The Automata Zoo

Stefan Hoffmann

November 2021

1 Introduction

Welcome to the Automata Zoo! Here, I collect different types of automata, provide characterizations and pointers to their natural habitats, i.e., literature.

The goal is to give, for each type of automata, a short overview, supply examples and some characterizations.

The selection is subjective, and encompasses mostly types that I encountered during my research, or that I simply wanted to write about.

2 Notation

I assume some familiarity with basic set theory, algebra (I use terms as, for example, semigroups, monoid, groups, homomorphisms and kernels) and also automata theory, for example as contained in [8]. I assume the reader understands my drawings of automata, and what I mean when I say certain patterns (often only given graphically) are forbidden in an automaton. So, the purpose the present section is to fix notations, but not to give an introduction to the mentioned concepts themselves.

A *semi-automaton* is a triple $\mathcal{A} = (Q, \Sigma, \delta)$ where Q is the *finite state set*, Σ the *input alphabet* and $\delta \subseteq Q \times \Sigma \times Q$ the *transition relation*. An *automaton* is a semi-automaton together with a designated start state and a set of final states, formally a quintuple $\mathcal{A} = (Q, \Sigma, \delta, s_0, F)$ where (Q, Σ, δ) is a semi-automaton and $s_0 \in Q$ is the *start (or initial) state* and $F \subseteq Q$ is the *set of final state*.

We define a relation $\hat{\delta} \subseteq Q \times \Sigma^* \times Q$ to be the smallest relation such that $(q, \varepsilon, q') \in \hat{\delta}$ and $(q, u, q') \in \hat{\delta}$ for $u = u_1 u_2 \cdots u_n$ with $u_i \in \Sigma$ if there exist $q_0, q_1, \ldots, q_n \in Q$ with $q_0 = q$, $q_n = q'$ and $(q_{i-1}, u_i, q_i) \in \delta$ for $i \in \{1, 2, \ldots, n\}$.

We have $\delta \subseteq \hat{\delta}$ and this extension of δ will also be denoted by δ in the following. In fact, by associating every letter $a \in \Sigma$ with the relation $\{(q, q') \mid (q, a, q') \in \delta\}$ we have a map from Σ to the set of all binary homogeneous relations over Q and this extension results by the application of the universal property of the free monoid Σ^* to this map.

With an automaton $\mathcal{A}$, we associated the language *accepted (or recognized)* by $\mathcal{A}$ as $L(\mathcal{A}) = \{u \in \Sigma^* \mid \exists q \in F : (q_0, u, q) \in \delta\}$, i.e., the set of labels of paths from the start state to a final state. If the context is clear, we also call semi-automata simply automata. Also notions defined for semi-automata are also valid for automata.

An automaton (or semi-automaton) $\mathcal{A}$ is

deterministic if for every $q \in Q$ and $a \in \Sigma$ there exists *at most one* $q' \in Q$ such that $(q, a, q') \in \delta$,

complete if for every $q \in Q$ and $a \in \Sigma$ there exists *at least one* $q' \in Q$ such that $(q, a, q') \in \delta$.

For a deterministic semi-automaton, we identify δ with a (partial) function $\delta : Q \times \Sigma \to Q$, and its extension (also denoted by δ) with a partial function from $Q \times \Sigma^*$ to Q.

For a deterministic and complete $\mathcal{A}$ and $u \in \Sigma^*$, let $f_u : Q \to Q$ be the function given by $f_u(q) = \delta(q, u)$. With every deterministic and complete semi-automaton $\mathcal{A}$, we associate the

transformation semigroup $\mathcal{TS}_{\mathcal{A}} = \{f_u \mid u \in \Sigma^+\}$,

transformation monoid $\mathcal{TM}_{\mathcal{A}} = \{f_u \mid u \in \Sigma^*\}$,

with function composition as the semigroup (monoid) operation.

Note that every word from Σ^+ (or Σ^*) corresponds to an element of the transformation semigroup (or monoid). We will make use of this identification of words with transformation of states for a given automaton without further mentioning it.

For more information on automata, see [8].

For a given set X, by $\mathcal{T}_X$ we denote the *transformation monoid on* X, i.e., the set of all mappings $X \to X$ with function composition as operation. Sometimes, we also call $\mathcal{T}_X$ the *transformation semigroup on* X. Transformation semigroups (monoids) on a set of states Q of an automaton are subsemigroups (submonoids) of $\mathcal{T}_Q$.

Note that automata, as defined here, are always finite.

3 Green's Relations

Semigroups (or monoids) enter into automata theory most obviously by the transformation semigroup (or monoid) of a given semi-automaton. Hence, it is natural to investigate automata by investigating their transformation semigroups (and in fact, as we can associate a syntactic monoid to every language, semigroup notions enter into language theory as well, in particular the beautiful Eilenberg-Schützenberger [3] correspondence between pseudovarieties of semigroups and varieties of languages is worth to mention here; see [13] for more information on that).

Green's relations [4] are a well-known notion from semigroup theory. Later we will present various classes of automata defined with the help of these relations in the transformation semigroup. Here, we will only introduce these relations and state some properties that are of relevance to us, in particular their characterization in the semigroup of all transformations $\mathcal{T}_X$ on a set. For more information, see [2].

Let S be a semigroups. If S contains an identity element, we set $S^1 = S$, otherwise let $S^1 = S \cup \{1\}$ with $1 \notin S$ and $1x = x1 = x$ for all $x \in S$, i.e, the semigroup adjoined with an identity element. Now, letd $s, t \in S$. We set:

- $s \,\mathcal{L}\, t$ if and only if $S^1 s = S^1 t$,
- $s \,\mathcal{R}\, t$ if and only if $sS^1 = tS^1$,
- $s \,\mathcal{J}\, t$ if and only if $S^1 s S^1 = S^1 t S^1$,
- $s \,\mathcal{D}\, t$ if and only if there exists $r \in S$ such that $s \,\mathcal{L}\, r$ and $r \,\mathcal{R}\, t$,
- $s \,\mathcal{H}\, t$ if and only if $s \,\mathcal{L}\, t$ and $s \,\mathcal{R}\, t$.

It is obvious that these are equivalence relations, and furthermore that $\mathcal{L}$ is a right-congruence and $\mathcal{R}$ is a left-congruence. Together with the next result, Theorem 3.1, we get the inclusion relations shown in Figure 1.

Theorem 3.1. *Let S be a semigroup and $s, t \in S$. If $s \,\mathcal{D}\, t$, then $s \,\mathcal{J}\, t$.*

Proof. By assumption there exists r such that $s \,\mathcal{L}\, r$ and $r \,\mathcal{R}\, t$. Hence, we find $x, y \in S^1$ such that $xs = r$ and $ry = t$. So $xsy = t$, which implies $t \in S^1 s S^1$. Similary, we can show $s \in S^1 t S^1$, which yields $s \,\mathcal{J}\, t$. $\square$

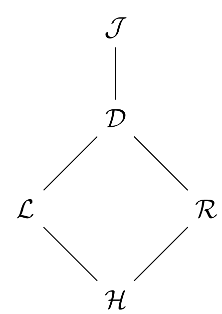


Figure 1: Hasse diagram for the inclusion relations between Green's relations on an arbitrary semigroup.

We will visualize Green's relations for a given semigroup by drawing a so called *eggbox diagram* [2]. In such a diagram, each $\mathcal{D}$-class is represented by a box, an each such box is subdivided into columns and rows. Each row represented an $\mathcal{R}$-class, and each column represents an $\mathcal{L}$-class. The elements are arranged such that those elements that are contained in the same column and the same row constitute a single $\mathcal{H}$-class. Furthermore, idempotents are marked by an asterix. See Figure 2 for an example.

$*111$	$*222$	$*333$

$*123$	231
312	132
321	213

$*122$ 211	$*133$ 311	233 322
212 $*121$	313 131	$*323$ 232
221 112	331 $*113$	332 $*223$

Figure 2: The eggbox diagram for $\mathcal{T}_{\{1,2,3\}}$ encompassing three $\mathcal{D}$-classes. A transformation on $\{1,2,3\}$ is written as a string, where the appearance of number j at position i means the transformation maps i to j, for example 111 is the mapping that maps everything to 1.

As we are concerned with finite state automata here, the following result, stating that the $\mathcal{D}$ and $\mathcal{J}$ equivalences coincide for finite semigroups, is of particular interest to us.

Theorem 3.2. *If S is a finite semigroup, then for $s,t \in S$ we have $s\,\mathcal{D}\,t$ if and only if $s\,\mathcal{J}\,t$.*

Proof. Let $s,t \in S$. By Theorem 3.1, we only have to show that $s\,\mathcal{J}\,t$ implies $s\,\mathcal{D}\,t$. Suppose $x,x',y,y' \in S^1$ such that $xsy = t$ and $x'ty' = s$. Set $r = x't$. Then $ry' = s$. By finiteness, there exist $i,j \geq 0$ and $p,q > 0$ such that $r(y'y)^{i+p} = r(y'y)^i$ and $(xx')^{j+q}t = (xx')^j t$. Multiplying the first equation by $(x'x)^i$ from the left gives $r(y'y)^p = r$, or $sy'(yy')^{p-1}y = r$. Hence $s\,\mathcal{R}\,r$. Similarly, we find $x(xx')^{q-1}r = (xx')^q t = t$. So $r\,\mathcal{L}\,t$. $\square$

The following result about the $\mathcal{H}$-classes is note-worthy.

Theorem 3.3 (Green's theorem)**.** *Let H be an $\mathcal{H}$-class of a semigroup S. Then, either $H^2 \cap H = \emptyset$ or $H^2 = H$ and H is a subsemigroup that forms a group*[1].

Let X be a set. Next, we characterize the relations $\mathcal{R}, \mathcal{L}$ and $\mathcal{D}$ for the monoid $\mathcal{T}_X$.

Theorem 3.4. *Let $f,g \in \mathcal{T}_X$. Then in $\mathcal{T}_X$ the following holds true:*

1. *$f\,\mathcal{L}\,g$ iff $f(X) = g(X)$,*
2. *$f\,\mathcal{R}\,g$ iff $\ker(f) = \ker(g)$,*
3. *$f\,\mathcal{D}\,g$ iff $|f(X)| = |g(X)|$.*

Proof. 1. First, suppose there exist $h,h' : X \to X$ such that, for all $x \in X$, we have $f(h(x)) = g(x)$ and $g(h'(x)) = f(x)$. The first equation implies $g(X) \subseteq f(X)$, the latter the other inclusion. Hence $f(X) = g(X)$.

[1] I avoid to say that H is a subgroup, as this notion is usually related to an overgroup, and S is not assumed to be a group here or even to contain an identity element (and the identity in H is in general only an idempotent of S).

Conversely, if $f(X) = g(X)$, then, for each $x \in X$, define a function $h : X \to X$ by selecting some $h(x) \in f^{-1}(g(x))$. Then, $f(h(x)) = g(x)$ for each $x \in X$. Similarly, we can find $h' : X \to X$ such that $f(x) = g(h'(x))$ for $x \in X$.

2. Suppose there exists $h : X \to X$ such that $h(f(x)) = g(x)$ for each $x \in X$. Hence, if $f(x) = f(y)$, then $g(x) = h(f(x)) = h(f(y)) = g(y)$ and so $\ker(f) \subseteq \ker(g)$. The other inclusion follows similarly.

   If $\ker(f) \subseteq \ker(g)$, define $h : X \to X$ by setting $h(f(x)) = g(x)$ for $x \in X$ and $h(x)$ arbitrary for $x \notin f(X)$. As $f(x) = f(y)$ implies $g(x) = g(y)$, this is in fact a well-defined function. Similarly, we find a function $h' : X \to X$ such that $h'(g(x)) = f(x)$ for all $x \in X$.

3. By definition and the previous items, $f \mathcal{D} g$ iff there exists $h : X \to X$ such that $\ker(f) = \ker(h)$ and $h(X) = g(X)$. As $|X/\ker(f)| = |f(X)|$ and $|X/\ker(h)| = |h(X)|$ (this is the first homomorphism theorem in the category of sets applied to f and g separately), we find $|g(X)| = |h(X)| = |X/\ker(f)| = |f(X)|$.

   Conversely, suppose $|f(X)| = |g(X)|$ and let $\varphi : f(X) \to g(X)$ be a bijective map. Define $h : X \to X$ by $h(x) = \varphi(f(x))$. Then $h(x) = h(y)$ iff $f(x) = f(y)$ and so $\ker(h) = \ker(f)$. Furthermore, $h(X) = \varphi(f(X)) = g(X)$. Hence, $f \mathcal{D} f$. □

Remark 3.5. *Note that for a subsemigroup S of $\mathcal{T}_X$ and $f, g \in S$, in general, for example the condition $|f(X)| = |g(X)|$ does not imply that $f \mathcal{J} g$ in S.*

4 Permutation Automata

A semi-automaton $\mathcal{A} = (Q, \Sigma, \delta)$ is a *permutation automaton*, if the maps (written as f_a in Section 2) $q \mapsto \delta(q, a)$, for each $a \in \Sigma$, are permutations, i.e., bijective mappings. Equivalently, the transformation monoid of $\mathcal{A}$ forms a group. As the equations $ax = b$ and $xa = b$ are uniquely solvable for elements a, b in a group G, we have $G \times G = \mathcal{H} = \mathcal{R} = \mathcal{L} = \mathcal{J} = \mathcal{D}$. Conversely, if $\mathcal{H} = S \times S$ for a semigroup S, then, by Theorem 3.3 (Green's theorem), S is a group (in fact, it is well-known and easy to see that a semigroup S forms a group if and only if the equations $ax = b$ and $xa = b$ are solvable for arbitrary elements $a, b \in S$, which essentially says that all elements are $\mathcal{R}$-equivalent and $\mathcal{L}$-equivalent and so $\mathcal{H}$-equivalent, which does not use Theorem 3.3).

So, an automaton is a permutation automaton iff in its transformation semigroup, all elements are $\mathcal{H}$-equivalent.

Observe that, by definition, a permutation automaton is complete and deterministic.

Almost immediately by the definition, a complete and deterministic automaton is a permutation automaton, if it does not contain the pattern shown in Figure 3.

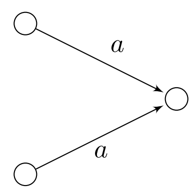


Figure 3: The forbidden pattern for permutation automata.

In a certain sense, a permutation automata can be seen as a finite permutation group (a subgroup of the set of all permutations on a set), and permutation groups were investigated since the work of Galois, Lagrange, Cauchy or Jordan, see [11] for a historical account of the genesis of permutation group theory. However, the focus was different, and in the context of automata theory, permutation automata were introduced, as it appears to be, by McNaugthon [10] in connection with the star-height problem and by Thierrin [15].

The languages recognized by permutation automata are called (pure-)group languages [10, 12]. The languages recognized by permutation automata whose transformation monoid (as a group) falls within the following classes were investigated: commutative groups [12], solvable groups [3, 14] and supersoluble groups [1].

Furthermore, three recent studies investigated the descriptional complexity (more precisely the state complexity) of severals operations on permutation automata [7], the projection operation [6] and the commutative closure operation [5].

5 Simple Transformation Semigroup

A semigroup S is *simple*, if all elements are $\mathcal{J}$-equivalent. Here, we investigate complete and deterministic automata whose transformation semigroup is simple. The main result here is a forbidden-pattern characterization, see Figure 6 and Theorem 5.7.

Remark 5.1. *We will use Theorem 3.4 frequently here. By this result, if $\mathcal{A}$ has a simple transformation semigroup, we can deduce that $|\delta(Q, u)| = |\delta(Q, v)|$ for all $u, v \in \Sigma^+$.*

Let $\mathcal{A} = (Q, \Sigma, \delta)$ and set $S = \delta(Q, a)$ for some $a \in \Sigma$. If every letter $b \in \Sigma$ permutes S, i.e., $\delta(S, b) = S$, and $S = \delta(Q, b)$, then $\mathcal{T}_{\mathcal{A}}$ is simple (in fact, in this case all elements are $\mathcal{L}$-equivalent). See Figure 5 for an example of such an automaton. However, there are other examples as well, see Figure 4.

Obviously, every permutation automaton has a simple transformation semigroup. The next gives a sufficient condition for the converse, more specifically, if at least one letter permutes the states, then all letters do.

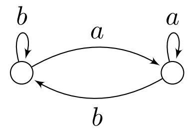
Proposition 5.2. *Let $\mathcal{A} = (Q, \Sigma, \delta)$ be a complete deterministic semi-automaton*

Element	Image of						Element	Image of					
	1	2	3	4	5	6		1	2	3	4	5	6
a^2b	2	3	4	5	2	5	ba	2	4	3	2	5	4
b^2	2	4	5	2	3	4	a	2	5	4	3	2	3
ab	3	2	5	4	3	4	ab^2a	3	4	5	2	3	2
b^3	3	5	2	3	4	5	bab	3	5	4	3	2	5
b^4	4	2	3	4	5	2	bab^2	4	2	5	4	3	2
ab^2	4	3	2	5	4	5	aba	4	5	2	3	4	3
a^2	5	2	3	4	5	4	b^2a	5	3	2	5	4	3
b	5	3	4	5	2	3	ab^3	5	4	3	2	5	2

Table 1: The elements of the transformation semigroup of the automaton from Figure 5. The image of the i-th state is written in the i-th column for each element ($i \in \{1, 2, 3, 4, 4, 5, 6\}$) for the left and right part of the table.

$\cdot$	a	a^2	b	b^2	b^3	b^4	ab	ab^2	ab^3	a^2b	ab^2a	aba	ba	bab	bab^2	b^2a
a	a^2	a	ab	ab^2	ab^3	a	a^2b	ab^2a	aba	ab	ab^2	ab^3	aba	a^2	a^2b	ab^2a
a^2	a	a^2	a^2b	ab^2a	aba	a^2	ab	ab^2a	aba	ab	ab^2a	aba	a^2ba	a	ab	ab^2
b	ba	b	b^2	b^3	b^4	b	bab	bab^2	b^2a	b^2	b^3	b^4	b^2a	ba	bab	bab^2
b^2	b^2a	b^2	b^3	b^4	b	b^2	ba	bab	bab^2	b^3	b^4	b	bab^2	b^2a	ba	bab
b^3	bab^2	b^3	b^4	b	b^2	b^3	b^2a	ba	bab	b^4	b	b^2	bab	bab^2	b^2a	ba
b^4	bab	b^4	b	b^2	b^3	b^4	bab^2	b^2a	ba	b	b^2	b^3	ba	bab	bab^2	b^2a
ab	aba	ab	ab^2	ab^3	a	ab	a^2	a^2b	ab^2a	ab	aba	a	ab^2a	ab^3	a	a^2b
ab^2	a^2	ab^2	ab^3	a	ab	ab^2	aba	a^2	a^2b	ab^3	a	ab	a^2b	ab^2a	aba	a^2
ab^3	a^2b	ab^3	a	ab	ab^2	ab^3	ab^2a	aba	a^2	aba	a	ab^2	a^2	a^2b	ab^2a	aba
a^2b	ab^3	a^2b	ab^2a	aba	a^2	a^2b	a	ab	ab^3	a^2b	aba	a^2	ab^2	ab^3	a	ab
ab^2a	ab^2	ab^2a	aba	aba	a^2b	ab^2a	ab^3	a	ab	aba	a^2	a^2b	ab	ab^2	ab^3	a
aba	ab	aba	a	ab	ab^2	ab^3	ab	ab^2	a^3	a^2	a^2b	a^2b	a	ab	ab^2	ab^3
ba	b	ba	bab	bab^2	b^2a	a^2b	bab	b^3	b^4	bab	bab^2	b^2	b^4	b	b^2	bab^2
bab	b^4	bab	bab^2	b^2a	ba	$baba$	b	b^2	b^3	bab^2	b^2a	ba	b^3	b^4	b	b^2
bab^2	b^3	bab^2	b^2a	ba	bab	bab^2	b^4	b	b^2	b^2a	ba	bab	b^2	b^3	b^4	b
b^2a	b^2	b^2a	ba	bab	bab^2	b^2a	b^3	b^4	b	ba	bab	bab^2	b	b^2	b^2	aba

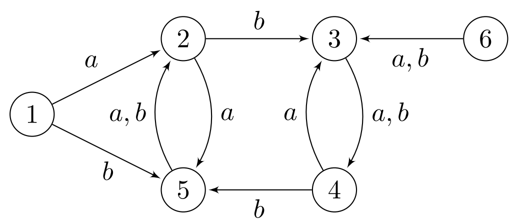
Table 2: The Cayley (or multiplication) table for the transformation semigroup of the automaton from Figure 5.



$\cdot$	a	b
a	a	b
b	a	b

$*\,a$
$*\,b$

Figure 4: Example of an automaton whose transformation semigroup is simple. On the right the eggbox-diagram is drawn, we have two $\mathcal{R}$-classes and a single $\mathcal{L}$-class.



a^2b a ab ab^2a ab^2 aba $*\,a^2$ ab^3
ba b^2 b^3 bab $*\,baba$ bab^2 b^2a b

Figure 5: Example of an automaton whose transformation semigroup is simple. On the right the eggbox-diagram is drawn, here, all elements are $\mathcal{L}$-equivalent.

with a simple transformation semigroup. Suppose there exists $a \in \Sigma$ such that $q \mapsto \delta(q,a)$, $q \in Q$, is bijective, then $\mathcal{A}$ is a permutation semi-automaton.

Proof. We have $\delta(Q,a) = Q$. If $b \in \Sigma$, then by Theorem 3.4 applied to the transformations induced by ab and a in $\mathcal{T}_\mathcal{A}$, we find $|\delta(Q,ab)| = |Q|$. As $\delta(Q,ab) = \delta(Q,b) \subseteq Q$, we have $\delta(Q,b) = Q$. $\square$

Corollary 5.3. *Let $\mathcal{A} = (Q,\Sigma,\delta)$ be a complete deterministic semi-automaton with a simple transformation semigroup. Then $\mathcal{A}$ is a permutation automaton if and only if the transformation semigroup contains an identity element.*

Proof. If $\mathcal{A}$ is a permutation automaton, then there obviously exists $u \in \Sigma^+$ acting like the identity on the states. Conversely, if $u = u_1 u_2 \cdots u_n$ with $u_i \in \Sigma$ and $\delta(q,u) = q$ for each $q \in Q$, then we must have $\delta(Q,u_1) = Q$ and by the previous result $\mathcal{A}$ is a permutation automaton. $\square$

By the previous results, we can deduce that for transformation *monoids*, simplicity is the same as saying the automaton is a permutation monoid (or the monoid a group). Hence, this concept only makes sense for semigroups that are not monoids.

Proposition 5.4. *Let $\mathcal{A} = (Q, \Sigma, \delta)$ be a complete and deterministic semi-automaton. Then the transformation* monoid *of $\mathcal{A}$ is simple if and only if $\mathcal{A}$ is a permutation automaton.*

Proof. If the transformation monoid only contains the identity transformation, the claim holds true. So, suppose f is a non-identity element of it. Then there exist g, h in the transformation monoid such that $g(f(h(q))) = q$ for all $q \in Q$. As f is a non-identity element, it is induced by a non-empty word $u \in \Sigma^+$, and similarly the concatenation of g, f and h is represented by some non-empty word. So, the identity element is contained in the transformation semigroup and by the previous claim, the result follows. □

In more abstract semi-theoretical language, this can be reformulated the following way.

Proposition 5.5. *A finite monoid M is simple if and only if it is a group.*

Proof. A group is certainly a simple monoid. Now suppose M is a simple and finite monoid. By finiteness, there exists a finite generating set $m_1, m_2, \ldots, m_n$ for M. Let $\mathcal{A} = (Q, \Sigma, \delta)$ be the automaton with $Q = M$, $\Sigma = \{m_1, m_2, \ldots, m_n\}$ and $\delta(m, m_i) = mm_i$ (this is essentially the right Cayley graph for the generating set). Then M is isomorphic to the transformation monoid and the transformation monoid is simple. By Proposition 5.4, $\mathcal{A}$ is a permutation automaton, which is equivalent to the fact that the transformation monoid is a permutation group. Hence, M is a group. □

By the next result, every letter permutes its image.

Proposition 5.6. *Let $\mathcal{A} = (Q, \Sigma, \delta)$ be a complete and deterministic semi-automaton with a simple transformation semigroup. Then, every letter permutes its image, i.e., if $S = \delta(Q, a)$ for $a \in \Sigma$, then $\delta(S, a) = S$.*

Proof. We have $\delta(S, a) = \delta(Q, aa) \subseteq \delta(Q, a) = S$. So, as with Theorem 3.4, $|\delta(Q, aa)| = |S|$, we find $\delta(S, a) = S$. □

By the previous result, if $a \in \Sigma$ and $t \notin \delta(Q, a)$, then there exists $s \in \delta(Q, a)$ such that $\delta(t, a) = \delta(s, a)$.

Next, we will show that the transformation semigroup of a given deterministic and complete semi-automaton is simple if and only if it does not contain the pattern from Figure 6. This characterization is originally from [9, Lemma 15], but we give here a different proof with the help of Theorem 3.4.

Theorem 5.7. *Let $\mathcal{A} = (Q, \Sigma, \delta)$ be a complete deterministic semi-automaton. The transformation semigroup of $\mathcal{A}$ is simple iff there does not exist $p, q \in Q$ and letters $a, b \in \Sigma$ such that $\delta(p, a) \neq \delta(q, a)$ and $\delta(p, ab) = \delta(q, ab)$.*

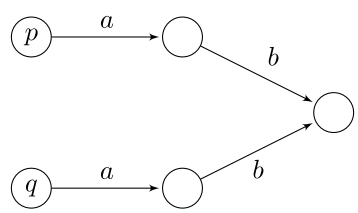


Figure 6: The forbidden pattern for automata whose transformation semigroup is simple. See Theorem 5.7.

Proof. If the transformation monoid of $\mathcal{A}$ is simple, by Theorem 3.2 and 3.4, $|\delta(Q,a)| = |\delta(Q,ab)|$ for all $a,b \in \Sigma$. In particular, the map $q \mapsto \delta(q,b)$ is injective for $q \in \delta(Q,a)$, which gives the claim.

Conversely, if the condition holds true, this says precisely that, for all $a \in \Sigma$ and $b \in \Sigma$, the map $q \mapsto \delta(q,b)$ is injective on $\delta(Q,a)$, which gives $|\delta(Q,a)| = |\delta(Q,ab)|$ for all $a,b \in \Sigma$. As the concatenation of injective maps is injective, for every $u \in \Sigma^+$ and $a \in \Sigma$, we have $|\delta(Q,au)| = |\delta(Q,a)|$. Furthermore, if $v = v_1 v_2 \cdots v_n$ with $v_i \in \Sigma$ and $n > 0$, we have $\delta(Q,v) = \delta(\delta(Q,v_1 \cdots v_{n-1}), v_n) \subseteq \delta(Q,v_n)$. Combining with the previous sentence, for every $u,v \in \Sigma^+$, we have that $|\delta(Q,uv)| = |\delta(Q,v)|$. In particular, combining these facts, $|\delta(Q,v)| = |\delta(Q,v_1)| \leq |\delta(Q,v_n)|$. So, for every $a,b \in \Sigma$, by considering $v = ab$, we have $|\delta(Q,a)| \leq |\delta(Q,b)|$. As $a,b \in \Sigma$ were chosen arbitrary, this can only hold if $|\delta(Q,a)| = |\delta(Q,b)|$ for all $a,b \in \Sigma$ (otherwise, fix two letters for which this is not the case and apply the assumption with them swapped). This gives, as before, for each $u,v \in \Sigma^+$, that $|\delta(Q,u)| = |\delta(Q,v)|$. $\square$

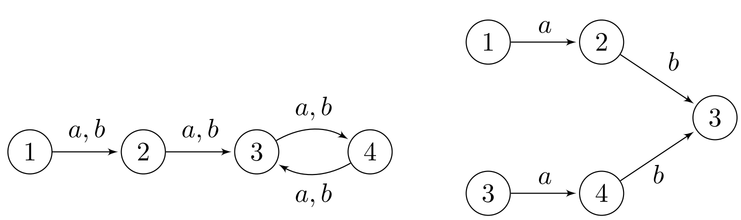


Figure 7: A non-simple automaton on the left that contains the forbidden pattern drawn on the right for comparison.

References

[1] Olivier Carton, Jean-Éric Pin, and Xaro Soler-Escrivà. Languages recognized by finite supersoluble groups. *J. Autom. Lang. Comb.*, 14(2):149–161, 2009.

[2] Alfred Hoblitzelle Clifford and Gordon Bamford Preston. *The Algebraic Theory of Semigroups*, volume 1. American Mathematical Society, 1961.

[3] S. Eilenberg. *Automata, Languages, and Machines, Volume B*. Academic Press, Inc., Orlando, FL, USA, 1976.

[4] James Alexander Green. On the structure of semigroups. *Annals of Mathematics (second series)*, 54(1):163–172, 1951.

[5] Stefan Hoffmann. State complexity bounds for the commutative closure of group languages. In Galina Jirásková and Giovanni Pighizzini, editors, *Descriptional Complexity of Formal Systems - 22nd International Conference, DCFS 2020, Vienna, Austria, August 24-26, 2020, Proceedings*, volume 12442 of *Lecture Notes in Computer Science*, pages 64–77. Springer, 2020.

[6] Stefan Hoffmann. State complexity of projection on languages recognized by permutation automata and commuting letters. In Nelma Moreira and Rogério Reis, editors, *Developments in Language Theory - 25th International Conference, DLT 2021, Porto, Portugal, August 16-20, 2021, Proceedings*, volume 12811 of *Lecture Notes in Computer Science*, pages 192–203. Springer, 2021.

[7] Michal Hospodár and Peter Mlynárčik. Operations on permutation automata. In Natasa Jonoska and Dmytro Savchuk, editors, *DLT 2020, Tampa, FL, USA, May 11-15, 2020, Proceedings*, volume 12086 of *Lecture Notes in Computer Science*, pages 122–136. Springer, 2020.

[8] Dexter Kozen. *Automata and computability.* Undergraduate texts in computer science. Springer, 1997.

[9] Michal Kunc. Regular solutions of language inequalities and well quasi-orders. *Theor. Comput. Sci.*, 348(2-3):277–293, 2005.

[10] R. McNaughton. The loop complexity of pure-group events. *Information and Control*, 11(1/2):167–176, 1967.

[11] Peter M. Neumann. *The Mathematical Writings of Évariste Galois*. Heritage of European Mathematics. European Mathematical Society, 2011.

[12] Jean-Éric Pin. Syntactic semigroups. In Grzegorz Rozenberg and Arto Salomaa, editors, *Handbook of Formal Languages, Volume 1*, pages 679–746. Springer, 1997.

[13] Jean-Éric Pin. *Mathematical Foundations of Automata Theory*. Ebook, Version from December 2020. `https://www.irif.fr/~jep/PDF/MPRI/MPRI.pdf`.

[14] Denis Thérien. Languages of nilpotent and solvable groups (extended abstract). In Hermann A. Maurer, editor, *Automata, Languages and Programming, 6th Colloq., Graz, Austria, July 16-20, 1979, Proceedings*, volume 71 of *LNCS*, pages 616–632. Springer, 1979.

[15] Gabriel Thierrin. Permutation automata. *Mathematical systems theory*, 2(1):83–90, 1968.