Languages, automata and computation II
Tutorial 4

Winter semester 2023/2024

In this tutorial we study automata theory in sets with atoms.

Sets with atoms

In this tutorial we explore sets with atoms. Fix a countable set A and consider the relational structure $\mathbb{A} = (A, =)$ over the signature consisting only of equality. Elements of $\mathbb{A}$ are called *atoms* and automorphisms α of this structure are the bijections of A. Automorphisms are extended to sets homomorphically, e.g., $\alpha\,(x, y) = (\alpha x, \alpha y)$.

A *set with atoms* is any set built from set constructors and atoms from $\mathbb{A}$. An automorphism α *fixes* a set with atoms x if $\alpha x = x$. Fix a tuple $\bar{a} \in A^n$. An $\bar{a}$*-automorphism* is an automorphism fixing $\bar{a}$. A set with atoms x is *supported by* $\bar{a}$ if it is fixed by every $\bar{a}$-automorphism, and it is *equivariant* if it is supported by the empty tuple $()$. A set with atoms is *finitely supported* if it is supported by some tuple of atoms, and it is *legal* if it is finitely supported and all its elements are legal (this is a recursive definition).

The *orbit* of element x is the set $\mathsf{orbit}(x) = \{\alpha x \mid \text{automorphism } \alpha\}$ of elements which can be obtained by applying some automorphism to x. Thus an equivariant set x is a union of orbits (of its elements), and an equivariant set is *orbit finite* if this union is finite. The set of orbits of a set x is $\mathsf{Orbits}(x) = \{\mathsf{orbit}(y) \mid y \in x\}$ (this is a partition of x). For instance, for given $a, b \in \mathbb{A}$ with $a \neq b$, we have $\mathsf{orbit}(a, a) = \left\{(c, d) \in \mathbb{A}^2 \mid c = d\right\}$ and $\mathsf{orbit}(a, b) = \left\{(c, d) \in \mathbb{A}^2 \mid c \neq d\right\}$. Since there are no more orbits in $\mathbb{A}^2$, we have $\mathsf{Orbits}(\mathbb{A}^2) = \{\mathsf{orbit}(a, a), \mathsf{orbit}(a, b)\}$ and $\mathbb{A}^2$ is the union of two orbits.

Exercise 1. Fix the equality atoms $(\mathbb{A}, =)$. For each of the following sets with atoms decide whether they are 1) legal, 2) equivariant, 3) orbit finite.

1. $\mathbb{A}^2$.
2. $\mathbb{A}^* := \mathbb{A}^0 \cup \mathbb{A}^1 \cup \mathbb{A}^2 \cup \cdots$.
3. $\mathbb{A}^\omega := \{a_1 a_2 \cdots \mid a_1, a_2, \cdots \in \mathbb{A}\}$.
4. $2^{\mathbb{A}} := \{B \mid B \subseteq \mathbb{A}\}$ (powerset).
5. $2^{\mathbb{A}}_{\text{fin}} := \{B \mid B \subseteq \mathbb{A}, B \text{ finite}\}$ (finite powerset).
6. $2^{\mathbb{A}}_{\text{fs}} := \{B \mid B \subseteq \mathbb{A}, B \text{ finitely supported}\}$ (finitely supported powerset).

7. $2^{\mathbb{A}}_{\text{eq}} := \{B \mid B \subseteq \mathbb{A}, B \text{ equivariant}\}$ (equivariant powerset).

Solution:

1. $\mathbb{A}^2$ is legal, equivariant, and it has two orbits.
2. $\mathbb{A}^*$ is legal, equivariant, and it has infinitely many orbits, since words of different length cannot be in the same orbit.
3. $\mathbb{A}^\omega$ is equivariant and it has infinitely many orbits. It is not legal since it has elements without a finite support such as $w = a_1 a_2 \cdots$ where all the $a_i \in \mathbb{A}$ are pairwise distinct.
4. The unrestricted powerset is illegal, equivariant, and orbit-infinite.
5. The finite powerset is legal, equivariant, and orbit-infinite, since subsets of different sizes cannot be in the same orbit.
6. The finitely supported powerset is legal, equivariant, and orbit-infinite.
7. The equivariant powerset is legal, equivariant, and contains just two elements: $\emptyset$ and $\mathbb{A}$.

□

Exercise 2. An atom structure $\mathbb{A}$ is called *oligomorphic* if $\mathbb{A}^n$ is orbit-finite for every $n \in \mathbb{N}$. Are the following atom structures oligomorphic?

1. $(\mathbb{N}, =)$.
2. $(\mathbb{Z}, \leq)$.
3. $(\mathbb{Q}, \leq)$.
4. $(\mathbb{Q}, +)$.

Solution:

1. Yes.
2. No, since already $\mathbb{Z}^2$ has infinitely many orbits. Automorphisms of this structure are integer translations $\alpha(x) = x + k$, $k \in \mathbb{Z}$. Consequently, all pairs in the orbit of $(x, y) \in \mathbb{Z}^2$ have the same $y - x$ value. In particular $(0, 0)$, $(0, 1)$, $(0, 2)$, $\ldots$, are all in pairwise distinct orbits.
3. Yes.
4. Automorphisms of $(\mathbb{Q}, +)$ satisfy $\alpha(0) = 0$ and $\alpha(x + y) = \alpha(x) + \alpha(y)$. Consequently, for every rational $x \in \mathbb{Q}$, if we write $x = p/q$ for integers $p, q \in \mathbb{Z}$ we have $p \cdot x = q$ and by applying α to both sides $\alpha(p \cdot x) = p \cdot \alpha(x)$ and $\alpha(q) = q \cdot \alpha(1)$. Consequently,

$$\alpha(x) = x \cdot \alpha(1).$$

Consequently α is uniquely determined by how it acts on 1, and thus the automorphisms of this structure are of the form $\alpha(x) = k \cdot x$ for some $k \in \mathbb{Q}$. It follows that applying α to a pair (x, y) preserves the ratio $\frac{y}{x}$. Thus $\mathbb{Q}^2$ has infinitely many orbits.

□

Exercise 3. Consider an orbit-finite set X and an equivariant relation $R \subseteq X \times X$. For every $n \in \mathbb{N}$, let $R_n = R^0 \cup R^1 \cup R^2 \cup \cdots \cup R^n$. Show that the following chain computing the reflexive-transitive closure of R terminates:

$$R_0 \subseteq R_1 \subseteq \cdots \subseteq X \times X.$$

Solution: Each R_n is an equivariant subset of $X \times X$: R_0 is just the identity relation, which is equivariant, and equivariant relations are closed under compostion and union. Since $X \times X$ is orbit-finite (X being orbit-finite and the equality atoms being oligomorphic), R_n is a union of finitely many orbits of $X \times X$. One can then show that there is some $n \leq |\mathsf{Orbits}(X \times X)|$ s.t. $R_n = R_{n+1} = \cdots$. $\square$

Orbit-finite automata

Fix an oligomorphic atom structure $\mathbb{A}$, which will usually consist of a countable set with equality $(\mathbb{A}, =)$. A *orbit-finite automaton* (OFA) is a tuple $A = (\Sigma, Q, I, F, \Delta)$ where Σ is a orbit-finite *input alphabet* (often $\Sigma = \mathbb{A}$), Q is a orbit-finite set of *states*, $I, F \subseteq Q$ are equivariant subsets of Q (thus orbit-finite), called *initial*, resp., *final* states, and $\Delta \subseteq Q \times \Sigma\mathbb{Q}$ is an equivariant set of *transitions* (thus orbit-finite).

Exercise 4. Consider an orbit-finite automaton with input alphabet $\hat{\Sigma} := \Sigma \times \mathbb{A}$ where Σ is finite. Consider the following *projection* mapping $\pi : \hat{\Sigma}^* \to \Sigma^*$ which forgets the data part of a word:

$$\pi : (\sigma_1, a_1) \cdots (\sigma_n, a_n) \mapsto \sigma_1 \cdots \sigma_n.$$

Show that the projection $\pi L \subseteq \Sigma^*$ of a data language $L \subseteq \hat{\Sigma}^*$ recognised by an orbit-finite automaton is a regular language.

Solution: Let $A = \left(\hat{\Sigma}, Q, I, F, \Delta\right)$ be a OFA. Build a $\mathsf{NFA} B$ whose states are orbits of Q, initial states are orbits of I, and final states are orbits of F. A transition $(p, (\sigma, a), q) \in \Delta$ of A induces a transition $\mathsf{orbit}(p) \xrightarrow{\sigma} \mathsf{orbit}(q)$ of B. One then shows that $L(B) = \pi L(A)$. $\square$

The following is a summary of (non)-closure properties of languages of finite data words recognised by OFA and its deterministic variant.

	$\cup$	$\cap$	$\_^R$	$\Sigma^* \setminus \_$
Deterministic OFA	✓	✓	×	✓
Nondeterministic OFA	✓	✓	✓	×

Exercise 5. Show a nondeterministic OFA language which is not recognised by a deterministic OFA.

Solution: Consider the language of all words $w \in \mathbb{A}^*$ s.t. the last letter appears at least twice:

$$L = \{a_1 \cdots a_n \in \mathbb{A}^* \mid \text{there is } 1 \leq i < n \text{ s.t. } a_i = a_n\}.$$

This language is OFA recognisable, in dimension one: The automaton guesses the occurrence of a_i and checks that it appears at the end of the word.

This language is not recognisable by a deterministic OFA. By way of contradiction, let A be a deterministic OFA recognising L. Build a long word of pair-wise distinct letters $w = a_1 \cdots a_n \notin L$ and look at the corresponding run of the automaton

$$I \ni q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n.$$

There is some $n \in \mathbb{N}$ s.t. some a_i is not in the support of the last state,

$$a_i \notin \mathsf{supp}\, q_n.$$

Let $b \in \mathbb{A}$ be a fresh input symbol. Since $w \cdot b \notin L$, the extended run is rejecting:

$$I \ni q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n \xrightarrow{b} q \notin F.$$

Let α be any atom automorphism fixing $\mathsf{supp}\, q_n$ s.t. $\alpha(b) = a$. In particular, $\alpha(q_n) = q_n$. The following modified run

$$I \ni q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n = \alpha(q_n) \xrightarrow{\alpha(b)} \alpha(q) \notin F.$$

shows $w \cdot a \notin L(A)$ since F is equivariant (and thus the same applies to its complement) and the automaton is deterministic. Since $w \cdot a \in L$, this contradicts $L(A) = L$. □

The previous exercise is subsumed by the following one (since determinstic OFA are closed under complement).

Exercise 6. Show that the class of nondeterministic OFA languages is not closed under complement.

Solution: Consider the language $L \subseteq \mathbb{A}^*$ containing all words where a data value appears at least twice, which is easily seen to be OFA-recognisable. By way of contradiction, assume that its complement is recognised by some OFA A. Consider a very long word $w = a_1 \cdots a_n \in \Sigma^*$ of pairwise distinct data values, and look at some accepting run of A when reading it

$$I \ni q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} \cdots \xrightarrow{a_n} q_n \in F.$$

For n sufficiently large there are indices $1 \leq i \leq j < k \leq n$ s.t. $a_i, a_k \notin \mathsf{supp}\, q_j$. There exists an atom automorphism α which fixes $\mathsf{supp}\, q_j$ (and thus $\alpha(q_j) = q_j$) s.t. $\alpha(a_i) = a_k$. The following run is also accepting

$$I \ni \alpha(q_0) \xrightarrow{\alpha(a_1)} \cdots \xrightarrow{\alpha(a_j)} \alpha(q_j) = a_i \xrightarrow{a_{j+1}} \cdots \xrightarrow{a_n} q_n \in F.$$

and thus $w' = \alpha(a_1) \cdots \alpha(a_j) a_{j+1} \cdots a_n \in L(A)$. However the data value a_k appears at least twice in w', thus $w' \in L$, which is a contradiction. □

Exercise 7. Show that the class of deterministic OFA languages is not closed under reversal.

Solution: The language "the last letter appears at least twice" from the solution of Exercise 5 cannot be recognised by a deterministic OFA, however its reversal can. □

Exercise 8. Show that the class of non-guessing OFA languages is not closed under reversal.

Solution: Consider the language L of all words $w \in \mathbb{A}^*$ s.t. "the first letter appears exactly once". This can be recognised by a determinstic OFA, which is non-guessing. Its reversal L^R contains all words where "the last letter appears exactly once". We show that it cannot be recognised without guessing. By way of contradiction let A be a non-guessing OFA recognising L^R. Consider a long word $w = a_1 \cdots a_n \in \mathbb{A}^*$ with pairwise distinct data values. Since $w \in L(A)$, there is an accepting run

$$I \ni q_0 \xrightarrow{a_1} \cdots \xrightarrow{a_{n-1}} q_{n-1} \xrightarrow{a_n} q_n \in F.$$

There is $a_i \notin \mathsf{supp}\, q_{n-1}$. Since the automaton is without guessing $\mathsf{supp}\, q_{n-1} \subseteq \{a_1, \ldots, a_{n-1}\}$, and since a_n is fresh also $a_n \notin \mathsf{supp}\, q_{n-1}$. There is an automorphism α that 1) fixes all elements in $\mathsf{supp}\, q_{n-1}$ (and in particular $\alpha(q_{n-1}) = q_{n-1}$), and 2) maps a_i to $\alpha(a_i) = a_n$. The following run is also accepting

$$I \ni \alpha(q_0) \xrightarrow{\alpha(a_1)} \cdots \xrightarrow{\alpha(a_{n-1})} \alpha(q_{n-1}) \xrightarrow{a_n} q_n \in F,$$

however it accepts a word where the last letter a_n appears twice, which is a contradiction. □

Exercise 9 (Universality is undecidable for nondeterministic OFA)**.** Consider the data alphabet $\hat{\Sigma} = \Sigma \times \mathbb{A} \cup \{\$\}$ with Σ finite and $\$ \notin \Sigma$. Consider the following data language

$$L = \{w\$w \mid w = (b_1, a_1) \cdots (b_n, a_n) \text{ and the } a_i\text{'s are pairwise distinct}\}.$$

Show that the complement $\hat{\Sigma}^* \setminus L$ of L can be recognised by

1. A nonguessing OFA of dimension two (two registers in the sense of register automata).

2. A nondeterministic OFA of dimension one, which uses guessing.

Solution: A case analysis on all kind of mistakes in the encoding yield the required automaton. □

Exercise 10 (Emptiness is PSPACE-complete for OFA)**.** Show that the emptiness problem for OFA is PSPACE complete.

Proof. For the PSPACE upper bound, we just orbitise the OFA A producing an NFA B (of exponential size) for which the emptiness question has the same answer, and then we check in NL that B is nonempty. This gives a $\mathsf{NPSPACE}$ algorithm for nonemptiness of A, and thus a PSPACE algorithm by courtesy of Savitch's theorem.

Regarding PSPACE-hardness, we reduce from emptiness of intersection of many NFA's $A_1, \ldots, A_n$, which is a PSPACE-hard problem. The idea is to construct a register automaton A without input such that a tuple of registers $\bar{r}_i$ encodes the current control location of A_i. We can assume that A_i has n states thus a tuple of $n+1$ registers $\bar{r}_i$ suffices: Automaton A_i is in state $j \in \{1, \ldots, n\}$ iff register j in $\bar{r}_i$ equals register 0 in $\bar{r}_i$. □

Exercise 11. Consider the following decision problem. In input we are given a nondeterministic OFA A and a deterministic OFA B. In output we answer yes iff $L(A) \subseteq L(B)$. Is this problem decidable? What if B is an unambiguous OFA?

Proof. Yes. We can complement B into some deterministic OFA B', and then check that $L(A) \cap L(B')$ is empty.

If B is merely unambiguous this does not work anymore (unambiguous OFA are not closed under complement, even in the space of all OFA). However we can turn A into a deterministic OFA A' and B into a unambiguous OFA B' s.t. the answer to language inclusion is the same. Now by complementing A' into A'' we can check universality of

$$L(B') \cap L(A') \cup L(A'').$$

The latter language can be recognised by a unambiguous OFA: 1) $L(A'')$ is deterministic, so unambiguous, 2) $L(B') \cap L(A')$ is unambiguous, 3) the two previous languages are disjoint, so that their union is also unambiguous. $\square$