} & k_{i+1,j+1} \end{bmatrix} = \begin{bmatrix} y_{i,j} & y_{i+1,j} \\ y_{i,j+1} & y_{i+1,j+1} \end{bmatrix}.$$

Thus

$$T'_k = \begin{bmatrix} \begin{bmatrix} y_{0,0} & y_{1,0} \\ y_{0,1} & y_{1,1} \end{bmatrix} & \begin{bmatrix} y_{2,0} & y_{3,0} \\ y_{2,1} & y_{3,1} \end{bmatrix} & \cdots & \begin{bmatrix} y_{2n-2,0} & y_{2n-1,0} \\ y_{2n-2,1} & y_{2n-1,1} \end{bmatrix} \\ \begin{bmatrix} y_{0,2} & y_{1,2} \\ y_{0,3} & y_{1,3} \end{bmatrix} & \begin{bmatrix} y_{2,2} & y_{3,2} \\ y_{2,3} & y_{3,3} \end{bmatrix} & \cdots & \begin{bmatrix} y_{2n-2,2} & y_{2n-1,2} \\ y_{2n-2,3} & y_{2n-1,3} \end{bmatrix} \\ \vdots & \vdots & \ddots & \vdots \\ \begin{bmatrix} y_{0,2m-2} & y_{1,2m-2} \\ y_{0,2m-1} & y_{1,2m-1} \end{bmatrix} & \begin{bmatrix} y_{2,2m-2} & y_{3,2m-2} \\ y_{2,2m-1} & y_{3,2m-1} \end{bmatrix} & \cdots & \begin{bmatrix} y_{2n-2,2m-2} & y_{2n-1,2m-2} \\ y_{2n-2,2m-1} & y_{2n-1,2m-1} \end{bmatrix} \end{bmatrix},$$

which is an element of $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ since it is a $2n \times 2m$ grid with cell values of 0 or 1 and the partition has not been changed. We conclude that $\rho(T'_k) = T_k$ and that ρ is a surjection thus a bijection.

$\square$

Now that we have established our preliminary proofs and have established our notation and syntax for working with Margolus rules, Margolus neighborhoods, multiverses, our maps, etc., we can begin to uncover the group-theoretic structure that arises from Margolus rules acting on multiverses.

3 Group-theoretic Structure of the Pairwise-Iterated Margolus Rule

Let $\mathfrak{M}({}_\alpha T_{2n\times 2m})$ be the multiverse of $2n \times 2m$ grids on tori for some fixed $n, m \in \mathbb{N}$ with an arbitrary 2×2 block partitioning and let $\mathfrak{M}({}_\beta T_{2n\times 2m})$ be the multiverse of the same grid but with the alternate partitioning $(\phi(\mathfrak{M}({}_\alpha T_{2n\times 2m}))$. Let $r_1, r_2 \in R$ with extensions $\rho_1 : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$, $\rho_2 : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$, respectively, and let $f = \phi \circ \rho_1$ and $g = \phi^{-1} \circ \rho_2$.

$$\begin{tikzcd}
\mathfrak{M}({}_\alpha T_{2n\times 2m}) \arrow[r, "\rho_1"] \arrow[dr, "f"'] \arrow[rr, bend left, "g\circ f"] & \mathfrak{M}({}_\alpha T_{2n\times 2m}) \arrow[d, "\phi"] & \mathfrak{M}({}_\alpha T_{2n\times 2m}) \\
 & \mathfrak{M}({}_\beta T_{2n\times 2m}) \arrow[r, "\rho_2"] \arrow[ur, "g"] & \mathfrak{M}({}_\beta T_{2n\times 2m}) \arrow[u, "\phi^{-1}"']
\end{tikzcd}$$

So we have two bijections $f : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$ and $g : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$ that stem from reversible Margolus rules.

Theorem 3.1. *The set generated by pairs of bijective functions $f : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$,$g\mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$, that extend from reversible Margolus rules paired with function composition forms a group, G.*[4]

Proof. Take $\rho_1, \rho_2, \rho_3 \in G$. We know that bijective functions are always associative. Hence we have $\rho_1(\rho_2\rho_3) = (\rho_1\rho_2)\rho_3$.

Take $\rho, \rho' \in G$. Then $\rho = g_0 \circ f_0 \circ .... \circ g_p \circ f_p$ and $\rho' = g'_0 \circ f'_0 \circ ... \circ g'_j \circ f'_j$ for pairs $g_i \circ f_i$, $g'_i \circ f'_i$ of bijections stemming from reversible Margolus rules. Then $\rho\circ\rho' = (g_0\circ f_0\circ ...\circ g_p\circ f_p)\circ(g'_0\circ f'_0\circ ...\circ g'_j\circ f'_j) = (g_0\circ f_0)\circ ...\circ(g_p\circ f_p)\circ(g'_0\circ f'_0)...\circ(g'_j)\circ f'_j)$ is generated by pairs of bijections of the form $g\circ f$ with g, f stemming from reversible Margolus rules and therefore $\rho \circ \rho' \in G$.

We must show that G has an identity Id. We know that the identity map Id_M on the Margolus neighborhood is a reversible Margolus rule. Put $r_1 = r_2 = Id_M$ with extensions $r'_1 : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m}), r'_2 : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m})$. The extension of the Margolus rule identity is the identity map on the entire grid.

[4] G = ($\langle g \circ f\rangle | g$ and f extend from reversible Margolus rules with $f : \mathfrak{M}({}_\alpha T_{2n\times 2m}) \to \mathfrak{M}({}_\beta T_{2n\times 2m})$ and $g : \mathfrak{M}({}_\beta T_{2n\times 2m}) \to \mathfrak{M}({}_\alpha T_{2n\times 2m}), \circ)$

[5] Now we can put $g = \phi \circ r_2'$ and $f = \phi^{-1} \circ r_1'$ and conclude $g \circ f : \mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ is the identity element $Id \in G$.[6]

Now take $\rho \in G$; we want to show that $\rho^{-1} \in G$. Since G is generated by pairs $g \circ f$, we need only show that $(g \circ f)^{-1} \in G$. That is, if $G \ni \rho = g \circ f$, then ρ is an arbitrary generator of G, so if $\rho^{-1} \in G$ then it follows that $g^{-1} \in G$ for all $g \in G$.

Say $\rho = g \circ f$ for some bijections $f : \mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$ and $g : \mathfrak{M}({}_{\beta}T_{2n\times 2m}) \to \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ extended from some reversible Margolus rules r_f and r_g, respectively such that $f = \phi \circ r_f$ and $g = \phi^{-1} \circ r_g$. Both $f^{-1} : \mathfrak{M}({}_{\beta}T_{2n\times 2m}) \to \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$ and $g^{-1} : \mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$ are also bijections stemming from reversible rules, namely r_f^{-1} and r_g^{-1} therefore $f^{-1} \circ g^{-1} \in G$. Take Id_α to be the identity map $\mathfrak{M}({}_{\alpha}T_{2n\times 2m}) \to \mathfrak{M}({}_{\alpha}T_{2n\times 2m})$, a reversible conservative Maroglus rule, and likewise let Id_β be the identity map $\mathfrak{M}({}_{\beta}T_{2n\times 2m}) \to \mathfrak{M}({}_{\beta}T_{2n\times 2m})$. Then let $g^{-1} = \phi \circ Id_\alpha \circ \phi^{-1} \circ r_g^{-1} \circ \phi \circ Id_\alpha$ and $f^{-1} = \phi^{-1} \circ Id_\beta \circ \phi \circ r_f^{-1} \circ \phi^{-1} \circ Id_\beta$. Say $f^{-1} \circ g^{-1} = b$ then

$$\begin{aligned}
b \circ \rho &= (f^{-1} \circ g^{-1}) \circ (g \circ f) \\
&= f^{-1} \circ (g^{-1} \circ g) \circ f \\
&= (\phi^{-1} \circ Id_\beta \circ \phi \circ r_f^{-1} \circ \phi^{-1} \circ Id_\beta) \circ ((\phi \circ Id_\alpha \circ \phi^{-1} \circ r_g^{-1} \circ \phi \circ Id_\alpha) \circ (\phi^{-1} \circ r_g)) \circ (\phi \circ r_f) \\
&= Id \\
&= ((\phi \circ r_f) \circ (\phi^{-1} \circ r_g)) \circ ((\phi \circ Id_\alpha \circ \phi^{-1} \circ r_g^{-1} \circ \phi \circ Id_\alpha) \circ (\phi^{-1} \circ Id_\beta \circ \phi \circ r_f^{-1} \circ \phi^{-1} \circ Id_\beta)) \\
&= (g \circ f) \circ (f^{-1} \circ g^{-1}) \\
&= \rho \circ b.
\end{aligned}$$

Therefore $\rho \circ b = Id = b \circ \rho$ and thus $b = \rho^{-1}$ and $\rho^{-1} \in G$.

$\square$

Now that we know G to be a group, we can furthermore show that G acts on the multiverse of partitioned grids for some fixed n, m.

[5] Let Id_M be the identity map on the multiverse of Margolus neighborhoods. Then every 2×2 block of the grid is mapped to itself. First of all, this map is reversible. Since every MN is mapped to itself, then so is every cell in the grid. Thus the entire grid is mapped to itself by r_1' and r_2'.

[6] Take $g \in G$, then $Id \circ g = g = g \circ Id$.

3.1 Group Action

We drop the "α" and denote this set as $\mathfrak{M}(T_{2n\times 2m})$ as the partitioning is arbitrary and we are now "skipping over" the alternate partitioning. We take G to be the group defined in Theorem 3.1 and claim that

$$G \times \mathfrak{M}(T_{2n\times 2m}) \to \mathfrak{M}(T_{2n\times 2m})$$
$$g * x \mapsto g(x)$$

$g \in G$, $x \in \mathfrak{M}(T_{2n\times 2m})$ is a group action.

Proof. For all $n, m \in \mathbb{N}$, $\mathfrak{M}(T_{2n\times 2m})$ is nonempty. We must only show associativity and that the identity in G stabilizes all elements of $\mathfrak{M}(T_{2n\times 2m})$. Let $g_1, g_2 \in G$ and $x \in \mathfrak{M}(T_{2n\times 2m})$. We know that g_1 and g_2 are bijections on the grid. As x is some universe, we have

$$g_1 : \mathfrak{M}(T_{2n\times 2m}) \to \mathfrak{M}(T_{2n\times 2m}), x \mapsto g_1(x)$$

$$g_2 : \mathfrak{M}(T_{2n\times 2m}) \to \mathfrak{M}(T_{2n\times 2m}), x \mapsto g_2(x).$$

In particular, g_1 and g_2 are both bijections on $\mathfrak{M}(T_{2n\times 2m})$ since they are composed of reversible rules which extend to bijective maps. Since g_1 and g_2 are bijections on $\mathfrak{M}(T_{2n\times 2m})$, they enjoy associativity on the domain $\mathfrak{M}(T_{2n\times 2m})$. Therefore

$$(g_1 * g_2) * x \equiv (g_1 \circ g_2)(x) = g_1(g_2(x)) \equiv g_1 * (g_2 * x)$$

We conclude that $(g_1 * g_2) * x = g_1 * (g_2 * x)$.

Let $Id \in G$ be the identity of G. This is the identity map, which as we have seen is an element of G. For $x \in \mathfrak{M}(T_{2n\times 2m})$, consider $Id * x$. Thus we have $Id * x \equiv Id(x) : x \mapsto x$. We conclude that $Id * x = x$ and in closing that $G \times \mathfrak{M}(T_{2n\times 2m}) \to \mathfrak{M}(T_{2n\times 2m})$ is a group action.

$\square$

4 Group Action of the Lonely Universes

We have shown that we have a group action

$$G \times \mathfrak{M}(T_{2n\times 2m}) \to \mathfrak{M}(T_{2n\times 2m})$$

where G is the group given by the set of pairs of bijections on the $2n \times 2m$ grid on a torus that extend from reversible Margolus rules along with function composition and $\mathfrak{M}(T_{2n\times 2m})$ is the multiverse of $2n \times 2m$ grids on tori (whose cells assume a value either 0 or 1).

We are interested more specifically in the group action

$$C \times \mathfrak{M}(T^L_{2n\times 2m}) \to \mathfrak{M}(T^L_{2n\times 2m})$$

where C is the subgroup of G whose elements are bijections that extend from reversible, *conservative* rules and $\mathfrak{M}(T^L_{2n\times 2m})$ is the set of lonely universes $T^L_{2n\times 2m}$ (recall that a lonely universe is a grid with one living cell) for some n, m. Let's prove that C is indeed a group and, in particular, a subgroup of G.

Proposition 4.1. *The set generated by pairs of bijections $g \circ f : \mathfrak{M}(_\alpha T_{2n\times 2m}) \to \mathfrak{M}(_\alpha T_{2n\times 2m})$ for g and f extending from reversible, conservative Margolus rules paired with function composition is a subgroup of G.*

Proof. Let C be the group of bijections on $T^L_{2n\times 2m}$ that arise from reversible, conservative Margolus rules and let G be the group of bijections on $T_{2n\times 2m}$ stemming from reversible Margolus rules. Indeed the elements of C are a subset of those of G and in particular, $C \neq \emptyset$ for all n, m.

Let $c_1, c_2 \in C$. We want to show that $c_1 * c_2 \in C$. First of all, note that $c_1 * c_2 := c_1 \circ c_2$. We know that the composition of bijections is still a bijection. Therefore, since a map is a bijection if and only if it is reversible, $c_1 \circ c_2$ is reversible. We claim further that the composition of bijections induced by conservative rules will also be conservative. If the total number of living cells is preserved block-by-block, then the total number of living cells across the entire grid will be preserved. Furthermore, the shift preserves the number of living cells. We conclude that a time-step extending from a conservative Margolus rule is conservative across the entire grid. Now that we know the conservative property is extended to the entire grid, we can conclude that this is preserved by function composition. Finally, since $c_1 \circ c_2$ is both conservative and reversible, $c_1 \circ c_2 \in C$.

Now we show $c_1^{-1} \in C$. Given that c_1 is an extension of some Margolus rule, say Mc_1, take c_1^{-1} to be the extension of Mc_1^{-1}. This rule is both conservative and reversible by definition and thus c_1^{-1} will be both conservative and reversible.

Indeed, C is a subgroup of G. $\square$

Now that we know C is a subgroup of G, let's consider the group action of C acting on $\mathfrak{M}(T^L_{2n\times 2m})$.

Theorem 4.2. *Let $\mathfrak{M}(T^L_{2n\times 2m})$ be the multiverse of lonely universes $T^L_{2n\times 2m}$ for some n, m, then for $c \in C$ and $x \in \mathfrak{M}(T^L_{2n\times 2m})$,*

$$\begin{gathered} C \times \mathfrak{M}(T^L_{2n\times 2m}) \to \mathfrak{M}(T^L_{2n\times 2m}) \\ c * x \mapsto c(x) \end{gathered}$$

is a group action.

Proof. We need to show that $Id \in C$ stabilizes all elements of $\mathfrak{M}(T^L_{2n\times 2m})$ and that the action is associative.

We know $Id \in C$ to be the identity map on the grid. So for any $x \in \mathfrak{M}(T^L_{2n\times 2m})$, we know $Id(x) \mapsto x$ and thus stabilizes the entire set.

As for associativity, we know that bijections on a set always have associativity. We follow the logic of the proof that the similar action of G on $\mathfrak{M}(T^L_{2n\times 2m})$ is indeed an action. Take $c_1, c_2 \in C$. So we write

$$(c_1 * c_2) * x \equiv (c_1 \circ c_2)(x) = c_1(c_2(x)) \equiv c_1 * (c_2 * x).$$

We conclude that

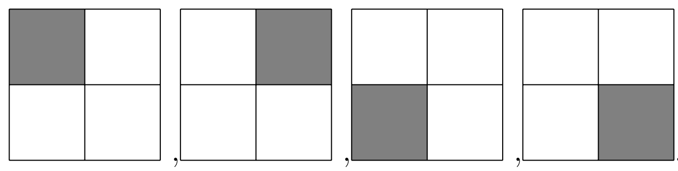
$$\begin{gathered} C \times \mathfrak{M}(T^L_{2n\times 2m}) \to \mathfrak{M}(T^L_{2n\times 2m}) \\ c * x \mapsto c(x) \end{gathered}$$

for all $c \in C, x \in \mathfrak{M}(T^L_{2n\times 2m})$ is indeed a group action. $\square$

4.1 Generators and Symmetry

We are interested in the bijections on the multiverse of lonely universes $\mathfrak{M}(T^L)$ for some fixed $n, m \in \mathbb{N}$ that arise from reversible, conservative Margolus rules. We want to know what these bijections and the resulting subgroup and group action structure look like.

Since we are dealing with conservative rules, we know that the empty MN, that is, the Margolus neighborhood with no living cells, has to map to itself at every time-step. Thus we can characterize universes uniquely by the position of the living cell. In a Margolus neighborhood we only have four possibilities to consider:



Reversible, conservative Margolus rules behave as permutations on these four elements. When we construct a map to be applied to every element of the multiverse, the map can be written as a composition of permutations of these four MN with intermediary shifts of the grid to represent the re-partitioning process, ϕ. Recall that C is the group of pairs of bijections that extend from reversible, conservative rules and that this process looks like

$$\begin{tikzcd}
\mathfrak{M}({}_\alpha T_{2n\times 2m}) \arrow[r, "\rho_1"] \arrow[dr, "f"'] \arrow[rr, bend left, "g\circ f"] & \mathfrak{M}({}_\alpha T_{2n\times 2m}) \arrow[d, "\phi"] & \mathfrak{M}({}_\alpha T_{2n\times 2m}) \\
& \mathfrak{M}({}_\beta T_{2n\times 2m}) \arrow[r, "\rho_2"] \arrow[ur, "g"] & \mathfrak{M}({}_\beta T_{2n\times 2m}) \arrow[u, "\phi^{-1}"']
\end{tikzcd}$$

In this way, these permutations that arise from reversible, conservative Margolus rules, along with the shift of the grid (re-partitioning), when taken sequentially are generators for all of the maps that we can construct from reversible, conservative Margolus rules. That is, all of the maps in the group C take the form of a Margolus rule extended to the grid, a shift, another Margolus rule extension, and an inverse shift. In this way we can construct explicit generators for the copy of C under the permutation representation. We explore this further after we develop the permutation representation of the action in Section 4.2.

Definition 6. "Let the group G act on the set A. For each fixed $g \in G$ we get a map σ_g defined by

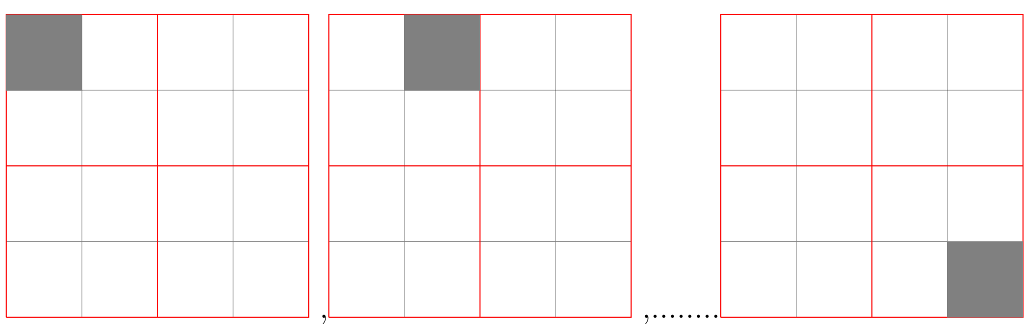
$$\sigma : A \to A$$
$$\sigma_g(a) = g * a.$$

... **(i)** for each fixed $g \in G$, σ_g is a permutation of A and **(ii)** the map from G to S_A defined by $g \mapsto \sigma_g$ is a homomorphism. The homomorphism from G to S_A given above is called the *permutation representation*[7] associated to the given action."[14, pages 42-43]

The permutations of these four elements, however, introduce, by symmetry, larger generators when we consider larger grids. Keep in mind that the permutations of these four 2×2 blocks are representative of the generators, but are not the generators themselves. In the same way that Margolus rules extend to bijections on the multiverse, so too the generators of the Margolus neighborhood permutations extend to permutations on multiverses of lonely universes.

This raises the question "which permutations arise on the multiverse of lonely universes when we act on the multiverse with maps that extend from reversible, conservative Margolus rules?" The first attempt to address this question was "does the group action act on the multiverse of lonely universes by all possible permutations?" For the 2×2 case, this turns out to be true. In the example that follows, we show that this is not true for the 4×4 case. After developing a few more tools, we, in Section 5, attempt to answer this question with the culmination of the ideas in this paper.

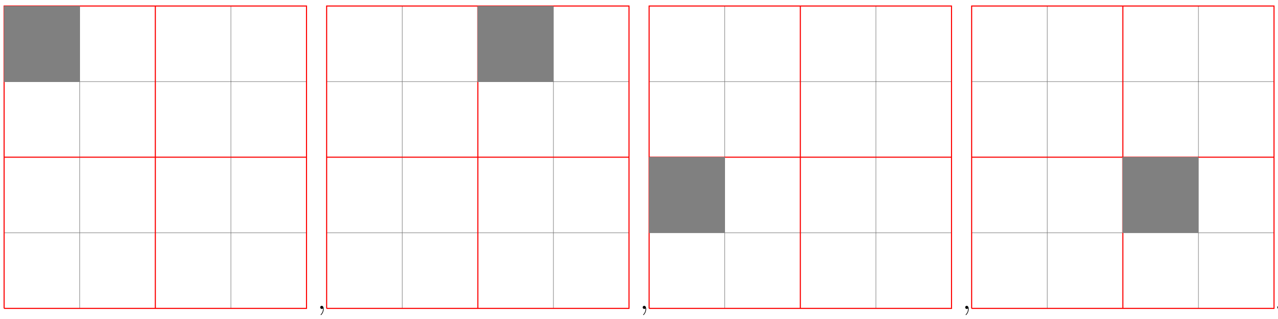
Example 4.1. Consider the multiverse of lonely universes of size 4×4. We have



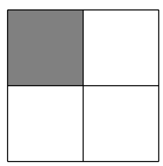
where the red lines represent the initial partition of the grid. The symmetry arises

[7]Note that while permutation representation is defined as a homomorphism, we often refer to the image of the homomorphism as the permutation representation.

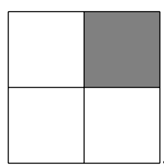
when we consider 4$-$*tuples* of elements such as



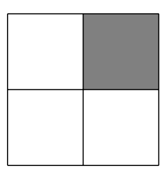
These four elements of the multiverse are acted on in the same way. That is, say a map $c \in C$ acts on the multiverse of lonely universes of size 4×4, denoted here $\mathfrak{M}(T^L_{4\times 4})$. This map c extends from a Margolus rule. Suppose c maps the MN



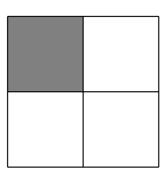
to



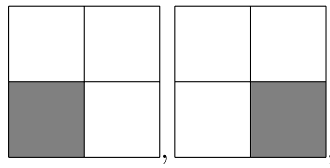
Since the rule c must be reversible, we must also map



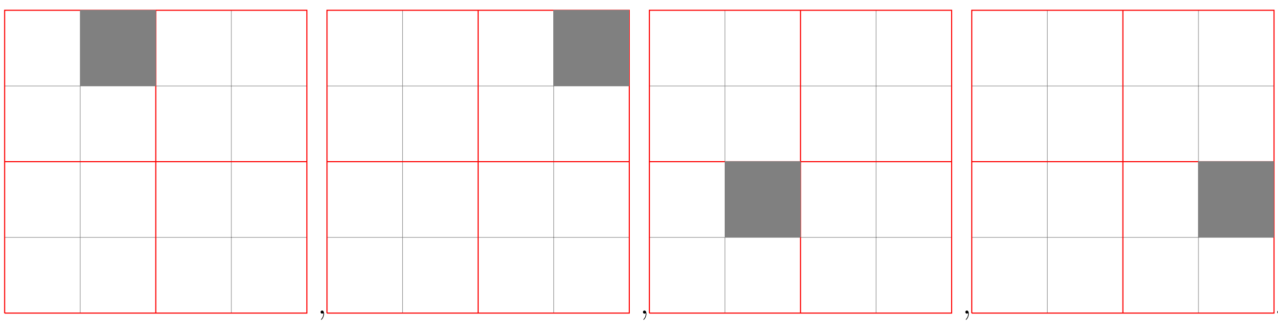
to another MN. For simplicity, we can map this MN to



resulting in a permutation, while mapping the other two MN,



to themselves. We end up with a permutation that swaps two MN and leaves the other two constant. As mentioned before, this is not the true generator for the action on the multiverse for n or m greater than 1, but this permutation tells us everything we need to know all the same. This one permutation extends to our "true" generator by the symmetric properties of the partitioned grid. Our $4-$*tuple* of universes gets mapped to the $4-$*tuple*



We see that we are in fact permuting four elements of the multiverse. That is, we have four transpositions. This is because these universes can be equated by a translation on the multiverse of some vertical measure by some multiple of $2m$ or horizontally by some multiple of $2n$. This is the idea of symmetry in the group action; this is where the symmetric group meets the Margolus rule structure.

4.2 Permutation Representation

To simplify discussion, we will shift from working with this group action directly to working with its permutation representation. Fortunately, working with the action on lonely universes is rather intuitive; the same cannot be said of multiverses that allow universes with more than one living cell.

We know that $|\mathfrak{M}(T^L_{2n\times 2m})| = 4mn$ since there is one element in the set for each cell that can be occupied by the "1" value. The homomorphism associated to the action is $\psi : C \to S_{4nm}$. Furthermore, we know that C is isomorphic to a subgroup of S_{4nm} by Cayley's Theorem. The rest of this paper is dedicated to examining this action and subgroup.

As just mentioned, there is an intuitive bijection from a multiverse of lonely universes to the elements of S_{4nm}. As a matter of notation, we will refer to S_{4mn} as the group that of permutations of elements $\{0, 1, 2, 3, ..., 4mn-1\}$, not the canonical $\{1, 2, 3, 4..., 4mn\}$. This notation is established here to avoid confusion when we shift our focus from the abstract S_{4mn} group to its representation in the Python programming language. Now we construct a bijection as follows: let $T_{0,0}$ be the lonely universe whose living cell is located at $x_{0,0}$, and so forth. Then

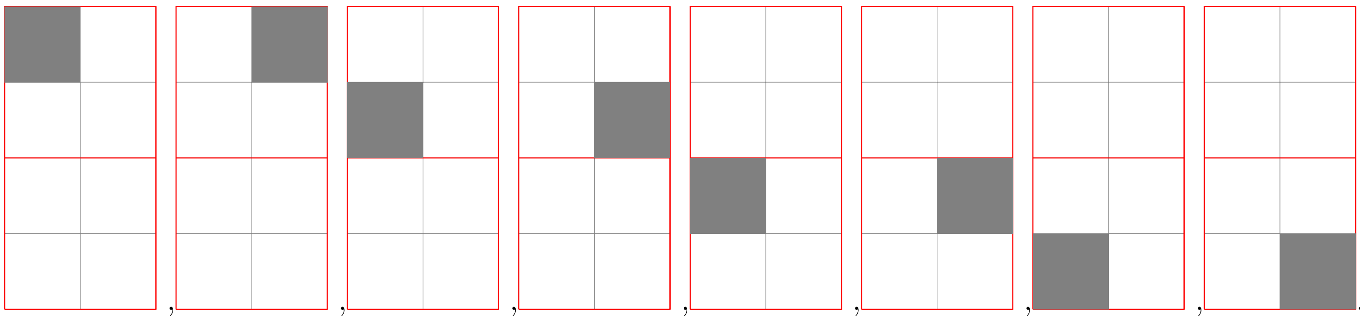
$$\begin{aligned}
\mathfrak{M}(T^L) &\to \{0, 1, 2, 3, ...4mn-1\} \\
T_{0,0} &\mapsto 0 \\
T_{1,0} &\mapsto 1 \\
&\vdots \\
T_{2n-2,2m-1} &\mapsto 4mn-2 \\
T_{2n-2,2m-1} &\mapsto 4mn-1
\end{aligned}$$

is a bijection.

After fixing n and m and orienting and (arbitrarily) partitioning the grid, we can represent $T \in \mathfrak{M}(T^L_{2n\times 2m})$ with a matrix

$$T = \begin{bmatrix}
\begin{bmatrix} x_{0,0} & x_{1,0} \\ x_{0,1} & x_{1,1} \end{bmatrix} & \begin{bmatrix} x_{2,0} & x_{3,0} \\ x_{2,1} & x_{3,1} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,0} & x_{2n-1,0} \\ x_{2n-2,1} & x_{2n-1,1} \end{bmatrix} \\
\begin{bmatrix} x_{0,2} & x_{1,2} \\ x_{0,3} & x_{1,3} \end{bmatrix} & \begin{bmatrix} x_{2,2} & x_{3,2} \\ x_{2,3} & x_{3,3} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,2} & x_{2n-1,2} \\ x_{2n-2,3} & x_{2n-1,3} \end{bmatrix} \\
\vdots & \vdots & \ddots & \vdots \\
\begin{bmatrix} x_{0,2m-2} & x_{1,2m-2} \\ x_{0,2m-1} & x_{1,2m-1} \end{bmatrix} & \begin{bmatrix} x_{2,2m-2} & x_{3,2m-2} \\ x_{2,2m-1} & x_{3,2m-1} \end{bmatrix} & \cdots & \begin{bmatrix} x_{2n-2,2m-2} & x_{2n-1,2m-2} \\ x_{2n-2,2m-1} & x_{2n-1,2m-1} \end{bmatrix}
\end{bmatrix}, x_{i,j} \in \{0, 1\}.$$

Example 4.2. Consider $\mathfrak{M}(T^L_{2\times 4}) =$



These elements map to $0, 1, 2, 3, 4, 5, 6, 7$, respectively.

We have a bijection from a given multiverse into the elements $\{0, 1, 2, ..., 4mn-1\}$ and we have introduced this notion of symmetry in the group action that results from the Margolus rule structure. Let's take the symmetry phenomenon and manifest it in the permutation representation.

4.3 Generators Revisited

We have touched on the idea of generators of the possible rules that act on a given multiverse. Now that we have introduced the permutation representation of the action, we can express these generators with precision. Recall Example 4.1. Consider the permutations of the four elements of the multiverse of lonely universes to be the group S_4. In Example 4.1 we considered the effects of the permutation $(0, 1)$. We saw that on the multiverse $\mathfrak{M}(T^L_{4\times 4})$ this permutation extended to the permutation $(0, 1)(2, 3)(8, 9)(10, 11)$ by symmetry. The latter is what we refer to as one of the generators of the subgroup of S_{16}. The rows of the grid are entered as elements of the list.

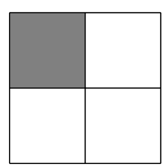
As a visual aid, consider

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

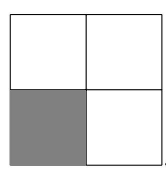
It is natural to think of $0, 1, 2, ..., 4nm - 1$ as cells, but this is not the case. We are trying to capture permutations of the multiverse $\mathfrak{M}(T^L_{2n\times 2m})$. Therefore 0 represents the lonely universe with the living cell in position 0, and so forth, reading left to right and top to bottom.[8]

Let's take another example.

Example 4.3. Take the multiverse of lonely universes of size 2×4, $\mathfrak{M}(T^L_{2\times 4})$. We know this set to have 8 elements. Say we want to observe the extension of the $(0, 2)$ generator. That is, the permutation that swaps



with

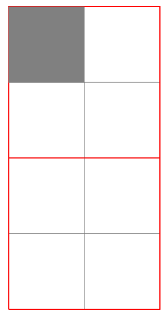
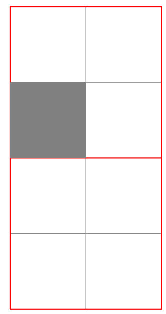


.

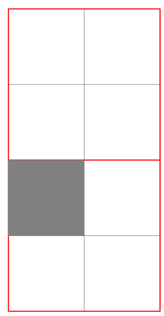
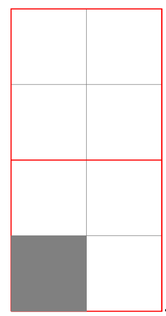
Then we have a composition of two transpositions that represents the generator of this action on the multiverse, namely $(0, 2)(4, 6)$. Visually, this action permutes the

[8]Refer to the bijection described on page 24.

universes

 with 

as well as

 with .

Now that we have a solid theoretical framework for examining the subgroups of S_{4mn}, we turn our attention to the action on specific multiverses.

5 Computing the Subgroups

We examine the action on $\mathfrak{M}(T^L_{2n\times 2m})$ for particular n, m. For each pair (n, m), we have a group action. Each group action in turn has a permutation representation that is a subgroup of S_{4mn} given n, m.

5.1 Subgroup sizes for a given n, m

Perhaps the most basic property of these subgroups that we can compute is their sizes. Using the Python programming language we are able to compute the sizes of some of these subgroups for some small n, m. Program 1 will take a $2n \times 2m$ grid and return the size of the subgroup that is generated by the permutations revealed in the previous section(the extensions of $(0, 1), (0, 2), (0, 3)$ and the shift of the grid). The raw code along with a detailed description of the workings of the code can be found in the appendix.

Program 1 was designed to be true to the structure that we have described here. First, four functions are developed: an identity map and three permutations. Then we define the shift function and its inverse which serve as the partitioning step. Subsequently we define eight new functions that represent time-steps: four functions being a permutation followed by the shift, four functions being a permutation followed by the inverse shift. Recall that we have deconstructed the group structure to the extent that we understand the group action as this four-step process. For this reason, we can claim that this program finds the entire subgroup. Finally we apply these permutations to the identity and add unique permutations to a list. The size of the list gives the size of the subgroup.

This table lists the size grid used as well as, below, the size of the subgroup.

2x2	4x4	6x6	8x8
24	1,536	17,496	98,304

We do face new challenges when we start to consider cases where n is not equal to m. In particular, we need to address two question:

1. Does changing the orientation of the grid change the size of the subgroup?

   For example, is the permutation representation of the group action on the 2×4 grid on a torus the same as that of the action on the 4×2? Since we are working on a torus, the intuitive answer is "yes, the sizes of the subgroups are the same." We can only deal with their sizes while acknowledging that the structures may be different. *For small n, m, it is supported that the subgroups have the same sizes.*

2x4	4x2	2x6	6x2	2x8	8x2	2x10	10x2	2x12	12x2
192	192	648	648	1,536	1,536	3,000	3,000	5,184	5,184

4x6	6x4	4x8	8x4	4x10	10x4	4x12	12x4
5,184	5,184	12,288	12,288	24,000	24,000	41,472	41,472

   This leads us to the second question.

2. Does the shape of the universe matter?

This question builds on question 1. Supposing the orientation does not matter, what if we change n and m while keeping the number of MN the same? For example, how, if at all, does the representation of the action on the 4×4 multiverse differ from that of the action on the 2×8? They share the same number of MN, does that mean that they will yield the same subgroup (up to isomorphism)? *We see that the shape does not appear to be influential.* In the following figure, like-colored cells have the same number of MN (aside from the black cells for which no value was calculated). The number in each cell is the size of the subgroup of the permutation representation of the action on the multiverse of lonely universes of size $2n \times 2m$ for $2n$ the index found in the column and $2m$ the index found in the row.[9]

x	2	4	6	8	10	12
2	24	192	648	1,536	3,000	5,184
4	192	1,536	5,184	12,288	24,000	41,472
6	648	5,184	17,496	41,472	81,000	139,968
8	1,536	12,288	41,472	98,304		
10	3,000	24,000	81,000			
12	5,184	41,472	139,968			

Note how small these subgroups are relative to S_{4mn}. For example, consider the size of the subgroup of S_{36} relative to S_{36} : $\dfrac{17,496}{36!} = \dfrac{2^3 * 3^7}{36!} = \dfrac{2 * 3^5}{35!} \approx 4.07033 * 10^{-38}$.

5.2 Subgroup Structure

Now that we have computed some subgroup sizes, we can consider their structures as well. Due to the large sizes of the subgroups, we are only going to consider the 2×2 and 2×4 cases explicitly. The methods shown here of deconstructing the

[9] Although, at least for this data set, n and m can be exchanged arbitrarily.

structure could be likewise applied to larger n, m. Specifically, we will compute the generators for each subgroup for specific n, m. For simplicity, when $n \neq m$, we will take $m > n$. We will assume that the two are identical up to isomorphism and therefore that we have covered both cases.

Recall that we are taking pairs of bijections. So, in each case, we list the permutations that correspond to the three generating permutations on the Margolus neighborhood, (0,1),(0,2) and (0,3), as well as the permutation that corresponds to the shift of the grid, and it's inverse. From these, we can write the generators of the subgroup as a sequence of compositions: a permutation (possible the identity), a shift, a permutation, an inverse shift. In this way, the permutations and shifts are like generators for our generating set, although, not in the strict mathematical sense.

Lemma 3. *Disjoint cycles commute.[14, page 32]*

- 2×2, subgroup of S_4
  This case is completely known. We have a subgroup of S_4 that is of size 24. We conclude that this particular subgroup is in fact S_4.

  - **Permutations:** (), (0,1), (0,2), (0,3)

  - **Shift**: (0,3)(1,2)

  - **Shift Inverse**: (0,3)(1,2)

  From here we take sequences to find our generators and simplify. In this case, we see that we can use our three permutations as generators.
  Generators (2 $\times$ 2):

  - (0,1)
  - (0,2)
  - (0,3)

  This generates all of S_4.

- **2 × 4, subgroup of S_8**

  From the symmetry we get the permutations (in cycle decomposition form): (0,1)(4,5), (0,2)(4,6), and (0,3)(4,7). The shift of the grid can be thought of as a permutation (0,7,4,3)(1,6,5,2). Equivalently, this permutation could be (0,3,4,7)(1,2,5,6). (Note (0,7,4,3)(1,6,5,2) = $((0,3,4,7)(1,2,5,6))^{-1}$). The size of the subgroup generated by these permutations is given by Program 1 to be 192.

  - **Permutations:** (), (0,1)(4,5), (0,2)(4,6), (0,3)(4,7)
  - **Shift:** (0,7,4,3)(1,6,5,2)
  - **Shift Inverse:** (0,3,4,7)(1,2,5,6)

  We have 16 possible combinations of permutations to serve as generators for this subgroup.

  - (0,3,4,7)(1,2,5,6)∘()∘(0,7,4,3)(1,6,5,2)∘() = ()
  - (0,3,4,7)(1,2,5,6)∘()∘(0,7,4,3)(1,6,5,2)∘(0,1)(4,5) = (0,1)(4,5)
  - (0,3,4,7)(1,2,5,6)∘()∘(0,7,4,3)(1,6,5,2)∘(0,2)(4,6) = (0,2)(4,6)
  - (0,3,4,7)(1,2,5,6)∘()∘(0,7,4,3)(1,6,5,2)∘(0,3)(4,7) = (0,3)(4,7)
  - (0,3,4,7)(1,2,5,6)∘(0,1)(4,5)∘(0,7,4,3)(1,6,5,2)∘() = (2,3)(6,7)
  - (0,3,4,7)(1,2,5,6)∘(0,2)(4,6)∘(0,7,4,3)(1,6,5,2)∘() = (1,7)(3,5)
  - (0,3,4,7)(1,2,5,6)∘(0,3)(4,7)∘(0,7,4,3)(1,6,5,2)∘() = (0,7)(3,4)
  - (0,3,4,7)(1,2,5,6)∘(0,1)(4,5)∘(0,7,4,3)(1,6,5,2)∘(0,1)(4,5) = (0,1)(2,3)(4,5)(6,7)
  - (0,3,4,7)(1,2,5,6)∘(0,2)(4,6)∘(0,7,4,3)(1,6,5,2)∘(0,1)(4,5) = (0,7,1)(3,5,4)
  - (0,3,4,7)(1,2,5,6)∘(0,3)(4,7)∘(0,7,4,3)(1,6,5,2)∘(0,1)(4,5) = (0,1,7)(3,4,5)
  - (0,3,4,7)(1,2,5,6)∘(0,1)(4,5)∘(0,7,4,3)(1,6,5,2)∘(0,2)(4,6) = (0,3,2)(4,7,6)
  - (0,3,4,7)(1,2,5,6)∘(0,2)(4,6)∘(0,7,4,3)(1,6,5,2)∘(0,2)(4,6) = (0,2)(1,7)(3,5)(4,6)
  - (0,3,4,7)(1,2,5,6)∘(0,3)(4,7)∘(0,7,4,3)(1,6,5,2)∘(0,2)(4,6) = (0,2,7)(3,4,6)
  - (0,3,4,7)(1,2,5,6)∘(0,1)(4,5)∘(0,7,4,3)(1,6,5,2)∘(0,3)(4,7) = (0,2,3)(4,6,7)
  - (0,3,4,7)(1,2,5,6)∘(0,2)(4,6)∘(0,7,4,3)(1,6,5,2)∘(0,3)(4,7) = (0,5,3)(1,7,4)

- (0,3,4,7)(1,2,5,6)∘(0,3)(4,7)∘(0,7,4,3)(1,6,5,2)∘(0,3)(4,7) = (0,4)(3,7)

We conclude that $\{(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (2,3)(6,7), (1,7)(3,5), (0,7)(3,4), (0,1)(2,3)(4,5)(6,7), (0,7,1)(3,5,4), (0,1,7)(3,4,5), (0,3,2)(4,7,6), (0,2)(1,7)(3,5)(4,6), (0,2,7)(3,4,6), (0,2,3)(4,6,7), (0,5,3)(1,7,4), (0,4)(3,7)\}$ is a generating set for the subgroup. This is not, however, a minimal generating set. For example, $(0,7,1)(3,5,4) = ((0,1,7)(3,4,5))^{-1}$, therefore this cannot be the minimal generating set.

Theorem 5.1. *The subgroup of S_8 that is the permutation representation of the action on the multiverse of lonely universes of size 2×4 has a minimal generating set* $\{(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (1,7)(3,5)\}$.

Proof. We have shown that the set $\{(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (2,3)(6,7), (1,7)(3,5), (0,7)(3,4), (0,1)(2,3)(4,5)(6,7), (0,7,1)(3,5,4), (0,1,7)(3,4,5), (0,3,2)(4,7,6), (0,2)(1,7)(3,5)(4,6), (0,2,7)(3,4,6), (0,2,3)(4,6,7), (0,5,3)(1,7,4), (0,4)(3,7)\}$ is a generating set for the subgroup. First we show that all of these elements can be written as a composition of $(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (2,3)(6,7)$, and $(1,7)(3,5)$:

- (0,7)(3,4) = (1,7)(3,5) ∘ (0,1)(4,5) ∘ (1,7)(3,5)
- (0,1)(2,3)(4,5)(6,7) = (0,1)(4,5) ∘ (2,3)(6,7)
- (0,7,1)(3,5,4) = (1,7)(3,5) ∘ (0,1)(4,5) ∘ (1,7)(3,5) ∘ (0,1)(4,5)
- (0,1,7)(3,4,5) = (0,1)(4,5) ∘ (1,7)(3,5)
- (0,3,2)(4,7,6) = (2,3)(6,7) ∘ (0,2)(4,6)
- (0,2)(1,7)(3,5)(4,6) = (0,2)(4,6) ∘ (1,7)(3,5)
- (0,2,7)(3,4,6) = (1,7)(3,5) ∘ (0,1)(4,5) ∘ (1,7)(3,5) ∘ (0,2)(4,6)
- (0,2,3)(4,6,7) = (0,3)(4,7) ∘ (0,2)(4,6)
- (0,5,3)(1,7,4) = (1,7)(3,5)∘ (0,1)(4,5) ∘ (1,7)(3,5) ∘ (0,3)(4,7)
- (0,4)(3,7) = (1,7)(3,5)∘ (0,1)(4,5) ∘ (1,7)(3,5) ∘ (0,3)(4,7)

So $\{(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (2,3)(6,7), (1,7)(3,5)\}$ is a generating set for $\{(0,1)(4,5), (0,2)(4,6), (0,3)(4,7), (2,3)(6,7), (1,7)(3,5), (0,7)(3,4), (0,1)(2,3)(4,5)(6,7), (0,7,1)(3,5,4), (0,1,7)(3,4,5), (0,3,2)(4,7,6), (0,2)(1,7)(3,5)(4,6),$

$(0,2,7)(3,4,6),(0,2,3)(4,6,7),(0,5,3)(1,7,4),(0,4)(3,7)\}$. Program 2, available in the appendix, was used to compute the sizes of subgroups generated by the remaining five permutations. When we consider only

$$\{(0,1)(4,5),(0,2)(4,6),(0,3)(4,7),(1,7)(3,5)\},$$

we still generate a subgroup of 192 elements. It follows that we can write $(2,3)(6,7)$ as a composition of these four elements. Indeed, we find that

$$(2,3)(6,7) = (0,3)(4,7) \circ (0,2)(4,6) \circ (0,3)(4,7).$$

Therefore,

$$\{(0,1)(4,5),(0,2)(4,6),(0,3)(4,7),(1,7)(3,5)\}$$

is a generating set for the subgroup. When we attempt to remove any other element from this set, we only generate a subgroup of 24 elements. We conclude that this is the minimal generating set for the subgroup.

□

We will not explicitly write the generating set for larger size universes, but we will write the permutations from which we create the generating set.

- **4 × 4**
  - **Permutations:** (0,1)(2,3)(8,9)(10,11), (0,4)(2,6)(8,12)(10,14), (0,5)(2,7)(8,13)(10,15), (0,5,10,15)(1,6,11,12)
  - **Shift:**(0,5,10,15)(1,6,11,12)(2,7,8,13)(3,4,9,14)
  - **Shift Inverse:** (0,15,10,5)(1,12,11,6)(2,13,8,7)(3,14,9,4)
- **4 × 6**
  - **Permutations:**
  - (0,1)(2,3)(8,9)(10,11)(16,17)(18,19),
    (0,4)(2,6)(8,12)(10,14)(16,20)(18,22),
    (0,5)(2,7)(8,13)(10,15)(16,21)(18,23)
  - **Shift:** (0,5,10,15,16,21,2,7,8,13,18,23)(1,6,11,12,17,22,3,4,9,14,19,20)
  - **Shift Inverse:** (0,23,28,13,8,7,2,21,16,15,10,5)(1,20,19,14,9,4,3,22,17,12,11,6)
- **6 × 6**

- **Permutations:**
- (0,1)(2,3)(4,5)(12,13)(14,15)(16,17)(24,25)(26,27)(28,29),
  (0,6)(2,8)(4,10)(12,18)(14,20)(16,22)(24,30)(26,32)(28,34),
  (0,7)(2,9)(4,11)(12,19)(14,21)(16,23)(24,31)(26,33)(28,35)
- **Shift:** (0,7,14,21,28,35)(1,8,15,22,29,30)(2,9,16,23,24,31)(3,10,17,18,25,32)
  (4,11,12,19,26,33)(5,6,13,20,27,34)
- **Shift Inverse:** (0,35,28,21,14,7)(1,30,29,22,15,8)(2,31,24,23,16,9)(3,32,25,18,17,10)
  (4,33,26,19,12,11)(5,34,27,20,13,6)

What are the trends here? First of all, the size of the shift permutation will always be $4nm$ (recall that the grids are of size $2n \times 2m$, $n, m \in \mathbb{N}$). The number of generators will always be 3, not including the shift. The generators are composed of transpositions. When $n = m$, the shift permutation is composed of $2n$ cycles of length $2n$ (note also that this is true for the 2×2 case although this was not discussed). The number of cycles in a generator is $n * m$.

6 Future Investigation

- Is symmetry the only restriction on the permutations?
- Why don't the relative dimensions of the grid impact the size of the subgroup?
- What happens when we move to multiverses of two living cells? Three? ...

A Program 1: Code for Finding Subgroup Size with Explanation

```
Program 1
from copy import deepcopy
import itertools
#First we choose "data"; each sub-array represents a row (or column, arbitrarily).
#data = [[0,1,2,3],[4,5,6,7],[8,9,10,11],[12,13,14,15],[16,17,18,19],[20,21,22,23]]
#data = [[0,1],[2,3],[4,5],[6,7]]
```

```
#data = [[0,1,2,3,4,5],[6,7,8,9,10,11],[12,13,14,15,16,17],[18,19,20,21,22,23],
#[24,25,26,27,28,29],[30,31,32,33,34,35]]
data = [[0,1,2,3,4,5,6,7,8,9],[10,11,12,13,14,15,16,17,18,19],
[20,21,22,23,24,25,26,27,28,29],[30,31,32,33,34,35,36,37,38,39],
[40,41,42,43,44,45,46,47,48,49],[50,51,52,53,54,55,56,57,58,59]]
#data = [[0,1],[2,3]]
#data = [[0,1,2,3],[4,5,6,7]]

#"elle" will record all of the elements in the subgroup.
elle = list()
elle.append(data)

# We are going to construct first the three generating permutations. We only need
# these three since we can apply a
# permutation followed by a shift and the inverse shift at which point we can apply
# any of these permutations once
# again. In this way, we can apply any of the three permutations consecutively.
# Since transpositions generate the group
# of all permutations on the set, this is all we need.



def perm01(lst):
    #this is the first generator, permuting 0 and 1
    out = deepcopy(lst)
    x_size = len(lst[0])
    y_size = len(lst)


    for i in range(x_size/2):
        for j in range(y_size/2):

            out[(2*j)%y_size][(2*i)%x_size], out[(2*j)%y_size][(2*i + 1)%x_size]
            = out[(2*j)%y_size][(2*i + 1)%x_size], out[(2*j)%y_size][(2*i)%x_size]

    return out
def perm02(lst):
    #second generator permuting 0 and 2
```

```
    out = deepcopy(lst)
    x_size = len(lst[0])
    y_size = len(lst)


    for i in range(x_size/2):
        for j in range(y_size/2):
            out[(2*j)%y_size][(2*i)%x_size], out[(2*j+1)%y_size][(2*i + 1)%x_size]
            = out[(2*j+1)%y_size][(2*i + 1)%x_size], out[(2*j)%y_size][(2*i)%x_size]
    return out


def perm03(lst):
    #third generator, permuting 0 and 3
    out = deepcopy(lst)
    x_size = len(lst[0])
    y_size = len(lst)


    for i in range(x_size/2):
        for j in range(y_size/2):
            out[(2*j)%y_size][(2*i)%x_size], out[(2*j+1)%y_size][(2*i)%x_size]
            = out[(2*j+1)%y_size][(2*i)%x_size], out[(2*j)%y_size][(2*i)%x_size]
    return out


def ID(lst):
    #This is the identity. We include it in order to make the program more true
    #to the process described in the paper.
    return lst

def shift(lst):
    #This is the shift map which captures the initial re-partitioning of the grid.


    out = deepcopy(lst)

    x_size = len(lst[0])
```

```
    y_size = len(lst)
    for i in range(y_size):
        for j in range(x_size):
            out[(i-1)%(y_size)][(j-1)%(x_size)] = deepcopy(lst[i][j])



    return out

def shiftINV(lst):


    out = deepcopy(lst)

    x_size = len(lst[0])
    y_size = len(lst)
    for i in range(y_size):
        for j in range(x_size):
           out[(i+1)%(y_size)][(j+1)%(x_size)] = deepcopy(lst[i][j])

    return out

def IDshift(lst):
    #This is a time-step: Identity map followed by shift
    perm = ID(lst)
    out = shift(perm)
    return out

def perm01shift(lst):
    #Time-step: Permutation (0,1) then shift
    perm = perm01(lst)
    out = shift(perm)
    return out



def perm02shift(lst):
    #Time-step: Permutation (0,2) then shift
```

```
    perm = (perm02(lst))
    out = shift(perm)
    return out



def perm03shift(lst):
    #Time-step: Permutation (0,3) then shift
    perm = (perm03(lst))
    out = shift(perm)
    return out


def IDshiftINV(lst):
    #Subsequent time-step: Identity then inverse shift
    perm = (ID(lst))
    out = shiftINV(perm)
    return out

def perm01shiftINV(lst):
    #Subsequent time-step: Permutaion (0,1) then inverse shift
    perm = (perm01(lst))
    out = shiftINV(perm)
    return out

def perm02shiftINV(lst):
    #Subsequent time-step: Permutation (0,2) then inverse shift
    perm = (perm02(lst))
    out = shiftINV(perm)
    return out

def perm03shiftINV(lst):
    #Subsequent time-step: Permutation (0,3) then inverse shift
    perm = (perm03(lst))
    out = shiftINV(perm)
    return out
```

```
#We are going to have a list to store which permutations are "new".
#This way we can act on only "new" permutations using our functions.
IWASSAVINGTHOSE = [data]

#This "while" loop terminates when we get nothing new out of acting
#on permutations with our functions. This means that we have found everything.
while len(IWASSAVINGTHOSE) != 0:

     copy = deepcopy(IWASSAVINGTHOSE)
     #for k in range(len(IWASSAVINGTHOSE)):
     ## This step is broken into repetitive steps. We write a first
     ## time-step, followed by all the possible subsequent
     ## time-steps.
     for k in range(len(copy)):
            IDEN = IDshift(copy[k])
            #We have the identity map followed by a shift.

            IDEN01 = perm01shiftINV(IDEN)
            if IDEN01 not in elle:
                IWASSAVINGTHOSE.append(IDEN01)
                elle.append(IDEN01)
            # We take the first time-step from above (IDEN) and we
            # follow it with another time-step, and record the
            # output if it is not already recorded.

            IDEN02 = perm02shiftINV(IDEN)
            if IDEN02 not in elle:
                IWASSAVINGTHOSE.append(IDEN02)
                elle.append(IDEN02)
            #Another possible time-step following IDEN

            IDEN03 = perm03shiftINV(IDEN)
            if IDEN03 not in elle:
                IWASSAVINGTHOSE.append(IDEN03)
                elle.append(IDEN03)
```

```
            #Another possible time-step following IDEN

# We don't need to consider IDshiftINV since we would return to the identity,
# which is already recorded.

            permA = perm01shift(copy[k])
            #We move on to the next ``starting" time -step and all
            #the possible subsequent time-steps follow.

            permB = perm01shiftINV(permA)
            if permB not in elle:
                IWASSAVINGTHOSE.append(permB)
                elle.append(permB)

            permC = perm02shiftINV(permA)
            if permC not in elle:
                IWASSAVINGTHOSE.append(permC)
                elle.append(permC)

            permD = perm03shiftINV(permA)
            if permD not in elle:
                IWASSAVINGTHOSE.append(permD)
                elle.append(permD)

            permE = IDshiftINV(permA)
            if permE not in elle:
                IWASSAVINGTHOSE.append(permE)
                elle.append(permE)

            BpermA = perm02shift(copy[k])

            BpermB = perm01shiftINV(BpermA)
            if BpermB not in elle:
                IWASSAVINGTHOSE.append(BpermB)
                elle.append(BpermB)

            BpermC = perm02shiftINV(BpermA)
            if BpermC not in elle:
```

```
            IWASSAVINGTHOSE.append(BpermC)
            elle.append(BpermC)

        BpermD = perm03shiftINV(BpermA)
        if BpermD not in elle:
            IWASSAVINGTHOSE.append(BpermD)
            elle.append(BpermD)

        BpermE = IDshiftINV(BpermA)
        if BpermE not in elle:
            IWASSAVINGTHOSE.append(BpermE)
            elle.append(BpermE)


        CpermA = perm03shift(copy[k])

        CpermB = perm01shiftINV(CpermA)
        if CpermB not in elle:
            IWASSAVINGTHOSE.append(CpermB)
            elle.append(CpermB)

        CpermC = perm02shiftINV(CpermA)
        if CpermC not in elle:
            IWASSAVINGTHOSE.append(CpermC)
            elle.append(CpermC)

        CpermD = perm03shiftINV(CpermA)
        if CpermD not in elle:
            IWASSAVINGTHOSE.append(CpermD)
            elle.append(CpermD)

        CpermE = IDshiftINV(CpermA)
        if CpermE not in elle:
            IWASSAVINGTHOSE.append(CpermE)
            elle.append(CpermE)

#Now that we have applied every function to this permutation,
#we can remove it from the list of "new" elements.
```

```
            IWASSAVINGTHOSE.remove(copy[k])


print len(elle)

# As an additional measure we make sure that there aren't multiple copies
# saved in the list.
k = 0
for i in range(len(elle)):
    for j in range(len(elle)):
        if elle[i] == elle[j]:
            k+=1

print k
#k should be the same as "final".

#IWASAVINGTHOSE should be empty. That is, there should be no
#elements that we didn't apply all of our functions to.
print IWASSAVINGTHOSE

#Finally we remove any copies in the code.
elle.sort()
final = list(k for k,_ in itertools.groupby(elle))
print len(final)
```

B Program 2: Code for Finding Minimal Generating Set

```
Program 2
from copy import deepcopy
import itertools

#this represents our identity element of S_8
data = [[0,1,2,3,4,5,6,7]]

#"elle" will record all of the unique permutations that we find.
elle = []
elle.append(data)

#Now we define 5 functions that represent the five elements of the generating set.
#This function is (0,1)(4,5).
def zeroone(lst):
    out = deepcopy(lst)
    out[0], out[1] = out[1], out[0]
    out[4], out[5]  = out[5], out[4]
    return out

#This function is (0,2)(4,6).
def zerotwo(lst):
    out = deepcopy(lst)
    out[0], out[2] = out[2], out[0]
    out[4], out[6]  = out[6], out[4]
    return out

#This function is (0,3)(4,7).
def zerothree(lst):
    out = deepcopy(lst)
    out[0], out[3] = out[3], out[0]
    out[4], out[7]  = out[7], out[4]
    return out
```

```
#This function is (2,3)(6,7).
def twothree(lst):
    out = deepcopy(lst)
    out[2], out[3] = out[3], out[2]
    out[6], out[7]  = out[7], out[6]
    return out

#This function is (1,7)(3,5).
def oneseven(lst):
    out = deepcopy(lst)
    out[7], out[1] = out[1], out[7]
    out[3], out[5]  = out[5], out[3]
    return out

#"NEW" will record data that is "new" in the sense that
#it has yet to be run through all of the functions.
NEW = (data)

#We want to stop when we aren't getting new output.
while len(NEW) != 0:
    copy = deepcopy(NEW)
    for k in range(len(copy)):

#In each block, we apply a function to something that is "new" and
#then, if we have not already seen the output, we store it both as
#a "new" element and store it in "elle".
        perm1 = zeroone(copy[k])
        if perm1 not in elle:
            elle.append(perm1)
            NEW.append(perm1)

        perm2 = zerotwo(copy[k])
        if perm2 not in elle:
            elle.append(perm2)
            NEW.append(perm2)

        perm3 = zerothree(copy[k])
```

```
        if perm3 not in elle:
           elle.append(perm3)
           NEW.append(perm3)

#This is the function that we remove.
        perm4 = twothree(copy[k])
        if perm4 not in elle:
            elle.append(perm4)
            NEW.append(perm4)

        perm5 = oneseven(copy[k])
        if perm5 not in elle:
            elle.append(perm5)
            NEW.append(perm5)

        NEW.remove(copy[k])

print len(elle)


k = 0
for i in range(len(elle)):
    for j in range(len(elle)):
        if elle[i] == elle[j]:
            k+=1

print k

#We make sure that "NEW" is empty.
print NEW

#Now we check that there are no copies in the set.
elle.sort()
final = list(k for k,_ in itertools.groupby(elle))
print len(final)

for k in range(len(elle)):
    print elle[k]
```

```
elle.remove([])
print len(elle)
```

References

[1] M. Gardner. "Mathematical Games: The fantastic combinations of John Conway's new solitaire game 'Life'". *Scientific American 223 (October 1970): 120-123.*

[2] J. von Neumann and A.W. Burks (1966). *Theory of self-reproducing automata.* Urbana, University of Illinois Press.

[3] T. Toffoli and N. Margolus (1987). "The Margolus neighborhood". *Cellular Automata Machines: A New Environment for Modeling (II.12)*, MIT Press, pp. 119-138.

[4] D. Richardson. "Tessellation with Local Transformations," *J. Comp. Syst. Sci. 6 (1972), 373-388.*

[5] A. Moreira, N. Boccara, & E. Goles. "On conservative and monotone one-dimensional cellular automata and their particle representation". *Theoretical Computer Science - Special issue: Theoretical aspects of cellular automata Vol. 325 Issue 2* (1 October, 2004): 285-316. Web. Accessed 9 July, 2016.

[6] T. Toffoli and N. Margolus. "Invertible cellular automata: A review". *Physica D 45 (1990). (pg 229-253).*

[7] H. Segerman. "Re: First draft." Message to Austin Warner. 10 July, 2016. Email.

[8] J. Kari. "Reversible Cellular Automata". *Developments in Language Theory Vol. 3572* (2005): 57-68. Web. Accessed 9 July, 2016.

[9] J. Durand-Lose, "Representing Reversible Cellular Automata with Reversible Block Cellular Automata," *Discrete Mathematics and Theoretical Computer Science Proceedings AA (DM-CCG)*, 2001, 145–154. Web. Accessed 31 July, 2016.

[10] C. Hand, "Simple Cellular Automata on a Spreadsheet". *Computers in Higher Education Economics Review Vol. 17* (2005). Web. Accessed 10 July, 2016.

[11] Unknown. "Block cellular automaton". *Wikipedia.* Web. Accessed 10, July 2016.

[12] Unknown. "Critters (block cellular automaton)". *Wikipedia.* Web. Accessed 10, July 2016.

[13] D. Shiffman. "Figure 7.14: Rule 90". *The Nature of Code*. Web. Accessed 10, July 2016.

[14] D. S. Dummit and R. M. Foote. *Abstract Algebra* 3rd Ed. John Wiley & Sons. Print.