UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Matematika – 1. stopnja

Program dvojne diplome iz matematike
z Univerzo v Trstu

Carlo Lanzi Luciani

Automatne Grupe

Delo diplomskega seminarja

Mentorja: izr. prof. dr. Ganna Kudryavtseva
prof. Alessandro Logar

Ljubljana, 2020

UNIVERSITÀ DEGLI STUDI DI TRIESTE
DIPARTIMENTO DI MATEMATICA
E GEOSCIENZE

UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Programma di doppio titolo
in Matematica

Program dvojne diplome
iz matematike

Double Degree Program in Mathematics

Carlo Lanzi Luciani

Gruppi di Automi

Tesi finale

Automatne Grupe

Delo diplomskega seminarja

Groups of Automata

Final Thesis

Supervisori/Mentorja/Supervisors
izr. prof. dr. Ganna Kudryavtseva
prof. Alessandro Logar

2020

Contents

Groups of Automata

Abstract

In this Bachelor Thesis we present some interesting examples and results on groups generated by Automata. First we treat the mathematical idea of input and output and we give the definition of Finite Synchronous Deterministic Mealy Automaton. We exploit it to generate groups and we link all these concepts with some abstract structures, as the tree homomorphism or the wreath product. We include some interesting examples, such as the Infinite Dihedral Group and the Lamplighter Group. Finally we present the Classification of all groups generated by 2-state-Automata over a 2-letter-alphabet.

Math. Subj. Class. (2010): 68Q45, 68Q70, 20E07, 20E22, 20E08, 18B20, 20M05

Keywords: Automata, Finite Automata, Word Spaces, Moore Diagram, Wreath Product, Semidirect Product, Recursion, Classification of Groups

Automatne Grupe

Razširjeni povzetek

Kasual

Ključne besede: Automata, Končni Automata, Besedni Prostori, Moorejevi Diagrami, Semidirektni Produkti, Krožni Produkti, Rekursivnost, Klasifikacija Grup

Gruppi di Automi

Sintesi Estesa

Casuale

Parole chiave: Automi, Automi Finiti, Spazi di Parole, Diagrammi di Moore, Prodotti Circolari, Prodotti Semidiretti, Ricorsività, Classificazione di Gruppi

Part 1. Introduction

The word *automaton* comes from greek (plural *automat**A*** or *automatons*), and means "acting on one's self-will". Roughly speaking an Automaton is a very specific *model of computation*. We can heuristically say that a model of computation is a machine which, taking an input, gives an output(Figure 1).

FIGURE 1. Model of Computation

So to first understand the topic we must define and understand the structure of input and output.

1. Words Spaces and Alphabet Trees

Data given and received will be somehow written and read, through some kind of symbols. Mathematically:

Definition 1.1. An **Alphabet X** is a finite set of elements called **letters**.

Definition 1.2. The set $\mathbf{X}^* := \{x_1 \dots x_n | n \in \mathbb{N}, x_i \in \mathbf{X}\}$ is called the **Set of Finite Words** or **Finite Dictionary**, and its elements are called **words**. The element with no letters, written as $\varnothing$, is called the *empty word*.

Definition 1.3. Let $\mathbf{w} = x_1 \dots x_n$ and $\mathbf{u} = y_1 \dots y_m$ be words. The **lenght** of $\mathbf{w}$, written as $|\mathbf{w}|$, is n. The lenght of the empty word is 0. The **concatenation** of $\mathbf{w}$ and $\mathbf{u}$, written as $\mathbf{w}.\mathbf{u} = \mathbf{wu}$ is the word $x_1 \dots x_n y_1 \dots y_m$.

Example 1.4. Let $\mathbf{X}$ $\{0, 1\}$ then $0100.111 = 0100111$ and $11.0101 = 110101$. Let $\mathbf{X}$ be $\{0, j, 2\}$. Then $02j.20j = 02j20j$ and $j.2j = j2j$. ♢

Proposition 1.5. $(\mathbf{X}^*, \circ)$ *is a monoid, called the* ***Free Monoid on*** X

Proof. The operation $\circ$ is associative with $\varnothing$ beign an identity element. □

The reader may now wonder: can a word with an infinite lenght exist?

Definition 1.6. The **Set of Infinite Words** or the **Infinite Dictionary** is the set $\mathbf{X}^\omega := \{x_1 \dots x_i \dots | x_i \in \mathbf{X}\} = \mathbf{X}^{\mathbb{N}}$.

Remark 1. Note that if $\mathbf{u} = x_1 \dots x_n \in \mathbf{X}^*$ and $\mathbf{v} = y_1 \dots y_i \dots \in \mathbf{X}^\omega$, we can define $\mathbf{u}.\mathbf{v} := x_1 \dots x_n y_1 \dots y_i \dots \in \mathbf{X}^\omega$.

Definition 1.7. A word $\mathbf{w} = x_1 \dots x_n$ is the **beginning** or **prefix** of a word $\mathbf{u} \in \mathbf{X}^*$(or $\mathbf{u} \in \mathbf{X}^\omega$) if $\mathbf{u} = \mathbf{wv} = x_1 \dots x_n \mathbf{v}$ for some $\mathbf{u} \in \mathbf{X}^*$(or $\in \mathbf{X}^\omega$). In this case we set $\mathbf{v} = \mathbf{u} - \mathbf{w}$

Given $A \in \mathbf{X}^* \cup \mathbf{X}^\omega$, we denote $\mathcal{P}(A)$ the **longest common prefix of** A, that is uniquely defined.

1.1. **Topology on the Infinite Dictionaries.** We can provide the set $\mathbf{X}^\omega$ with a metric, and consequently a topology.

Let $\widetilde{\lambda} = (\lambda_n)_{n\in\mathbb{N}}$ be an arbitrary *decreasing* sequence of *positive* numbers such that $lim_{n\to\infty}\lambda_n = 0$. So we can define

$$d_{\widetilde{\lambda}}(\mathbf{w_1}, \mathbf{w_2}) = \lambda_n \tag{1}$$

on $\mathbf{X}^\omega$, where $n = |\mathcal{P}(\{\mathbf{w_1}, \mathbf{w_2}\})|$ is the lenght of the longest common prefix of the words $\mathbf{w_1}$ and $\mathbf{w_2}$. It can be shown that $d_{\widetilde{\lambda}}$ is a *metric*.

Proof. We proove just the triangular inequality Let $n = |\mathcal{P}(\{\mathbf{w_1}, \mathbf{w_2}\})|$. Let $\mathbf{w_3} \in \mathbf{X}^\omega$. We want to show that

$$d_{\widetilde{\lambda}}(\mathbf{w_1}, \mathbf{w_2}) \leq d_{\widetilde{\lambda}}(\mathbf{w_1}, \mathbf{w_3}) + d_{\widetilde{\lambda}}(\mathbf{w_3}, \mathbf{w_2})$$

Let's help ourselves with $p := |\mathcal{P}(\{\mathbf{w_1}, \mathbf{w_3}\})|$ and $q := |\mathcal{P}(\{\mathbf{w_3}, \mathbf{w_2}\})|$. Our problem becomes to show that $\lambda_n \leq \lambda_p + \lambda_q$. Let's suppose that $p = \mathbf{min}\{p, q\}$(if $q = \mathbf{min}\{p, q\}$ the proof is symmetrical). If $\mathbf{min}\{p, q\} = p \leq n$, since $\widetilde{\lambda}$ is decreasing, we obtain $\lambda_n \leq \lambda_p \leq \lambda_p + \lambda_q$. Let's prove that $p \leq n$ through reductio ad absurdum. Let $p > n$ then:

$x_1 \dots x_n =: \mathcal{P}(\{\mathbf{w_1}, \mathbf{w_2}\})$

$x_1 \dots x_n y_{n+1} \dots y_p =: \mathcal{P}(\{\mathbf{w_1}, \mathbf{w_3}\})$

$x_1 \dots x_n z_{n+1} \dots z_p \dots z_q =: \mathcal{P}(\{\mathbf{w_3}, \mathbf{w_2}\})$

But $x_1 \dots x_n y_{n+1} \dots y_p = x_1 \dots x_n z_{n+1} \dots z_p$ because they are of the same lenght and they are both prefixes of $\mathbf{w_3}$. Consequently the last word, of lenght p, is prefix both of $\mathbf{w_1}$ and $\mathbf{w_2}$, therefore is prefix of $x_1 \dots x_n$, so $p \leq n$, against the hypothesis $p > n$. □

This tell us that every set $\mathbf{wX}^\omega := \{\mathbf{wu}|\mathbf{u} \in \mathbf{X}^\omega\}$ can be seen as a ball of radius $\lambda_{|\mathbf{w}|}$ with the center in an arbitrary point $\mathbf{u} \in \mathbf{wX}^\omega$. In particular it can be shown that $\mathbf{wX}^\omega$ is *clopen*(open and closed).

Remark 2. It is often useful to set $\widetilde{\lambda} = (\frac{1}{n})_{n\in\mathbb{N}}$.

1.2. **Tree Structure of the Dictionaries.** It is very useful to represent $\mathbf{X}^*$ in the form of a tree graph: the verteces will be the elements of $\mathbf{X}^*$ with $\varnothing$ as the root. We then say that $\mathbf{v}$ is a child of $\mathbf{u}$ *iff* $\mathbf{u} = \mathbf{v}x$ for some $x \in \mathbf{X}$. An example is Figure 2.

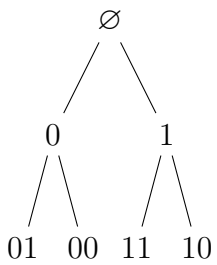


FIGURE 2. An example of a tree graph in the case $X = \{0, 1\}$

Convention 1. *We will mostly use the alphabet* $\mathbf{X} = \{0, 1\}$*; this will also help the reader to better visualise the concept. The case with* $|X| = n > 2$ *is similar.*

The set X^n, of which the elemts are the words of lenght n, is called the *the n-th floor of* X^*(see Figure 2).

Notice that $\mathbf{X}^\omega$ is the very bottom of the graph of Figure 2, therefore that's also called the *Boundary tree*.

Finally we define the notion of endomorphisms of a tree, and some of their properties.

Definition 1.8. Given A, B trees, $f : A \longrightarrow B$ is a **tree-homomorphism** if it preserves the root and the adjacency of the vertices, i. e.:

- If $a \in A$ is the root, $f(a)$ is the root
- If (u, v) is an edge of A, also $(f(u), f(v))$ is an edge of B(that is *f is a graph-homomorphism*)

If $A = B$, f is called *tree-endomorphism*. If $A = B$ and f is bijective we call it a *tree-automorphism*.

It is to verify that all the tree-homomorphisms form a *semigroup* under the composition of functions, and all the tree-automorphisms form its subsemigroup which is also a *group*.

Convention 2. *Very often we will simply write "-morphism" instead of "tree -morphism". The meaning will be clear from the context.*

2. Automata and Initial Automata

Now we will treat the formal definition of the very specific type of Automaton which we need, the *Deterministic Finite(Finite State) Synchronous Automaton*, or *Finite Mealy Automaton*, or *Finite Trasducer*. We will always call it simply *Automaton*, but the reader should know that this is a *very specific* case.

Definition 2.1. An **Automaton** is a 4-tuple $\mathcal{A} =< \mathbf{X}, \mathcal{Q}, \pi, \lambda >$ where:

- $\mathbf{X}$ is an alphabet, usually referred to as the **Input** and/or **Output Alphabet**,
- $\mathcal{Q}$ is a set called the **Set of Internal States of the Automaton**,
- $\pi : \mathbf{X} \times \mathcal{Q} \longrightarrow \mathcal{Q}$ is a function called the **Transition Function**,
- $\lambda : \mathbf{X} \times \mathcal{Q} \longrightarrow \mathbf{X}$ is a function called the **Output Function**.

We say that $\mathcal{A}$ is a $|\mathcal{Q}|$-*state*-automaton *on* $\mathbf{X}$.

This technical description explains us how an automaton performs the action of transforming an input in an output. We can imagine that for every *input letter* x we plug in the machine, and for every *state* q from which we decide to start, the machine moves to a state $p = \pi(x, q) \in \mathcal{Q}$ and returns an *output letter* $y = \lambda(x, q) \in \mathbf{X}$. Don't worry, there is also a way to visualise this.

Definition 2.2. Given an Automaton $\mathcal{A} =< \mathbf{X}, \mathcal{Q}, \pi, \lambda >$ we define its **Moore Diagram** as the oriented graph $G = (\mathcal{Q}, \mathcal{E})$ where 2 states q_1 and q_2 are connected whenever $\exists x \in \mathbf{X} \quad s.t. \quad \pi(x, q_1) = q_2$ and the label reffered to each edge is $x|\lambda(x, q_1)$. Practically: *input|output(input)*.

I thank Edward F. Moore, for he gave us an easy-manageable representation of these objects. An example is Figure 3.

We observe that for every automaton its Moore Diagram has the following property:

(2) $\forall q \in \mathcal{Q}$ and $\forall x \in \mathbf{X} \quad \exists! e \in \mathcal{E}$ such that the left side of its label reads "x"

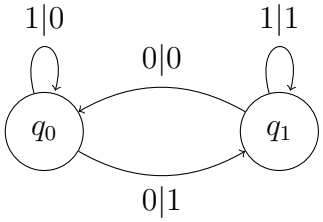


FIGURE 3. Example of a Moore Diagram of a 2-state-Automaton over the alphabet $\mathbf{X} = \{0, 1\}$

That is for every vertex in the graph and for every letter of the input alphabet, exists a unique arrow (from that vertex) through which we travel with that input. We can see many examples in Figure 3.

Remark 3. Moore Diagrams are so important not just for the visual insight, but also because whenever a graph respects the condition (Equation 2), it *uniquely* defines an automaton. So given $\mathcal{M} := \{G \mid G \text{ is a Moore Diagram}\}$, there is a unique correspondence between $\mathcal{M}$ and the set of all Automata. This means that instead of defining this types of machine through some tedious list of outputs(λ) and transitions(π) for each possible input $x \in \mathbf{X}$ and $q \in \mathcal{Q}$, we can simply draw them. Then to compute the outcome of (x, q), we just follow the arrow to $\pi(x, q)$ and look at the left side $\lambda(x, q)$ of the label.

Example 2.3. In Figure 2, given input $\mathbf{w} = 0$ and state $q = q_0$, then $\pi(\mathbf{w}, q) = q_1$ and $\lambda(\mathbf{w}, q) = 1$. If $\mathbf{w} = 1$ and $q = q_1$, then $\pi(\mathbf{w}, q) = q_1$ and $\lambda(\mathbf{w}, q) = 1$. ♢

Proposition 2.4. *We can recursively extend the Domain of π and λ from single letters in $\mathbf{X}$ to words in $\mathbf{X}^*$:*

- $\bar{\pi} : \mathbf{X}^* \times \mathcal{Q} :\longrightarrow \mathcal{Q}$

$$\bar{\pi}(\varnothing, q) = q$$
$$\bar{\pi}(x\mathbf{w}, q) = \bar{\pi}(\mathbf{w}, \bar{\pi}(x, q))$$

- $\bar{\lambda} : \mathbf{X}^* \times \mathcal{Q} :\longrightarrow \mathbf{X}^*$

$$\bar{\lambda}(\varnothing, q) = \varnothing$$
$$\bar{\lambda}(x\mathbf{w}, q) = \bar{\lambda}(x, q)\bar{\lambda}(\mathbf{w}, \bar{\pi}(x, q))$$

Remark 4. The two definitions (2.4) are equivalent to:

$$\bar{\pi}(\mathbf{w}x, q) = \bar{\pi}(x, \bar{\pi}(\mathbf{w}, q))$$

,

$$\bar{\lambda}(\mathbf{w}x, q) = \bar{\lambda}(\mathbf{w}, q)\bar{\lambda}(x, \bar{\pi}(\mathbf{w}, q))$$

Example 2.5. We can compute $\bar{\pi}$ and $\bar{\lambda}$ following the arrows as before, and then making the composition of the single right side of the labels. In the Figure 3, given input $\mathbf{w} = 0000$ and state $q = q_0$, then $\bar{\pi}(q, \mathbf{w}) = q_0$ and $\bar{\lambda}(q, \mathbf{w}) = 1010$. If $\mathbf{w} = 110$ and $q = q_1$, then $\bar{\pi}(q, \mathbf{w}) = q_0$ and $\bar{\lambda}(q, \mathbf{w}) = 110$. ♢

We need to point out something very important: to effectively make an Automaton a word-Transducer we need to specify an Initial State. In other words in Figure 3 to get an output we need to feed the machine both with an input x and a state q. So what happens if we fix $q \in \mathcal{Q}$, and we imagine to always take input starting from this state?

Definition 2.6. If an Automaton $\mathcal{A}$ has a fixed state q_0 we call it an **Initial Automaton with Initial State** q_0 and we write it as $\mathcal{A}_{q_0}$. Each $\mathcal{A}_{q_0}$ naturally defines $\bar{\lambda}_{q_0} : \mathbf{X}^* \longrightarrow \mathbf{X}^*$, with $\bar{\lambda}_{q_0}(\mathbf{w}) := \bar{\lambda}(\mathbf{w}, q_0)$, called the **Action of the Automaton** $\mathcal{A}_{q_0}$. Two *Initial Automata* are said to be *equivalent* if they define the same actions.

Proposition 2.7. *The Action $\bar{\lambda}_{q_0}$ of an Initial Automaton preserves the lenght of words, i.e. $|\bar{\lambda}_{q_0}(\mathbf{w})| = |\mathbf{w}|$.*

Proof. The statement can be easily verified for induction on $n = |\mathbf{w}|$. □

Remark 5. Given $\mathcal{A}_{q_0}$, we can define an *Infinite Action* $\bar{\lambda}_{q_0} : \mathbf{X}^\omega \longrightarrow \mathbf{X}^\omega$ by similar recursive formulas, and we can consequently declare that two Initial Automata are ω-*equivalent* if they determine the same Infinite Action. Two Automata are equivalent *iff* are ω-equivalent. This comes naturally from the fact that both the infinite and the finite action of an Automaton are defined through the same recursive definition of $\bar{\lambda}$.

Example 2.8. In Figure 4 there are two equivalent Initial Automata. ♢

Convention 3. *An Initial Automaton is usually drawn depicting the Initial State with a double circle around its vertex (Figure 4).*

If an Automaton is a structure, an Initial Automaton is an alive working machine, is our model of computation, that for each input produces an output travelling through the states. I underline this concept: an Automaton doesn't define any function, till we don't fix a state q_0. It is important to bear this in mind while you read.

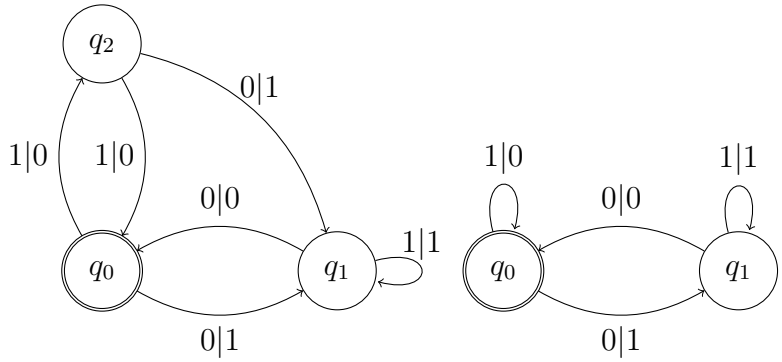


FIGURE 4. Two different Initial Automata which describe the same action. The double circle around q_0 tells us it's the Initial State.

Part 2. Automata, Trees and Algebra

Here we will show how Automata can define algebraic structures and how synchronous functions can be characterised.

Definition 2.9. Given Automata $\mathcal{A}_1 =< X, \mathcal{Q}_1, \pi_1, \lambda_1 >$ and $\mathcal{A}_2 =< X, \mathcal{Q}_2, \pi_2, \lambda_2 >$ we define their *composition* $\mathcal{B} := \mathcal{A}_1 * \mathcal{A}_2 =< X, \mathcal{Q}_1 \times \mathcal{Q}_2, \pi, \lambda >$ with π and λ defined as follows:

- $\pi(x, (s_1, s_2)) = (\pi_1(x, s_1), \pi_2(\lambda_1(x, s_1), s_2))$

- $\lambda(x,(s_1,s_2)) = \lambda_2(\lambda_1(x,s_1),s_2)$

with $x \in X$ and $(s_1,s_2) \in \mathcal{Q}_1 \times \mathcal{Q}_2$.

This is something similar to put two machine one next to the other. Informally this means we feed the second machine with the output of the first one(**??**), so we join both of them into a single bigger and more complicated transducer. Notice that the two Automata *are not* Initial Automata. Which consequences do we obtain on the Initial Automata that we can define from them?

Remark 6. Given *Initial* Automata $(\mathcal{A}_1)_{q_1}$ and $(\mathcal{A}_2)_{q_2}$, and calling respectively their actions $\bar{\lambda}^{\mathcal{A}_1}_{q_1}$ and $\bar{\lambda}^{\mathcal{A}_2}_{q_2}$, we can easily verify that:

$$\bar{\lambda}^{\mathcal{A}_2}_{q_2} \circ \bar{\lambda}^{\mathcal{A}_1}_{q_1} = \bar{\lambda}^{\mathcal{A}_1 * \mathcal{A}_2}_{(q_1,q_2)} = \bar{\lambda}^{\mathcal{B}}_{(q_1,q_2)}$$

where $\bar{\lambda}^{\mathcal{A}_1 * \mathcal{A}_2} = \bar{\lambda}^{\mathcal{B}}$ is the action of $\mathcal{A}_1 * \mathcal{A}_2 = \mathcal{B}$. This means the operation on the set of *Automata* defines an operation on the set of *Initial Automata*. However there are certain features that causes it not to be very maneuverable, so we will not use it.

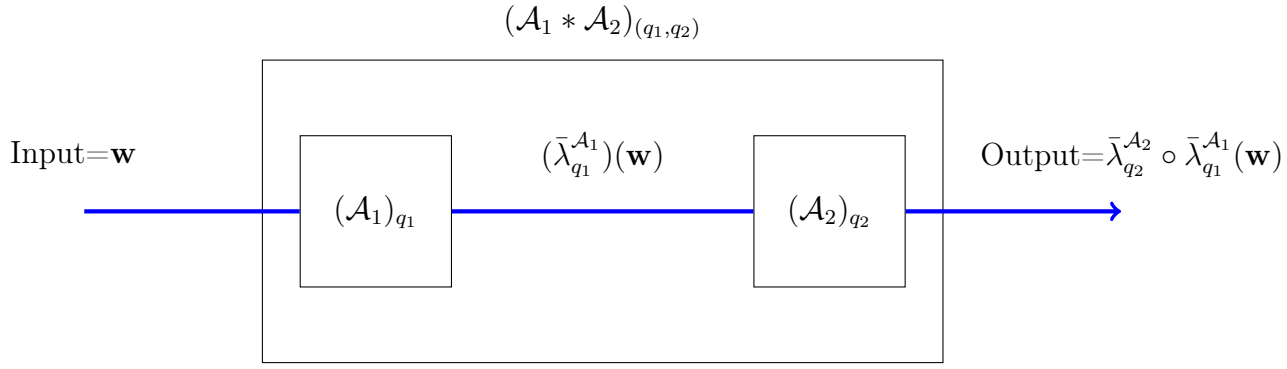


FIGURE 5. Concept of Composition of Automata in terms of Initial Automata.

Remark 7. With the operation of composition the set of all Automata on an alphabet $\mathbf{X}$ becomes a *semigroup*.

3. Synchronous Automatic functions

In this section, given an Action of an Initial Automaton, we describe and study its properties.

Definition 3.1. A transformation on $\mathbf{X}^*$(i.e. a function $f : \mathbf{X}^* \longrightarrow \mathbf{X}^*$) is called **Finite Synchronous Automatic** if it is the (finite) action of some Initial Automaton $\mathcal{A}_{q_0}$, i.e. if $f = \bar{\lambda}_{q_0}$.

Definition 3.2. A transformation on $\mathbf{X}^\omega$(i.e. a function $f : \mathbf{X}^\omega \longrightarrow \mathbf{X}^\omega$) is called **Infinite Synchronous Automatic** if it is the infinite action of some Initial Automaton.

Proposition 3.3. *The Finite Synchronous Automatic transformations form a semigroup $\mathcal{FSA}(X)$(F for "Finite", S stands for "Semigroup" and A for "Automata"). The Infinite Synchronous Automatic transformations form a semigroup isomorphic to $\mathcal{FSA}(X)$.*

Proof. The first point comes from the fact that the Composition of *Initial* Automata is an *Initial* Automaton, therefore $\mathcal{FSA}(X)$ is closed under composition of functions. The second point arise from Remark 5. □

Remark 8. If we are interested to describe the set of all possible actions, it is the same to study $\mathcal{FSA}(X)$ or its isomorphic copy. Thus we will just generally speak about **Synchronous Automatic** Transformations and we will usually refer to the finite case. Different usage will be clear from the context.

Now we provide an important characterization of Synchronous Automatic transformations:

Proposition 3.4. *A transformation $f : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ is Synchronous Automatic iff f is a tree-homomorphism on $\mathbf{X}^*$(see Figure 2 and 1.8).*

Proof. Just for the purpose of this proof, and just in the second part, we will use a more general definition of Automaton, allowing $\mathcal{Q}$ to be infinite.

($\Leftarrow$) Since f is synchronous automatic, there is an action $\bar{\lambda}_{q_0}$ of some Initial Automaton such that $f = \bar{\lambda}_{q_0}$. We need to show that $\bar{\lambda}_{q_0}$ (1) preserves the root and (2) is a graph endomorphism. By the definition $f(\varnothing) = \bar{\lambda}_{q_0}(\varnothing) = \varnothing$, thus (1) holds. Now we prove (2): if $\mathbf{v}$ is a child of $\mathbf{w}$(i.e. $\mathbf{v} = \mathbf{w}x$ for some $x \in \mathbf{X}$) is it true that $f(\mathbf{v})$ is a child of $f(\mathbf{w})$ (i.e. $f(\mathbf{v}) = f(\mathbf{w})y$ for some $y \in \mathbf{X}$)?

$$f(\mathbf{v}) = f(\mathbf{w}x) = \bar{\lambda}_{q_0}(\mathbf{w}x) = \bar{\lambda}(\mathbf{w}x, q_0) =$$
$$= \bar{\lambda}(\mathbf{w}, q_0)\bar{\lambda}(x, \bar{\pi}(q_0, \mathbf{w})) = f(\mathbf{w})\bar{\lambda}(x, \bar{\pi}(q_0, \mathbf{w}))$$

But $|\bar{\lambda}(x, \bar{\pi}(q_0, \mathbf{w}))| = 1$ because every action is lenght-preserving, thus $y = \bar{\lambda}(x, \bar{\pi}(q_0, \mathbf{w})) \in \mathbf{X}$, so $f(\mathbf{v}) = f(\mathbf{w}x) = f(\mathbf{w})y$, so (2) holds as well.

($\Rightarrow$) Let $f : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ be a tree-endomorphism. We must find an Initial Automaton such that its action is exactly f. We define $\mathcal{A} =< X, \mathcal{Q}, \pi, \lambda >:=< X, \mathbf{X}^*, \pi, \lambda >$ ($\mathcal{Q} = \mathbf{X}^*$ is *infinite*) with $\pi(\mathbf{q} \in \mathbf{X}^*, x \in \mathbf{X}^*) := qx$ and $\lambda(\mathbf{q} \in \mathbf{X}^*, x \in \mathbf{X}^*) := f(qx) - f(q)$. First: is the output function λ well defined, i.e. is the subtraction $f(\mathbf{q}x) - f(\mathbf{q})$ well defined? Yes, because f is a tree-endomorphism, so $f(\mathbf{q}x)$ is a child of $f(\mathbf{q})$. Now we need to check if $\bar{\lambda}_{\varnothing}$, the action of $\mathcal{A}_{\varnothing}$, corresponds to the function f. We verify that $\bar{\lambda}_{\varnothing}(\mathbf{w}) = f(\mathbf{w})$ by induction on $n = |\mathbf{w}|$.

(Case $n = 0$) $\bar{\lambda}(\varnothing, \varnothing) = \varnothing = f(\varnothing)$.

(Case $n \to n + 1$) Given $\mathbf{w} \in \mathbf{X}^* \setminus \{\varnothing\}$, it can be written as $\mathbf{v}x$, with $\mathbf{v} \in \mathbf{X}^*$ and $x \in \mathbf{X}$. Then $\bar{\lambda}(\varnothing, \mathbf{v}x) = \bar{\lambda}(\varnothing, \mathbf{v})\bar{\lambda}(\bar{\pi}(\varnothing, \mathbf{v}), x) = f(\mathbf{v})\bar{\lambda}(\mathbf{v}, x) = f(\mathbf{v})[f(\mathbf{v}x) - f(\mathbf{v})]$

□

Proposition 3.5. *If f is an endomorphism on $\mathbf{X}^*$, then $f(X^n) \subset X^n$. In particular if f is an automorphism $f(\mathbf{X}^n) = \mathbf{X}^n$, i.e. a permutation on $\mathbf{X}^n$.*

Proof. It can be easily proved by induction on n. □

Remark 9. The last proposition is a graph persepective on the lenght-preserving condition of actions of Automata.

Definition 3.6. Given a *tree-endomorphism* $g : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ and $\mathbf{v} \in \mathbf{X}^*$, we can define its **restriction in v** as the function $g_{|\mathbf{v}} : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ defined by the equality:

$$g(\mathbf{vw}) = g(\mathbf{v})g_{|\mathbf{v}}(\mathbf{w})$$

Remark 10. Since g is a tree-endomorphism, and since $\mathbf{v}\mathbf{X}^* \backsimeq g(\mathbf{v})\mathbf{X}^* \backsimeq \mathbf{X}^*$ (where $\backsimeq$ indicates that the two objects are isomorphic as trees) we have that $g : \mathbf{v}\mathbf{X}^* \longrightarrow g(\mathbf{v})\mathbf{X}^*$ is identifiable as $g|_{\mathbf{v}} : \mathbf{X}^* \longrightarrow \mathbf{X}^*$, and the latter one is consequently a tree endomorphism.

Let's give a description of the restriction $g_{|\mathbf{v}}$ in terms of Automata.

Proposition 3.7. *If $\bar{\lambda}_{q_0} : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ is the action of $\mathcal{A}_{q_0}$, then, for every $\mathbf{v} \in \mathbf{X}^*$, the action of $\mathcal{A}_{\bar{\pi}(\mathbf{v},q_0)}$ is given by $(\bar{\lambda}_{q_0})|_{\mathbf{v}} = \bar{\lambda}_{\bar{\pi}(\mathbf{v},q_0)}$, i.e. the restriction of $\bar{\lambda}_{q_0}$ in $\mathbf{v}$.*

Proof. Given $\mathbf{v}, \mathbf{w} \in \mathbf{X}^*$ we can easily prove by induction on $n = |\mathbf{w}|$ that $g(\mathbf{vw}) := \bar{\lambda}_{q_0}(\mathbf{vw}) = \bar{\lambda}_{q_0}(\mathbf{v})\bar{\lambda}_{\bar{\pi}(\mathbf{v},q_0)}(\mathbf{w}) = g(\mathbf{v})\bar{\lambda}_{\bar{\pi}(\mathbf{v},q_0)}(\mathbf{w})$, consequently $g|_{\mathbf{v}} = \bar{\lambda}_{\bar{\pi}(\mathbf{v},q_0)}$. □

4. Groups generated by Automata

The groups we will consider are subgroups of $\mathcal{FSA}(X)$, so we will need to have *invertible* elements, i.e. *invertible* transformations f in $\mathcal{FSA}(X)$. This means imposing a special condition on all the possible Initial Automata $\mathcal{A}_{q_0}$ describing f, i.e. $\bar{\lambda}_{q_0}$ is invertible. To formalise this concept there will be a bit of technical work.

Definition 4.1. Given an Initial Automaton $\mathcal{A}_{q_0}$, a state q is *accessible* if there exists a word $\mathbf{w} \in \mathbf{X}$ such that $\bar{\pi}(\mathbf{w}, q_0) = q$. We can also say that q *is accessible with respect to* q_0 or *from* q_0.

Practically this means that in the Moore Diagram there's a path from q_0 to q for the vertex $q \in \mathcal{Q}$.

Definition 4.2. An Initial Automaton $\mathcal{A}_{q_0}$ is called *accessible* if each $q \in \mathcal{Q}$ is accessible with respect to q_0. An Automaton is called *accessible* if each Initial Automaton definable by it is accessible.

Proposition 4.3. *Given an Automaton $\mathcal{A} =< X, \mathcal{Q}, \pi, \lambda >$ and a state $q_0 \in \mathcal{Q}$, $\bar{\lambda}_{q_0}$ is an invertible function if and only if for every accessible state $q \in \mathcal{Q}$ (respect to q_0) $\lambda_q : \mathbf{X} \longrightarrow \mathbf{X}$ is invertible.*

Proof. ($\Rightarrow$) Suppose that $\bar{\lambda}_{q_0}$ is an invertible function. Let us take an accessible state $q \in \mathcal{Q}$ and a word $\mathbf{w}$ such that $\bar{\pi}_{q_0}(\mathbf{w}) = q$. Let us consider $\lambda_q : \mathbf{X} \longrightarrow \mathbf{X}$. We check that λ_q is injective. Given $x \neq y$, from the converse we suppose that $\lambda_q(x) = \lambda_q(y)$. We would have that

$$\bar{\lambda}_{q_0}(\mathbf{w}x) = \bar{\lambda}_{q_0}(\mathbf{w})\underbrace{\bar{\lambda}_{\bar{\pi}(\mathbf{w},q_0)}(x)} = \bar{\lambda}_{q_0}(\mathbf{w})\underbrace{\lambda_q(x)} = \bar{\lambda}_{q_0}(\mathbf{w})\lambda_q(y) = \bar{\lambda}_{q_0}(\mathbf{w}y)$$

Consequently we would loose the injectivity of $\bar{\lambda}_{q_0}$, against the hypothesis of its invertibility.

Analogously we can see that λ_q is surjective: let us take a word $\mathbf{w} \in \mathbf{X}^*$ such that $\bar{\pi}(\mathbf{w}, q_0) = q$ and $y \in \mathbf{X}$. We search an $x \in \mathbf{X}$ such that $\lambda_q(x) = y$.

Since $\bar{\lambda}_{q_0}$ is invertible and synchronous automatic the word $\bar{\lambda}_{q_0}(\mathbf{w})y$ has a unique preimage, and it is of the form $\mathbf{w}x$ for some x. Therefore: $\bar{\lambda}_{q_0}(\mathbf{w})y = \bar{\lambda}_{q_0}(\mathbf{w}x) = \bar{\lambda}_{q_0}(\mathbf{w})\lambda_q(x)$.

($\Leftarrow$) The transition function moves necessarily to an accessible state q for each $\mathbf{w} \in \mathbf{X}^*$ provided. We know that $\lambda_p : \mathbf{X} \longrightarrow \mathbf{X}$ is invertible (i.e is a permutation of $\mathbf{X}$) for each accessible p, including all the states on the path to q. Now we will prove that $\bar{\lambda}_{q_0}$ is invertible on $\mathbf{X}^n$ by induction on n, consequently it will be invertible on $\bigcup_{n\in\mathbb{N}} \mathbf{X}^n = \mathbf{X}^*$.

($n = 1$) On $\mathbf{X}$ we have $\bar{\lambda}_{q_0} = \lambda_{q_0}$, therefore $\bar{\lambda}_{q_0}$ is invertible for hypothesis.

($n \to n+1$) Let us suppose that $\bar{\lambda}_{q_0}$ is invertible on $\mathbf{X}^n$. If $\mathbf{v} \in \mathbf{X}^{n+1}$ then $\mathbf{v} = \mathbf{w}x \in \mathbf{X}^n \times \mathbf{X}$ with $|\mathbf{w}| = n$. Thus $\bar{\lambda}_{q_0}(\mathbf{v}) = \bar{\lambda}_{q_0}(\mathbf{w}x) = \bar{\lambda}_{q_0}(\mathbf{w})\bar{\lambda}_{\bar{\pi}(\mathbf{w},q_0)}(x) = \bar{\lambda}_{q_0}(\mathbf{w})\lambda_p(x)$ for some p. We observe now that if we change $\mathbf{w}$ or x, we will obtain a different image with respect to $\bar{\lambda}_{q_0}$ on $\mathbf{X}^n$ and with respect to λ_{q_0} on $\mathbf{X}$ (injectivity). And if we search for the preimage of a word $\bar{\mathbf{w}}\bar{x} \in \mathbf{X}^{n+1}$, we know that there exists a preimage $\mathbf{w}$ of $\bar{\mathbf{w}}$ through $\bar{\lambda}_{q_0}$ and a preimage x of $\bar{x}$ with respect to $\lambda_{\bar{\pi}(\mathbf{w},q_0)}$. If we glue them together we obtain:

$$\bar{\lambda}_{q_0}(\mathbf{w}x) = \bar{\lambda}_{q_0}(\mathbf{w})\lambda_{\bar{\pi}(\mathbf{w},q_0)}(x) = \bar{\mathbf{w}}\bar{x}$$

So the surjectivity is proven.

□

Proposition 4.4. *The set $\mathcal{GA}(X)$, the set of all bijective synchronous automatic transformations on an alphabet X, is a group respect to the compostion operation. Besides it is isomorphic to $\mathcal{AUT}_{tree}(\mathbf{X}^*)$, the set of all tree-automorphism of $\mathbf{X}^*$.*

Proof. Since $\mathcal{FSA}(X)$ is isomorphic to the group of all tree-endomorphisms on $\mathbf{X}^*$, $\mathcal{GA}(X)$ is the image of the set of all bijective tree-endomorphisms on $\mathbf{X}^*$, i.e. is the image of $\mathcal{AUT}_{tree}(\mathbf{X}^*)$, which is a group. Consequently $\mathcal{GA}(X)$ is a group. □

Definition 4.5. An *Initial Automaton* $\mathcal{A}_{q_0}$ is called **invertible** if its action is invertible. An *Automaton* $\mathcal{A}$ is called **Invertible** if $\mathcal{A}_{q_0}$ is invertible for each $q_0 \in \mathcal{Q}$.

Convention 4. *Henceforth by an Automaton and an Initial Automaton we mean an **Invertible Automaton** and an **Invertible Initial Automaton**, respectively.*

Convention 5. *From now on we will sometime use a different label-notation for the Moore Diagram. Given an edge $q \longrightarrow p$, on its label there will be written not "$x|\lambda(x,q)$" but simply "x", and on the vertex q not "q" but the name of the application "$\lambda(\cdot,q)$". See Figure 6.*

Now we can finally introduce the last brick of the bridge between Automata and Algebra.

Definition 4.6. Given an Automaton $\mathcal{A} =< \mathbf{X}, \mathcal{Q}, \pi, \lambda >$ we can define $|\mathcal{Q}|$ Initial Automata, which define $|\mathcal{Q}|$ actions $\bar{\lambda}_q$ on $\mathbf{X}^*$ inside $\mathcal{FSA}(X)$. By the **Group Generated by** $\mathcal{A}$ we mean the subgroup of $\mathcal{GA}(X)$ generated by all the actions $\bar{\lambda}_q$ with $q \in \mathcal{Q}$:

$$< \{\bar{\lambda}_q : \mathbf{X}^* \longrightarrow \mathbf{X}^* | q \in \mathcal{Q}\} >$$

Where $< S >$ for a set S means the group generated by the elements of S.

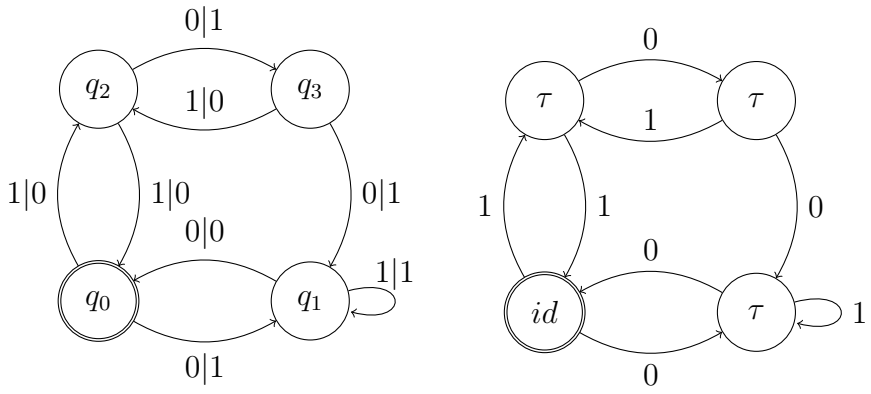


FIGURE 6. An example of the same Initial Automaton represented with the two different notations. In the right figure $\tau, id \in \mathcal{S}_2 := \mathcal{S}(\{0,1\}$, where, τ inverts the elements, and id leave them unchanged.

Part 3. Group products and their applications in this context

In mathematics "to understand" and "to represent in an appropriate way" are very often the same concept, and we could open a very interesting philosophical and historical debate on this declaration. In our case we need some advanced notions of Group Theory to understand, write down and work properly on the examples we are going to consider. I alert the reader: this section is quite technical.

5. Actions, Semidirect Product and Wreath Product

These quite complicated structures, regardless of the skeptisism of the reader, arise naturally very often in Algebra. This is particularly the case in environments involving some kind of recursion or selfsimilarity, as Automata do. The order of construction will be: Actions, Faithful Actions, Actions as a Group, Semidirect Product, Wreath Product.

Convention 6. *Given a set X, $\mathcal{S}(X)$ denotes the **Symmetrical Group of** X, so the group of all permutations $\sigma : X \longrightarrow X$.*

Proposition 5.1. *Let $(H, *)$ be a group provided with the operation $*$. We can define another operation on H, called the **opposite operation of** $*$ on H, defined as $f *' H := H * f$. Then $(H, *')$, called the opposite group of H, is isomorphic to $(H, *)$ by*

$$op_H : (H, *) \longmapsto (H, *')$$
$$op_H : g \longmapsto g^{-1}$$

Notice that the inverse of op_H is precisely op_H.

Proof. We have that $op_H(g*h) = (g*h)^{-1} = h^{-1}*g^{-1} = g^{-1}*'h^{-1} = op_H(g)*'op_H(h)$. It is obviously bijective, and its inverse is itself. □

Definition 5.2. Let $\circ$ be the composition of functions. We call the **product** of functions the operation "$\cdot$" defined as $f \cdot g := g \circ f$. Given a group of functions provided with the operation of composition of functions, it is obvious that the product of functions is its opposite operation.

5.1. **Step 0: Actions and Faithful Actions.** The algebraic actions (not to be confused with actions of Initial Automata previously defined) we will treat are the first brick of the Wreath Product, vital to understand Automata. There are two equivalent definitions for an action. The purpose of the first one is to give a more abstract characterization, which is comfortable to work with. The second one helps us to visualise what we are doing as a strange "multiplication", and to write it more easily.

The abstract definition is:

Definition 5.3. Given a group G and a set X,we call a **G-left-action on X as a set** or an **left-action of G on X as a set** an *homomorphism of groups* $T_l : G \longrightarrow (\mathcal{S}(X), \circ)$. We can also say that G *acts on X* ***as a set*** *from the left by* T_l. Equivalently we say that G acts on X as a set from the right by T_r if there exists an homomorphism $T_r : G \longrightarrow (\mathcal{S}(X), \cdot)$.

The more visualisable definition:

Proposition 5.4. *Let us take a group $(G, *)$ and a set X. Then G is acting on X from the left iff exist a function $\tau_l : G \times X \longrightarrow X$ such that:*

- $\tau_l(1, x) = x$ *for every* $x \in X$
- $\tau_l(g, \tau_l(h, x)) = \tau_l(g * h, x)$ *for every* $x \in X$ *and* $g, h \in G$

In this case we write $gx := \tau_l(g, x)$.

Proof. ($\Leftarrow$) We define $(T_l(g))(x) := \tau_l(g, x)$, therefore, for the property of τ_l we have $T_l(g * h)(x) = \tau_l(g * h, x) = \tau_l(g, \tau_l(h, x)) = T_l(g)(\tau_l(h, x)) = (T_l(g) \circ T_l(h))(x)$ for every $x \in X$

($\Rightarrow$) Analogous to the other sense.

□

Example 5.5 (Translations)**.** Given (V, A), where A is an affine space built on the vector space V, then V taken as a group is acting from the left on A by translating its points:

$$\tau_l(v, p) := v + p$$

♢

Proposition 5.6. *Given $(G, *)$ and X as in the previous statement, we have that G is acting on X from the right iff exist a function $\tau_r : X \times G \longrightarrow X$ such that:*

- $\tau_r(x, 1) = x$ *for every* $x \in X$
- $\tau_r(\tau_r(x, h), g) = \tau_r(x, h *_G g)$ *for every* $x \in X$ *and* $g, h \in G$

In this case we write $xg := \tau_r(x, g)$.

Proof. Analogous to the left case. □

What is the main difference between acting from the left or from the right? The order in which we let the element of G act on X. Let's see some examples:

Example 5.7. The symmetrical group $(\mathcal{S}(X), \circ)$ of a set X acts on X as a set from the left, in fact we can define $\tau_l(\sigma, x) = \sigma x := \sigma(x) \qquad \forall \sigma \in \mathcal{S}(X)$ and $\forall x \in X$. Analogously we can define $\tau_r(x, \sigma) = x\sigma := \sigma(x)$. Let X be such that $|X| > 3$, then $\mathcal{S}(X)$ is not abelian. Let us take $\sigma, \eta \in \mathcal{S}(X)$ such that $\sigma \circ \eta \neq \eta \circ \sigma$. Then there exists x such that $x\sigma\eta \neq \sigma\eta x$. This shows us that the difference between left and right actions becomes quite important! ♢

Example 5.8 (Translations)**.** Given instead the example 5.5, since the group of translations V is abelian, the left and right action definable through it on A yield always to the same result:

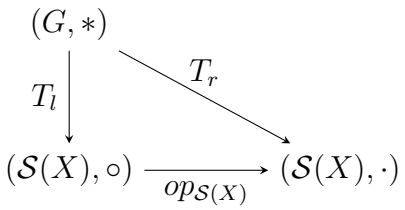
$$p + v + w \;=\; p + w + v \;= v + w + p \;=\; w + v + p$$

$\diamondsuit$

In general the difference between left and right actions becomes vital when we are treating with non-abelian groups.

Proposition 5.9. *Let G be a group acting on X as a set by the left with $T_l : G \longrightarrow (\mathcal{S}(X), \circ)$, and let $op_{\mathcal{S}(X)}$ be as defined in 5.1. Then $T_r := op_{\mathcal{S}(X)} \circ T_l : (G, *) \longrightarrow (\mathcal{S}(X), \cdot)$ is a right action of G on X such that $x\, gh = h^{-1}g^{-1}\, x$.*

Proof. We have the diagram: Then



$$\begin{aligned} x\, g * h = x\, gh = \; & T_r(g * h)\,(x) = op_{\mathcal{S}(X)}(T_l(g * h))\,(x) = \\ & = (T_l(g * h))^{-1}\,(x) = (T_l(h^{-1} * g^{-1}))\,(x) = \\ & = T_l(h^{-1}) \circ T_l(g^{-1})\,(x) = h^{-1}g^{-1}\, x = h^{-1} * g^{-1}\, x \end{aligned}$$

$\square$

The last proposition shows us a very useful way to convert a left action into a right action, or, if we prefer, a way to let a product of elements of G act in the reverse order on X. This becomes extremely tricky when we handle the symmetrical group $\mathcal{S}(X)$, mainly because $\circ$ is the opposite operation of $\cdot$. Take in fact $\mathcal{S}(X)$ is provided with the operation $\circ$, acting from the left on X. Let T_l the left-action previously seen (5.7). If T_r is the right action we derive from it, we write $x\sigma^{-1} = T_r(\sigma)(x) = \sigma(x)$. Then we have that:

$$x\sigma^{-1}\eta^{-1} = x(\eta\sigma)^{-1} = T_r(\eta \circ \sigma)(x) = (\eta \circ \sigma)(x) = (\sigma \cdot \eta)(x)$$

This motivates the introduction of this notation:

Convention 7. *We sign $(\sigma \cdot \eta)(x)$ as $x \cdot \sigma \cdot \eta$. For this reason, whenever we will encounter $(X, \mathcal{S}(X))$, we will assume that $\mathcal{S}(X)$ is provided with the operation $\circ$, and that $(\mathcal{S}(X), \circ)$ acts on X from the right as a set as:*

$$x\, fg := (f \cdot g)\,(x) \;=\; (g \circ f)(x)$$

Definition 5.10. Let G be a group acting on X as a set by left-action $T_l : G \longrightarrow (\mathcal{S}(X), \circ)$. Then T_l is called **faithful** if it's *injective*. We then say that G acts *faithfully* on X by T. In this case we say that (G, X) is a **Left Permutation Group**. Analogous definition can be given to the **Right Permutation Group** that we sign as (X, G) and to the faithful right action as set.

Remark 11. Take care: the Symmetric Group $\mathcal{S}(Y)$ is the set of *all* permutations of Y, while a Left Permutation Group (B, Y) is, mangling the definition, a subgroup of $\mathcal{S}(Y)$. If we have B group, we can take (B, B) as a left permutation group, with B acting on itself faithfully as a group by left multiplication. In this case B is effectively a subgroup of $\mathcal{S}(B)$.

Proposition 5.11. *A group G acts faithfully on a set X from the left iff for every h and g in G there exists an x in X such that $gx \neq hx$. A group G acts faithfully on a set X from the right iff for every h and g in G there exists an x in X such that $xg \neq xh$.*

Proof. ($\Leftarrow$): Let's take $T : G \longrightarrow (\mathcal{S}(X), \circ)$ defined by $T(g) = \phi_g$ where $\phi_g(x) := gx$. We can easily verify its injectivity and surjectivity, thus the thesis. ($\Rightarrow$): The group G is embeddable in $(\mathcal{S}(X), \circ)$ by T, thus we identify each g with ϕ_g, where $\phi_g(x) := gx$. Since ϕ is unique, we have that G acts faithfully on X. □

Till now we have seen actions **as a set**. What if the set X, acted upon, is also a group? Is there any way to also preserve the structure of group of X while we act on it? In other words, given G group, can he act on X such that

$$g\,(h *_H h') = gh\ *_H\ gh'$$

for $g \in G$ and $h, h' \in H$?

Definition 5.12. Given two groups H and N, we say that H *acts on N* ***as a group*** *from the left* if there exists an homomorphism $\gamma : H \longrightarrow (\mathcal{AUT}(N), \circ)$, where $(\mathcal{AUT}(N), \circ)$ is the group of all group-automorphisms on N. If instead there exists an homomorphism $\varphi : H \longrightarrow (\mathcal{AUT}(N), \cdot)$ we say that H acts on N as a group from the right.

Remark 12. Note that the group $\mathcal{AUT}(N)$ is a subgroup of $\mathcal{S}(N)$ (in symbols $\mathcal{AUT}(N) \subsetneq \mathcal{S}(N)$). We can consequently observe that if H acts on N *as a group* from the left ($\gamma : H \longrightarrow (\mathcal{AUT}(N), \circ)$), H acts on N *as a set* from the left, but **in general is not true the opposite**. The same can be applied for the actions from the right. By the way, since γ is also an action as a set, we can sign $\gamma(h)(n)$ as hn with $h \in H$ and $n \in N$. Equivalently, having $\varphi : H \longrightarrow (\mathcal{AUT}(N), \cdot)$, we can sign $\varphi(h)(n)$ as nh.

Proposition 5.13. *Let H and N be two groups, and let $\gamma : H \longrightarrow (\mathcal{AUT}(N), \circ)$ be the left-action of H on N as a set. Then $\varphi := op_{\mathcal{AUT}(N)} \circ \gamma : H \longrightarrow (\mathcal{AUT}(N), \cdot)$ is a right-action of H on N as a group given by $xh = \varphi(h)(x) = \gamma(h^{-1})(x) = h^{-1}x$.*

Proof. Analogous to the case treated with 5.1. □

5.2. **Step 1: Actions as a Group and Semidirect Product.** Be careful: we will define Semidirect Products using **right** actions as groups. There are possible definitions also with left actions as groups.

Definition 5.14. Let H, N be two groups, with operations $*_H$ and $*_N$, where H acts on N *as a group* **from the right** by $\varphi : H \longrightarrow (\mathcal{AUT}(N), \cdot)$. With this we define on $H \times N$ the following operation:

$$\star_\varphi : ((h_2, n_2), (h_1, n_1)) \longmapsto (\ h_2 *_H h_1\ ,\ \varphi(h_1)(n_2) *_N n_1)\ ) = (\ h_2h_1\ ,\ (n_2h_1) *_N n_1)\ )$$

We call $(H \times N, \star_\varphi)$ the **Semidirect Product of H and N relative to φ** and we write it down as $H \ltimes_\varphi N$. We can also refer to φ as the **underlying homomorphism** of the Semidirect Product between H and N.

To avoid future confusion, remember that the open side of $\ltimes$ goes towards the group acted upon as a group.

Proposition 5.15. *The Semidirect Product $H \ltimes_\varphi N$ is a group, where the identity is $(1_H, 1_N)$ and $(h^{-1}, \varphi(h^{-1}))(n^{-1}))$ is the inverse for each (h, n) in $H \ltimes_\varphi N$.*

Proof. We prove just the associativity, which is trickier than it seems. The rest of the proof is a verification. Let $(h'', n''), (h', n'), (h, n)$ be elements of $H \ltimes_\varphi N$. Then:

$$\begin{aligned}
((h'', n'') * (h', n')) \quad * \ (h, n) &= (h''h' \ , \ \varphi(h')(n'') \ * n') \ * \ (h, n) = \\
&= (h''h'h \ , \ (\varphi(h)(\ (\varphi(h')(n'')) * n' \ ) \ * n) = \\
&= (h''h'h \ , \ (\varphi(h) \circ \varphi(h)) \ (n'') \ * \ \varphi(h)(n') \ * n) = \\
&= (h''h'h \ , \ (\varphi(h') \cdot \varphi(h)) \ (n'') \ * \ \varphi(h)(n') \ * n) = \\
&= (h''h'h \ , \ \varphi(h'h)(n'') \ * \ \varphi(h)(n') \ * n) = \\
&= (h'', n'') \ * \ (h'h, \varphi(h)(n') \ * n) = \\
&= (h'', n'') \ * \quad ((h', n') * (h, n))
\end{aligned}$$

□

Example 5.16 (Dihedral Groups)**.** Among the first groups studied there were the groups of symmetries, which informally can be described as, given a geometric object A, one can consider the set of all geometrical transformation which have A unchanged. Let A be a regular polygon with n sides, identified with its corners (which means so $\mathbb{Z}_n$ or the group of the n-th roots of unity in $\mathbb{C}$). The group of symmetries of this figure is $\mathcal{D}_n$, the so called **n-Dihedral Group**. There are two types of transformations in it, the rotation of $\frac{k\pi}{n}$ degrees around the centre of the polygon, and the reflection with respect to one of the n possible axes of simmetry.

FIGURE 7. All the possible symmetries of an octagon visualised using a sign of STOP. The upper ones are all the rotations (elements $(0, k)$), and the lowest one all the reflections (elements $(1, k)$). The image has been taken from the section of wikipedia regarding dihedral groups

It turns out that this group is isomorphic to the Semidirect Product $\mathbb{Z}_2 \ltimes_\varphi \mathbb{Z}_n$, where $\varphi(0)(z) := id_{\mathbb{Z}_n}(z) = z$ and $\varphi(1)(z) := inv_{\mathbb{Z}_n}(z) := -z \, (mod \ n)$. So $\mathbb{Z}_2 \times \mathbb{Z}_n \ni (h_2, n_2) * (0, n_1) = (h_2 + 0, n_1 + n_2)$ and $(h_2, n_2) * (1, n_1) = (h_2, n_1 - n_2)$. We can notice that $(h, k) = (h, 0) * (0, k)$. The transformation $(0, k)$ is necessarily always a rotation, while $(h, 0)$ is the identity or the reflection through the central axis, depending if $h = 0$ or $h = 1$. Practically, if we have $(h, k) \in \mathbb{Z}_2 \times \mathbb{Z}_n$, h encodes the reflection and k the rotation. ♢

5.3. **Step 2: Restricted and Unrestricted Wreath Product.**

Definition 5.17. Given a group A and an arbitrary set of indices Y we define the **Direct Product** as:

$$A^Y = \prod_{\omega \in Y} A := \{\overline{a} = (a_\omega)_{\omega \in Y} : a \in A\}$$

and the **Direct Sum** as:

$$A^{(Y)} := \bigoplus_{\omega \in Y} A := \{\widetilde{a} = (a_\omega)_{\omega \in Y} : a \in A \text{ and } a \neq 1_A \text{ just for a finite number of indexes}\}$$

In case $|Y|$ is finite there's no difference between the two structures.

Remark 13. If A is a group we can extend its operation $*_A$ on the two structures component-wise, to obtain so another 'bigger' group.

Convention 8. *I alert the reader who might have seen other times the Semidirect Product and the Wreath Product: you should notice that these structures are not following a uniformous notation yet, and can vary slightly their definition depending on the context and the author.*

Now let us take (Y, B) a right permutation group and A a group. If we construct A^Y we have a group on which B can act faithfully from the right (permuting the indices Y, so we have an injective homomorphism $\Phi : B \longrightarrow (\mathcal{S}(A^Y), \cdot)$). Supposing we prove that $\Phi(B) \subset \mathcal{AUT}(A^Y)$, Φ is an action **as a group** from the right. Then we have everything we need to construct $B \ltimes_\Phi A^Y$. The same can be done substituting A^Y with $A^{(Y)}$.

Let us formalise this:

Proposition 5.18. *Given (Y, B) right permutation group and A group, we can extend the right-action of B to A^Y as a set faithfully. Consequently (A^Y, B) is a right permutation group. In addition, the action of B to A^Y is an action **as a group** from the right, where A^Y is provided with the component-wise operation on A. The same can be done substituting A^Y with $A^{(Y)}$.*

Proof. If $y\beta$ is the right-action of $\beta \in B$ on $y \in Y$, we can define the action Φ of β on $\overline{a} \in A^Y$ as follow:

$$\Phi(\beta \in B)\,(\overline{a} \in A^Y) = \Phi_\beta(\overline{a}) = \overline{a}\beta = (a_y)_{y \in Y}\beta := (a_{y\beta})_{y \in Y} = (a_y)_{y\beta^{-1} \in Y}$$

(1) We must prove that Φ is a right action on A^Y as a set. So that $\Phi(\beta)$ is bijective for every $\beta \in B$. So, for the injectivity we must prove that if $\overline{a} = (a_y)_{y \in Y} \neq (x_y)_{y \in Y} = \overline{x}$ there exists$y' \in Y$ such that $\Phi(\beta)(a_{y'}) \neq \Phi(\beta)(x_{y'})$. Let us look $\Phi_\beta(\overline{a})$ at the index $y'\beta$. We find that at the $y\beta'$-th component $a_{y\beta'}$ is different from $x_{y\beta'}$. So Φ_β is injective. The surjectivity is very simple: if we have $(a_y)_{y \in Y}$ the element $(a_{y\beta^{-1}})_{y \in Y}$ is its counter-image.

(2) We have now to prove the action is an action on A^Y as a group, i.e. that Φ_β is an automorphism: $\Phi_\beta(\overline{a} \star \overline{x}) = \Phi_\beta((a_y \star x_y)_{y \in Y}) = (a_{y\beta} \star x_{y\beta})_{y \in Y} = (a_{y\beta})_{y \in Y} \star (x_{y\beta})_{y \in Y} = \Phi_\beta(\overline{a}) \star \Phi_\beta(\overline{x})$

(3) We must prove that Φ is a faithful i.e. that if $\beta \neq \theta$ then $\Phi_\beta \neq \Phi_\theta$. If $\beta \neq \theta$, there exists y' such that $\beta(y') \neq \theta(y')$. Let us take $(a_y)_{y \in Y}$ and $(x_y)_{y \in Y}$ such that $a_{y'} \neq x_{y'}$. Then the elements $\Phi_\beta((a_y)_{y \in Y}) = (a_{y\beta})_{y \in Y}$ and $\Phi_\theta((x_y)_{y \in Y}) = (x_y\beta)_{y \in Y}$ at the index y'.

If we take $A^{(Y)}$ the proof is still valid because the permutation of indices always keeps a finite number of them different from the identity. □

Keep in mind that all this is still valid also if $(B, *) = (\mathcal{S}(X), \circ)$ and the right permutation group is $(Y, B) = (X, \mathcal{S}(X))$ with the notation introduced in 7.

Definition 5.19. Let (Y, B) be a right permutation group and let A be a group. We have then a right-action as a group $\Phi : B \longrightarrow (\mathcal{AUT}(A^Y), \cdot)$ defined as in 5.18. We call **Wreath Product**, and we sign it $B \wr A$ one of these two structures:

- The **Unrestricted Wreath Product**, is the semidirect product $B \ltimes_\Phi A^Y = B \wr A$ with $\Phi : B \longrightarrow (\mathcal{AUT}(A^Y), \cdot)$. We sign $(\Phi(\beta))(\overline{a} = (a_y)_{y\in Y}) = (a_{y\beta})_{y\in Y}$, where $y\beta$ is the right-action of $\beta \in B$ on $y \in Y$,
- The **Restricted Wreath Product**, is the semidirect product on $B \ltimes_\Phi A^{(Y)} = B \wr A$ where $\Phi : B \longrightarrow (\mathcal{AUT}(A^{(Y)}, \cdot)$. We sign $\Phi(\beta)$ as in the previous case.

Therefore, having $(\beta, \overline{p}), (\theta, \overline{q})$ in $B \times A^Y$ (or in $B \times A^{(Y)}$) their product is:

$$(\beta, \overline{p}) * (\theta, \overline{q}) = (\beta, (p_y)_{y\in Y}) * (\theta, (q_y)_{y\in Y}) := \\ = (\beta *_B \theta, (\Phi(\theta))(\overline{p}) * \overline{q}) = (\beta *_B \theta, (p_{y\theta} * q_y)_{y\in Y})$$

If we take a two group B, A we can anyway consider their wreath product considering (B, B) a right permutation group, where B acts faithfully on himself (as a set) by right multiplication.

Proof. Consequences of the previous proposition and of the semidirect product construction. □

Remark 14.
- From the context it will be understandable, or it will be stated, of which of the two structure are we talking about. Notice that if Y is finite there's no difference between the restricted and unrestricted wreath product.
- Be very careful: given the right permutation group $(X, \mathcal{S}(X))$ and a group A, because of the notation introduced in 7, given elements $(\eta, (r_x)_{x\in X})$, $(\beta, (p_x)_{x\in X})$, $(\theta, (q_x)_{x\in X})$ of $\mathcal{S}(X) \wr A = \mathcal{S}(X) \ltimes_\Phi A^X$, their product is:
$$(\eta \circ \beta \circ \theta \ , \ (r_{x\theta\beta} * p_{x\theta} * q_x)_{x\in X})$$
Where in the subscript $x\theta\beta$ we are acting by the right through product of functions.

Convention 9. *To work with wreath product, when $|Y|$ is finite, is usual to use a more precise notation. Let $(B, Y), A, B \wr A = B \ltimes_\Phi A^Y$ be as previously defined, with Y **finite** with k elements. Let us fix an indexing of Y $\{y_1, \dots, y_k\}$. Then $\overline{a} \in A^Y$ can be uniquely written as $(a_1, \dots, a_k)$. Then we write the generic element $(\beta, \overline{a}) \in B \wr A$ as $\beta(a_1, \dots, a_k)$. With this convention, given $\beta(a_1, \dots, a_k)$ and $\theta(g_1, \dots, g_k)$ in $B \wr A$, the multiplication rule becomes:*

$$\beta(a_1, \dots, a_k) * \theta(g_1, \dots, g_k) = \beta\theta((a_{1\theta}, \dots, a_{k\theta}) *_{A^Y} (g_1, \dots, g_k)) = \\ = \beta\theta(a_{1\theta}g_1, \dots, a_{k\theta}g_k)$$

With this notation, having $\beta(a_1, \dots, a_k)$, its inverse becomes:

$$\beta^{-1}((g_{1\beta^{-1}})^{-1}, \dots, (g_{k\beta^{-1}})^{-1})$$

Remark 15. Semidirect and wreath products arise often in mathematics. Interesting examples involve groups used to understand and solve Sudoku or the Rubik's Cube. Otherwise John Rodes in [**?**] states many examples of the applications of the theory of Automata, which as we will see, are deeply bounded with the structure of the Wreath Product.

After all this technical work we finally breathe with some examples.

6. Applications to Automata

We will see the wreath product in the context of Automata. First we will need to gather all the ingredients.

Proposition 6.1. *Let* $\mathbf{X}$ *be an alphabet. Denote by* $\mathcal{AUT}_{tree}(\mathbf{X}^*)$ *the set of tree-automorphisms on* $\mathbf{X}^*$*. Then there exists a* $\mathcal{AUT}_{tree}(\mathbf{X}^*)$*-left-action* T *on* $\mathbf{X}$ *as a set (*$T : (\mathcal{AUT}_{tree}(\mathbf{X}^*), \circ) \longrightarrow (\mathcal{S}(\mathbf{X}), \circ)$*) defined by* $T(f)(x) := f(x)$*.*

Proof. The function f is a tree-automorphism, so for 3.5, $f(\mathbf{X}) = (\mathbf{X})$, therefore $T(f)$ is a bijection on the alphabet $\mathbf{X}$. □

We now take the right permutation group $(\mathbf{X}, \mathcal{S}(\mathbf{X}))$ and the group $\mathcal{AUT}_{tree}(\mathbf{X}^*)$, where both $\mathcal{S}(\mathbf{X})$ and $\mathcal{AUT}(\mathbf{X}^*)$ are provided with the composition of functions $\circ$ as their operation. We have so an extension to a right-action as a group of $\mathcal{S}(\mathbf{X})$ on $\mathcal{AUT}_{tree}(\mathbf{X}^*)$ as $\Phi(\sigma \in \mathcal{S}(\mathbf{X}))(f_{x_1}, \ldots, f_{x_k}) = (f_{x_1\sigma}, \ldots, f_{x_k\sigma})$, where $x_i\sigma$ is the element $\sigma(x_i) \in \mathbf{X}$, and f_{x_1} is the restriction of f as defined in 3.6. We have now all the ingredients to define a very useful the wreath product in the context of automata.

Proposition 6.2. *Given an alphabet* $\mathbf{X}$ *of cardinality* k *let us arrange its elements into an order* $(x_1, \ldots, x_k)$ *and let* T *be as in the previous proposition. Let us take right permutation group* $(\mathbf{X}, \mathcal{S}(\mathbf{X}))$*, where* $\mathcal{S}(\mathbf{X})$ *acts as a group from the right on* $\mathcal{AUT}_{tree}(\mathbf{X}^*)^{\mathbf{X}}$ *by* $\Phi(\sigma \in \mathcal{S}(\mathbf{X}))(f_{x_1}, \ldots, f_{x_k}) = (f_{x_1\sigma}, \ldots, f_{x_k\sigma})$*. Let us define* $\psi : (\mathcal{AUT}_{tree}(\mathbf{X}^*), \circ) \longrightarrow (\mathcal{S}(X), \circ) \wr (\mathcal{AUT}_{tree}(\mathbf{X}^*), \circ) = \mathcal{S}(\mathbf{X}) \ltimes_\Phi \mathcal{AUT}_{tree}(\mathbf{X}^*)^{\mathbf{X}}$ *as*

$$\psi(f) = T(f)(f|_{x_1}, \ldots, f|_{x_k})$$

where $f|_{x_k}$ *is the restriction of* f *in* x_k *as defined in 3.6. Then* ψ *is an isomorphism of groups.*

Proof.

- Is ψ an homomorphism? Let $f, g \in \mathcal{AUT}_{tree}(\mathbf{X}^*)$ be in $\mathbf{X}^*$. Then:

$$\psi(f)\psi(g) = T(f)(f|_{x_1}, \ldots, f|_{x_k}) \quad T(g)(g|_{x_1}, \ldots, g|_{x_k}) = \\ = T(f)T(g)(f|_{(1T(g))}g|_{x_1}, \ldots\ f|_{(kT(g))}g|_{x_k}) = \psi(fg)$$

- Is the function injective? If $f \neq g$ we have that there exists $\mathbf{w} = y_1 \ldots y_n \in \mathbf{X}^*$ s.t. $u_1 \ldots u_n = (\mathbf{w}) \neq g(\mathbf{w}) = v_1 \ldots v_n$. If $T(f) \neq T(g)$ we have that $\psi(f) \neq \psi(g)$. Otherwise if $T(f) = T(g)$ then $u_1 = v_1$. But $f(y_1y_2 \ldots y_n) = u_1 f|_{y_1}(y_2 \ldots y_n) \neq g(y_1y_2 \ldots y_n) = v_1 g|_{y_1}(y_2 \ldots y_n)$, therefore the restrictions $g|_{y_1}$ and $f|_{y_1}$ are different and ψ is one-to-one.
- Is surjective? Let $\beta(a_1, \ldots, a_k)$ be an element of $\mathcal{S}(\mathbf{X}) \wr \mathcal{AUT}_{tree}(\mathbf{X}^*)$. Let us sign $(a_1, \ldots, a_k)$ by $(a_{x_1}, \ldots, a_{x_k})$. Given $\mathbf{w} = w_1 \ldots w_n \in \mathbf{X}^*$ with $n > 0$ we define $f(\mathbf{w}) := \beta(w_1).a_{w_1}(w_2 \ldots w_n)$ and $f(\varnothing) := \varnothing$. It's easy to verify that f is a tree-automorphism and that $\psi(f) = \beta(a_{x_1}, \ldots, a_{x_k})$.

□

The consequences of this result are very important: since for 4.4 $\mathcal{GA}(\mathbf{X})$, the set of synchronous automatic transformations on $\mathbf{X}$ can be identified with $\mathcal{AUT}_{tree}(\mathbf{X})$, the set of tree-automorphisms on $\mathbf{X}^*$, now we have that every element in $\mathcal{GA}(\mathbf{X})$ can be identified with some element $\beta(a_1, \ldots, a_k)$ in $\mathcal{S}(\mathbf{X}) \wr \mathcal{AUT}_{tree}(\mathbf{X})$ and viceversa. This leads to this result:

Proposition 6.3. *Let $\mathcal{A} =< \mathbf{X}, \mathcal{Q}, \pi, \lambda >$ be an automaton such that $|\mathcal{Q}|$, the cardinality of $\mathcal{Q}$, is n and $\mathbf{X} = \{x_1, \ldots, x_k\}$. Then the set of all the possible actions defined by $\mathcal{A}$ can be described with some n recurrent formulas*

$$\begin{aligned} f_1 &= \beta_1(h_{1,x_1}, \ldots, h_{1,x_k}) \\ f_2 &= \beta_2(h_{2,x_1}, \ldots, h_{2,x_k}) \\ &\ldots \\ f_n &= \beta_n(h_{n,x_1}, \ldots, h_{n,x_k}) \end{aligned}$$

where each h_{j,x_i} is equal to some f_j for some $j \in \{1, \ldots, n\}$, and each β_j is a permutation of the alphabet.

Proof. Each initial Automaton $\mathcal{A}_q$ with q state of $\mathcal{A}$ gives us a transformation in $\mathcal{GA}(\mathbf{X})$, therefore can be written down as one of the recursive formulas. □

Proposition 6.4. *The group $\mathcal{S}(\mathbf{X}) \wr \mathcal{AUT}_{tree}(\mathbf{X})$ acts **faithfully** on $\mathbf{X}^*$ as a set from the left by:*

$$\beta(a_{x_1}, \ldots, a_{x_k})(w_1 w_2 \ldots w_n) = \beta(w_1).a_{w_1}(w_2 \ldots w_n)$$

Proof. The group $\mathcal{S}(\mathbf{X}) \wr \mathcal{AUT}_{tree}(\mathbf{X}^*)$ is isomorphic to $\mathcal{GA}(\mathbf{X}) = \mathcal{AUT}_{tree}(\mathbf{X}^*)$. The group $\mathcal{AUT}_{tree}(\mathbf{X}^*)$ acts faithfully from the left on $\mathbf{X}^*$ because it is a subgroup of $\mathcal{S}(\mathbf{X}^*)$. Thus it is easy to verify that $\mathcal{S}(\mathbf{X}) \wr \mathcal{AUT}_{tree}(\mathbf{X}^*)$ acts by the left faithfully as stated through $\psi^{-1}(\beta(a_{x_1}, \ldots, a_{x_k}) \quad (w_1 w_2 \ldots w_n)$. □

Part 4. The Classification Theorem

In this part we present a result of 1999 which describe every possible group generated by a 2-state-automaton on a 2-letter-alphabet.

7. Introduction

Among the object we will see there are some which require a special attention, and I'm talking about the Infinite Dihedral Group, the Lamplighter Group and the Adding Machine.

7.1. Infinite Dihedral Group.

We have seen the the *finite* case of the Dihedral Group $\mathbb{Z}_2 \ltimes_\varphi \mathbb{Z}_n$(5.16), given as the simmetry group of the regular polygon of n sides. How do we generalise? One of the possible ways is to send $n \longrightarrow +\infty$.

Definition 7.1. We call by **Infinite Dihedral Group**, and we sign it by $\mathcal{D}_\infty$, the group $\mathbb{Z}_2 \ltimes_\varphi \mathbb{Z}$.

In the finite case $\mathbb{Z}_2 \ltimes_\varphi \mathbb{Z}_n$, the element $(0, k)$ was telling us to turn of $\frac{k\pi}{n}$ radiants the poligon A, so now, with $n \longrightarrow +\infty$ we may wonder how to interpret geometrically the transformation described by the same element. We need to find an object identifiable with $\mathbb{Z}$. We can help ourselfs thinking at it as the infinite line of integers.

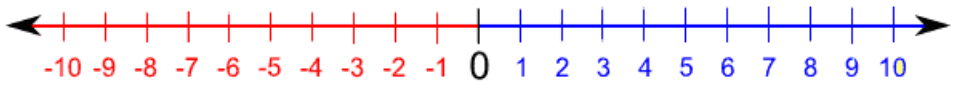


FIGURE 8. The line of integers (image taken from https://www.math-only-math.com/images/integers-numbers-on-number-line.png)

Then we can describe the action of the element $(0, k)$ on this figure as a shift on the right of k positions, and the element $(1, 0)$ as the reflection around the origin$(z \longmapsto -z)$. Therefore our Infinite Dihedral Group is the group of symmetries of $\mathbb{Z}$, that we represent as a line.

7.2. **Lamplighter Group.** According to the first reference to this algebraic object was in in 1983 and remained unnoticed for a while. I believe it would have been a perfect protagonist for a short short story of Borges, the notorious argentinian author who wrote many narratives on mathematical objects(I strongly advise his writings and the Wikipedia page) . Unfortunately Borges died in 1986.

Definition 7.2. The **Lamplighter Group** $\mathcal{L}$ is the *Restricted* Wreath Product $\mathbb{Z} \wr \mathbb{Z}_2 = \mathbb{Z} \ltimes \mathbb{Z}_2^{(\mathbb{Z})}$.

So the elements of $\mathcal{L}$ are of the form $(z, (h_i)_{i\in\mathbb{Z}})$ with $z \in \mathbb{Z}$ and $h_i \in \mathbb{Z}_2$, and just a finite amount of h_i different from $0_{\mathbb{Z}_2}$. Each $(z, (h_i)_{i\in\mathbb{Z}})$ can be imagined as an infinite dark road $(\mathbb{Z})$, with lampions every 10 meter (h_i), and just a finite amount of them turned on (the indexes i for which $h_i \neq \mathbb{Z}_2$). And in the specific position z, near some lampion, we can see a man, the "lamplighter".

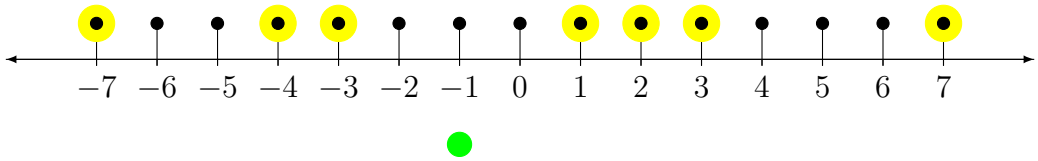


FIGURE 9. A possible representation of an **element** $(-1, (h_i)_{i\in\mathbb{Z}})$ in $\mathcal{L}$. The green circle represent the coordinate -1 of the lamplighter, while the yellow circles represent the lit on lampions at positions $\{-7, -4, -3, 1, 2, 3, 7\}$, i.e. the position i for which $h_i \neq 0$.

The product of two elements is:

$$(z_2, (h_i)_{i\in\mathbb{Z}}) * (z_1, (k_i)_{i\in\mathbb{Z}}) = (z_1 + z_2, (h_{i+z_1} +_{\mathbb{Z}_2} k_i)_{i\in\mathbb{Z}})$$

And, given an element $(z, (h_i)_{i\in\mathbb{Z}})$, its inverse is $(-z, (h_{i-z})_{i\in\mathbb{Z}_2})$.

7.3. **The Adding Machine.**

Definition 7.3. Let $\mathbf{X}$ be $\{0,1\}$. We call the **Adding Machine** the synchronous automatic transformation $f = \tau(id_{\mathcal{GA}(\mathbf{X})}, f) : \mathbf{X}^* \longrightarrow \mathbf{X}^*$, where τ is the transposition of $\mathcal{S}(\mathbf{X})$.

Why is it called adding machine? It depends on the way it acts on $\mathbf{X}^n$.

$$f(0y_2 \dots y_n) = \tau(0).id_{\mathcal{GA}(\mathbf{X})}(y_2 \dots y_n) = 1y_2 \dots y_n \tag{3}$$

$$f(1y_2 \dots y_n) = id(1).ab^{-1}(y_2 \dots y_n) = 1.ab^{-1}(y_2 \dots y_n) \tag{4}$$

Let us identify each sequence $y_1 \dots y_k \dots y_n$ with the number

$$t = y_1 + y_2 2 + \dots + y_k 2^{k-1} + \dots + y_n 2^{n-1} \quad \in \mathbb{Z}/2^n\mathbb{Z} = \mathbb{Z}_{2^n}$$

This means identifying $\mathbf{X}^n$ with $\mathbb{Z}_{2^n}$. We can so extend the action of f on $\mathbb{Z}_{2^n}$.

The equations above tells us that if $x_1 \dots x_k \dots x_n = \mathbf{w_1} x_k \mathbf{w_2} = \mathbf{w_1} 0 \mathbf{w_2}$ is a sequence, where $\mathbf{w_1}$ is a sequence of 1s, while at position k there's the **first element** $x_k = 0$, then $f(\mathbf{w_1} x_k \mathbf{w_2}) = f(\mathbf{w_1} 0 \mathbf{w_2}) = \mathbf{v_1} 1 \mathbf{w_2}$, where $\mathbf{v_1}$ is a sequence of 0s. Let us see:

$$\begin{aligned} & f\ (t \in \mathbb{Z}_{2^n}) = \\ &= f\ (x_1 + x_2 2 + \dots + x_{k-1} 2^{k-2} + x_k 2^{k-1} + x_{k+1} 2^k \dots + x_n 2^{n-1}) = \\ &= f\ (1 + 1 \cdot 2 + \dots + 1 \cdot 2^{k-2} + 0 \cdot 2^{k-1} + x_{k+1} 2^k \dots + x_n 2^{n-1}) = \\ &= 0 + 0 \cdot 2 + \dots + 0 \cdot 2^{k-2} + 1 \cdot 2^{k-1} + x_{k+1} 2^k \dots + x_n 2^{n-1} = t + 1 \in \mathbb{Z}_{2^n} \end{aligned}$$

The mistery is explained: f acts on $\mathbb{Z}_{2^n}$ by addign 1 to each number. Another thing on which we might interrogate ourselves is, what is $< f >$? We might need to watch to a particular property of f to understand this.

Definition 7.4. A left action of G on X is said to be transitive if, for each $x, y \in X$, there exists an element $g \in G$ such that $gx = y$.

Definition 7.5. A synchronous transformation $s : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ is called **spherically transitive** if $< s >$, the group generated by s, acts transitively on $\mathbf{X}^n$ for each n.

Proposition 7.6. *If a synchronous transformation $s : \mathbf{X}^* \longrightarrow \mathbf{X}^*$ is spherically transitive, then $< s >$ is infinite.*

Proof. Fix n. Let $\mathbf{w}$ be an element of $\mathbf{X}^n$. Then, for each $\mathbf{v} \in \mathbf{X}^n$ there exists $g_{\mathbf{v}} \in G$ such that $g_{\mathbf{v}} \mathbf{w} = \mathbf{v}$. This yields to $|G| \geqslant n$ for each n, so G is infinite. □

If we prove that f is spherically transitive we have that $< f >$ is infinite and cyclic, so that is isomorphic to $\mathbb{Z}$.

The transformation f acts on $\mathbf{X}^n$ as it does on $\mathbb{Z}_{2^n}$:

$$f(t) = t + 1$$

Consequently:

$$f^m(t) = t + m \quad \text{for every m} \in \mathbb{N}$$

And so f is spherically transitive, so infinite, so cyclic, so $< f >$ is isomorphic to $\mathbb{Z}$.

8. The Theorem

8.1. **Ouverture.**

Theorem 8.1. *Let $\mathcal{A} =< \mathbf{X}, \mathcal{Q}, \pi, \lambda >$ be an automaton. Let $\mathbf{X} = \{0, 1\}$ be the 2-letter-alphabet and let $|\mathcal{Q}| = 2$. Then the group generated by $\mathcal{A}$ is isomorphic to one of the following groups:*

1. *The trivial group $\{1\}$,*
2. *The 2nd order group $(\mathbb{Z}_2, +)$,*
3. *The direct sum $\mathbb{Z}_2 \oplus \mathbb{Z}_2$,*
4. *The infinite cyclic group $\mathbb{Z}$,*
5. *The Infinite Dihedral Group $\mathcal{D}_\infty$,*
6. *The Lamplighter Group $\mathcal{L}$.*

The first three structures are quite simple, and have many analogies with the fact that we are dealing **2** states on an alphabet with **2** letters. But I would like the reader to appreciate for a moment the beauty of this theorem, which shows us that also such complex groups as the last three can arise from such simple model of machines.

9. Proof

We will skip the demonstration regarding the case of isomorphism to the Lamplighter Group and the isomorphism with the Infinite Dihedral Group.

9.1. **Define the cases.** To prove it we need to examine case by case the group described by each possible Automaton $\mathcal{A}$ in this set. This means first to see in how many ways can we define π and λ.

(π) Graphically, from each state exit two possible arrows, and each can arrive to one of the two states. Algebraically, the function has domain $\mathbf{X} \times \mathcal{Q}$ and codomain $\mathcal{Q}$, so for its definition there are $|\mathcal{Q}|^{|\mathbf{X}\times\mathcal{Q}|} = 2^{2\dot{2}} = 16$ possibilities.

(λ) We can define the output function by its restrictions $\lambda(\cdot, q)$. For each $q \in \mathcal{Q}$ the function $\lambda(\cdot, q) : \mathbf{X} \longrightarrow \mathbf{X}$, must be a *permutation* of the alphabet $\mathbf{X}$. Since $\mathbf{X} = \{0, 1\}$, identifiable with $\mathbb{Z}_2$, there are only two possible permutation, the inversion σ, which exchanges the two symbols, and the identity id, which leaves them unchanged. So there are 2 possibilities for $\lambda(\cdot, q)$ and there 2 states q in $\mathcal{Q}$. This means there are $2\dot{2} = 4$ possible definitions for λ.

Overall this means $16 \cdot 4 = 64$ possible definition of $\mathcal{A}$.

In a more formal and more operative way:
Recursive definition. Let $\{q, s\}$ be the states of the automaton $\mathcal{A}$. Consequently the group is generated by $a = \mathcal{A}_q$ and $b = \mathcal{A}_s$. As we have seen in 6.3, we can define $\mathcal{A}$ by recursive formulas:

$$\begin{aligned} a &= \sigma^{i_1}(x_{11}, x_{12}) \\ b &= \sigma^{i_2}(x_{21}, x_{22}) \end{aligned} \tag{5}$$

Where $\sigma^{i_1}, \sigma^{i_2}$ are elements of $\mathcal{S}_2 := \mathcal{S}(\mathbf{X}) = \mathcal{S}(\{0, 1\})$ and $x_{ij} \in \{a, b\}$ (notice that we have two possibilities for each variable of the equation, so $2^6 = 64$ cases as seen before). We imagine that $i_1, i_2 \in \{0, 1\}$ and $\sigma^0 := id_{\mathcal{S}(2)}$ (the identity function), while $\sigma^1 = \tau$ is the other element of $\mathcal{S}_2$, the inversion.

We will proceed this way: first we define $a = \bar{a}$, and then we will analyse which group arises for each one of the 8 possible definition of b with $a = \bar{a}$. We will continue till we see every possible case of a and b.

Remark 16. Remember that $\mathcal{GA}(\mathbf{X})$ acts **faithfully** on $\mathbf{X}^*$, therefore two elements c, d of $\mathcal{GA}(\mathbf{X})$ acts in the way same on $\mathbf{X}^*$ iff $c = d$.

Convention 10. *Pay attention: from now on id will stand by the identity permutation for some set, usually $\mathcal{GA}(\mathbf{X})$ or $\mathcal{S}(\mathbf{X})$ and not always we will specify which one, it will be understandable by the context.*

9.2. **Trivial case.** If $\sigma^{i_1} = \sigma^{i_2} = id_{\mathcal{S}_2} = id$, then both a and b acts on the word of $\mathbf{X}$ as the identity, so $a = b = id_{\mathcal{GA}(\mathbf{X})}$, so we easily obtain the trivial group $\{1\}$. So we no loger need to consider the 16 cases:

$$a = id(x_{11}, x_{12}) = id$$
$$b = id(x_{21}, x_{22}) = id$$

Let's so consider from now on that $\sigma^{i_1} := \tau$, the transposition of $\mathcal{S}_2$. The cases with $\sigma^{i_1} := id$ are symmetrical.

9.3. **Cases a=t(a,a).** If instead we have $a = \tau(a|_0 = a, a|_1 = a)$, then:

$$\begin{aligned} a^2 &= \tau(\ a|_0 = a\ ,\ a|_1 = a\ ) * \tau(\ a|_0 = a\ ,\ a|_1 = a\ ) = \\ &= \tau\tau(a|_{0\tau}a|_0, a|_{1\tau}a_1) = \\ &= id(a^2, a^2) \end{aligned}$$

Therefore $a^2 = id(a^2, a^2) = id_{\mathcal{GA}(\mathbf{X})}$. This means a acts on $\mathbf{X}^*$ changing each word to its opposite, and has order 2. Now let's see b.

(1.1) If $b = id(b, b)$, $b = a^2$ acts on $\mathbf{X}^*$ as the identity, therefore $< a, b >$ is isomorphic to $(\mathbb{Z}_2, +)$ by $a \longmapsto 1$ and $b \longmapsto 0$.

(1.2) If $b = \tau(a, a)$, then $b = a$, so $< a, b >$ is again isomorphic to $\mathbb{Z}_2$.

(1.3) If $b = \tau(b, b)$, again $b = a$ and so $< a, b >$ is isomorphic to $\mathbb{Z}_2$.

(1.4) If $b = id(a, a)$, then b acts on $\mathbf{X}^*$ by changing each letter but the first one, so $b^2 = id_{\mathcal{GA}(\mathbf{X})}$. Besides ab acts by changing just the first letter, and the same does ba, so remembering 16, $ba = ab$. We can see so that $< a, b >= \{a^2, a, b, ab\}$ is isomorphic to $(\mathbb{Z}_2 \oplus \mathbb{Z}_2, +)$ by

$$\begin{aligned} a^2 = b^2 &\longmapsto (0, 0) \\ a &\longmapsto (1, 1) \\ b &\longmapsto (0, 1) \\ ab = ba &\longmapsto (1, 0) \end{aligned}$$

(1.5) If $b = \tau(b|_0 = a, b|_1 = b) = \tau(a, b)$ then

$$ba^{-1} = ba = \tau(b|_0, b|_1)\tau(a|_0, a|_1) = id(b|_1 a|_0, b|_0 a|_1) = id(ba, aa) = \\ = id(ba, aa)$$

Therefore ba acts by leaving unchanged the first letter, and by proposing again either itself, either $a^2 = id_{\mathcal{S}(\mathcal{GA}(\mathbf{X}))} = id$. So ba leaves each word unchanged, $ba = id$. So $b = a^{-1} = a$, because a has order 2. So again we have an isomorphism to $\mathbb{Z}_2$.

(1.6) Having $b = id(a,b)$ we obtain the Infinite Dihedral Group. The proof is omitted.

(1.7)-(1.8) Given $c = \sigma(c|_0, c|_1)$, with $\sigma \in \mathcal{S}_2$, we have that the conjugation $a^{-1}ca = aca = \tau(a,a)\sigma(c|_0, c|_1)\tau(a,a) = \sigma(ac|_1 a, ac|_0 a)$. Besides, $< a,b >=< a, a^{-1}ba >$ (given a and b we can generate $a^{-1}ba$, and given a and $a^{-1}ba$ we can generate b). Consequently the case (7) $< a,b = \tau(b,a) >$ is analogous to the case (5) $< a, a^{-1}ba >=< a, d = \tau(a,d) >$, while the case (7) $< a,b = id(b,a) >$ is analogous to the case $< a, a^{-1}ba >=< a, d = id(a,d) >$.

9.4. **Cases a=t(b,a).** Let a be $a = \tau(a|_0 = b, a|_1 = a) = \tau(b,a)$. The cases in which $a = \tau(a|_0 = a, a|_1 = b)$ are symmetrical to this one.

(2.1) If $b = \tau(b|_0 = b, b|_1 = a) = \tau(b,a)$ then $a = b$, therefore $a = \tau(a,a) = b = \tau(a,a)$ and so $< a,b >$ is isomorphic to $\mathbb{Z}_2$.

(2.2) If $b = \tau(b|_0 = a, b|_1 = b) = \tau(a,b)$ then:

$$\begin{aligned} ba^{-1} &= \tau(b|_0, b|_1)\,\tau(a|_1^{-1}, a|_0^{-1}) = \tau\tau(b|_1 a|_1^{-1}, b|_0 a|_0^{-1}) = \\ &= id(ba^{-1}, ab^{-1}) \\ ab^{-1} &= \tau(a|_0, a|_1)\tau(b|_1^{-1}, b|_0^{-1}) = id(a|_1 b|_1^{-1}, a|_0 b|_0^{-1}) = \\ &= id(ab^{-1}, ba^{-1}) \end{aligned}$$

This yields, that if $c := ba^{-1}$ and $d := ab^{-1}$, then:

$$c = id(c,d)d = id(d,c)$$

So $c = d = id_{\mathcal{GA}(\mathbf{X})}$, because they both leave each word unchanged. This gives us the equality $id_{\mathcal{GA}(\mathbf{X})} = c = ba^{-1}$ which leads to $a = b = \tau(a,b) = \tau(b,b)$, and consequently to $a^2 = id(a^2, a^2) = is_{\mathcal{GA}(\mathbf{X})}$. So $< a,b >=< a >= \{id_{\mathcal{GA}(\mathbf{X})}, a\}$, which is isomorphic to $\mathbb{Z}_2$.

(2.3) If $b = \tau(b,b)$, then by calling $b' := a$ and $a' := b$, we get $a' = \tau(a',a')$ and $b' = \tau(a',b')$ and we see again the case (1.5), so isomorphism with $\mathbb{Z}_2$.

(2.4) If $b = \tau(a,a)$ we see:

$$\begin{aligned} ba^{-1} &= \tau(a,a)\tau(a^{-1}, b^{-1}) = id(id, ab^{-1}) \\ ab^{-1} &= \tau(b,a)\tau(a^{-1}, a^{-1}) = id(id, ba^{-1}) \end{aligned}$$

Then we see that defining $c := ba^{-1}$ and $d := ab^{-1}$, we get the same conclusion as in case (2.2), isomorphism with $\mathbb{Z}_2$.

(2.5) If $b = id(b,b)$, then $b = id_{\mathcal{GA}(\mathbf{X})}$, and $a = \tau(id_{\mathcal{GA}(\mathbf{X})}, a)$. Here a is the adding machine. Therefore $< a,b >=< a >$ is isomorphic to $\mathbb{Z}$.

(2.6) If $b = id(a,a)$, the group $G :=< a,b >$ is isomorphic to $\mathbb{Z}$. To arrive to this result we shall prove that G is cyclic. We omit the proof that its cardinality is infinite. Then G, being infinite and cyclic, is isomorphic to $\mathbb{Z}$. Let us see this:

$$\begin{aligned} ba &= id(a,a)\tau(b,a) = \tau(ab, a^2) \\ ab &= \tau(b,a)id(a,a) = \tau(ba, a^2) \end{aligned}$$

Which yields to $ba = ab$, which tells us that $< a,b >$ is abelian. Besides

$$ba^2 = ba\,a = \tau(ba, a^2)\tau(b,a) = id(a^2 b, a^2 b)$$

Consequently $ba^2 = 1$. We claim that $G :=< a, b >=< ab >$. In fact ab generates b by $(ab)^2 = abab = b(ba^2) = b$, and ab and b generate a by $abb^{-1} = a$. Therefoe G is cyclic generated by ab.

(2.7) If $b = id(b, a)$ then $< a, b >= \mathcal{L}$ is the Lamplighter Group $\mathcal{L}$, but we are going to skip the demonstration.

(2.8) Given $b = id(a, b)$ we can reach the symmetric case of the (2.7). Let us take $b^{-1} = id(a^{-1}, b^{-1})$, $a^{-1} = \tau(a^{-1}, b^{-1})$. In general, since $a^{-1}, b^{-1} \in < a, b >$, and consequently $a, b \in < a^{-1}, b^{-1} >$, we have that $< a, b >=< a^{-1}, b^{-1} >$. So we can observe the group generated by a^{-1}, b^{-1}. Let us now take a generic element $d = \tau(d, d) \in \mathcal{GA}(\mathbf{X})$. Then:

$$(b^{-1})^d = d^{-1}b^{-1}d = \tau(d^{-1}, d^{-1})\, id(a^{-1}, b^{-1})\, \tau(d, d) = id((b^{-1})^d, (a^{-1})^d)$$
$$(a^{-1})^d = d^{-1}a^{-1}d = \tau(d^{-1}, d^{-1})\, \tau(a^{-1}, b^{-1})\, \tau(d, d) = \tau((b^{-1})^d, (a^{-1})^d)$$

Why did we do this? Let us call $b' := b^{-1}$ and $a' = a^{-1}$. We showed that we can observe directly the group generated by a', b'. Let us take the generic element $x_1x_2 \dots x_k$ with $x_i \in \{a', b'\}$. We observe that its conjugate by d, $(x_1x_2 \dots x_k)^d$, is the same as $(x_1)^d(x_2)^d \dots (x_k)^d$. This tells us that the each element in $< a'^d, b'^d >$ is conjugate to some element of $< a', b' >$ and viceversa. So the conjugate of the group $< a', b' >$ is $< a'^d, b'^d >$, therefore they are isomorphic. So again, with another jump, we can call $b'' := (b^{-1})^d = b'^d$ and $a'' := (a^{-1})^d = a'^d$ and watch at $< a'', b'' >$ that is isomorphic to $< a, b >$. For the equations above we have that:

$$a'' = \tau(b'', a'')$$
$$b'' = id(b'', a'')$$

That is the symmetrical of the case (2.7).

9.5. **Cases a=t(b,b).** Let $a = \tau(b, b)$.

(3.1)-(3.2) The case $b = \tau(a, b)$ is analogous to (2.4), while the case $b = \tau(b, a)$ is symmetrical to (2.4), both leading to $\mathbb{Z}_2$.

(3.3)-(3.4) If $b = \tau(b, b)$ then $b = a = \tau(a, a)$, and we have the case (1.2) with $\mathbb{Z}_2$. If $b = \tau(a, a)$ we arrive to the same conclusion.

(3.5) If $b = (b, b)$ then $b = id$ and $a = \tau(id, id)$, so $< a >= \{id, a\}$ is isomorphic to $\mathbb{Z}_2$.

(3.6)-(3.7) This cases lead to the Infinite Dihedral Group. The proof is omitted.

(3.8) If $b = id(a, a)$, then:

$$a^2 = (b^2, b^2)$$
$$b^2 = (a^2, a^2)$$
$$ba = \tau(ab, ab)$$
$$ab = \tau(ba, ba)$$

This yields $a^2 = b^2 = id$ and to $ab = ba = \tau(ab, ab)$ (abelian group). For this reason we can see each possible word $x_1x_2 \dots x_k$ with $x_i \in \{a, b\}$ as $a^n b^m$ where $n+m = k$. In addition we know that $a^n = a^{n \,(mod\, 2)}$ and $b^m = b^{m \,(mod\, 2)}$, so each possible composition of a and b is an element $a^i b^j$, where $i, j \in \{0, 1\}$.

So the group $< a, b >= \{id, a, b, ab\}$ is isomorphic to $\mathbb{Z}_2 \oplus \mathbb{Z}_2$ by:

$$id \longmapsto (0, 0)$$
$$a \longmapsto (1, 0)$$
$$b \longmapsto (0, 1)$$
$$ba \longmapsto (1, 1)$$

References

[1]