AUTOMATA AND LOGIC

LECTURER: DR. DANIEL BORCHMANN

EXERCISES: DIPL.-MATH. FRANCESCO KRIEGEL

TIMES: 4 + 2 + 0

LECTURES: TUESDAY 6. DS
WEDNESDAY 5. DS

EXERCISES: THURSDAY 6. DS

EXCEPTIONS POSSIBLE!

ALPHABET: finite, non-empty set Σ

FORMAL LANGUAGE: $L \subseteq \Sigma^*$

CLASS OF LANGUAGES: $\mathcal{K}$ such that $\mathcal{K}_\Sigma \subseteq \mathcal{P}(\Sigma^*)$
for each alphabet Σ

CHARACTERIZATION: find property P such that

$$L \in \mathcal{K}_\Sigma \text{ iff } L \subseteq \Sigma^* \underline{\text{ and }} L \text{ satisfies } P$$

Examples: automata, grammars, ...

CLOSURE PROPERTIES: under which operations is $\mathcal{K}_\Sigma$ closed?

Examples: intersection, union, complement, homomorphic images, ...

DECIDABILITY: which problems are decidable for $L \in \mathcal{K}_\Sigma$?

Examples: $w \in L$? $L \neq \emptyset$? $L_1 \subseteq L_2$? ...

(assume: languages given by a finite characterization)

EXAMPLE CHARACTERIZATION: CHOMSKY HIERARCHY

TYPE 0

- general Chomsky grammars
- accepted by Turing machines

TYPE 1
context sensitive

- generated by context-sensitive grammars (transitions $u \to v$ where $1 \le |u| \le |v|$)
- accepted by Turing machines with a linearly bounded tape

TYPE 2
context free

- generated by context-free grammars (transitions $X \to v$ where X is a non-terminal)
- accepted by push-down automata

TYPE 3
regular

- generated by right-linear grammars (transitions $X \to uY$, $X \to u$, where X, Y are non-terminals, u terminal)
- accepted by finite automata

EXAMPLES OF CHARACTERIZATIONS

Do deterministic automata yield the same class as non-deterministic ones?

TYPE 0: YES TYPE 1: OPEN TYPE 2: NO TYPE 3: YES

EXAMPLES OF CLOSURE PROPERTIES

Is the class closed under complement?

TYPE 0: NO TYPE 1: YES TYPE 2: NO TYPE 3: YES

EXAMPLES OF DECISION PROBLEMS

CLASS	TYPE 0	TYPE 1	TYPE 2	TYPE 3
$w \in L$?	undec.	dec.	dec.	dec.
$L_1 = L_2$	undec.	undec.	undec.	dec.

Structure of the Lecture

1. Regular languages of finite words
   - algebraic characterization
     - language L ⟷ monoid M_L (syntactic monoid)
     - L regular iff M_L finite
   - logical characterization
     - logical formulas can define languages
     - there is a logic that defines the regular languages (monadic second-order logic)
   - generalized-definite languages, star-free languages
2. Languages of infinite words
   - same as 1, but with other acceptance conditions
   - closure properties, decidability of emptiness problem, connection to logic
3. Tree languages

PROPOSITION 1.3: If $\mathcal{L}_1, \mathcal{L}_2 \in Reg_\Sigma$, then $\mathcal{L}_1 \cap \mathcal{L}_2 \in Reg_\Sigma$.

Proof $\mathcal{A}_1 = (Q_1, \Sigma, I_1, \Delta_1, F_1)$, $\mathcal{A}_2 = (Q_2, \Sigma, I_2, \Delta, F_2)$ automata st $\mathcal{L}_1 = \mathcal{L}(\mathcal{A}_1)$, $\mathcal{L}_2 = \mathcal{L}(\mathcal{A}_2)$.

Define $\mathcal{A} := (Q_1 \times Q_2, \Sigma, I_1 \times I_2, \Delta, F_1 \times F_2)$, where

$$\Delta := \{((q_1, q_2), a, (q_1', q_2')) \mid (q_1, a, q_1') \in \Delta_1, (q_2, a, q_2') \in \Delta_2\}$$

Then

$$(q_1, q_2) \xrightarrow{w}_{\mathcal{A}} (q_1', q_2') \text{ iff } q_1 \xrightarrow{w}_{\mathcal{A}_1} q_1', \; q_2 \xrightarrow{w}_{\mathcal{A}_2} q_2'$$

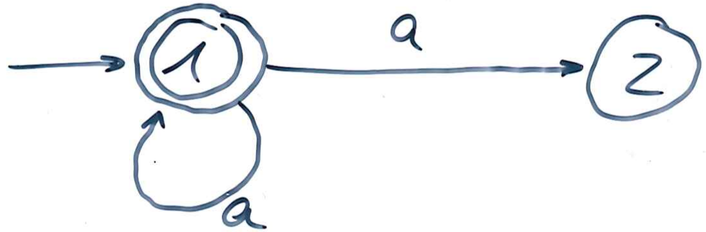
Therefore, $w \in \mathcal{L}(\mathcal{A})$ iff $w \in \mathcal{L}(\mathcal{A}_1) \cap \mathcal{L}(\mathcal{A}_2)$.

$\square$

NOTE: If $\mathcal{A} = (Q, \Sigma, I, \Delta, F)$ is non-deterministic, then $\overline{\mathcal{A}} := (Q, \Sigma, I, \Delta, Q \setminus F)$ need not satisfy $\mathcal{L}(\overline{\mathcal{A}}) = \Sigma^* \setminus \mathcal{L}(\mathcal{A})$

EXAMPLE:

$\mathcal{A}$:



$Q = \{1, 2\}$, $\Sigma = \{a\}$, $I = \{1\}$

$\Delta = \{(1, a, 1), (1, a, 2)\}$, $F = \{1\}$

$\mathcal{L}(\mathcal{A}) = a^*$, $\mathcal{L}(\overline{\mathcal{A}}) = a^+$,

$\Sigma^* \setminus \mathcal{L}(\mathcal{A}) = a^* \setminus a^* = \emptyset$

BUT: it works if $\mathcal{A}$ is deterministic

POWER SET CONSTRUCTION: $\mathcal{A} = (Q, \Sigma, I, \Delta, F)$ finite automaton

Define $\mathcal{P}(\mathcal{A}) := (\mathcal{P}(Q), \Sigma, q_0, \delta', F')$, where

- $q_0 := I$
- $\delta'(P, a) := \{ q \in Q \mid \exists p \in P : (p, a, q) \in \Delta \}$
- $F' := \{ P \subseteq Q \mid P \cap F \neq \emptyset \}$

Then

$$\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{P}(\mathcal{A}))$$

and $\mathcal{P}(\mathcal{A})$ is deterministic.

Proposition 1.5: If $\mathcal{L} \in Reg_\Sigma$, then $\overline{\mathcal{L}} := \Sigma^* \setminus \mathcal{L} \in Reg_\Sigma$.

Proof Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$ be a deterministic finite automaton st $\mathcal{L} = \mathcal{L}(\mathcal{A})$.

Then

$$w \in \mathcal{L} \text{ iff } \delta(q_0, w) \in F,$$

ie,

$$w \in \overline{\mathcal{L}} \text{ iff } \delta(q_0, w) \in Q \setminus F.$$

Therefore, the automaton

$$\overline{\mathcal{A}} := (Q, \Sigma, q_0, \delta, Q \setminus F)$$

accepts $\overline{\mathcal{L}}$.

$\square$

Minimization of Deterministic Automata

$\mathcal{A} = (Q, \Sigma, I, \delta, F)$ deterministic automaton

1. Remove unreachable states, i.e., states $q \in Q$ such that there is no $w \in \Sigma^*$ with $\delta(q_0, w) = q$.

2. Identify equivalent states: for $q \in Q$ let

$$\mathcal{A}_q := (Q, \Sigma, q, \delta, F),$$

and define

$$q \sim_{\mathcal{A}} q' \text{ iff } \mathcal{L}(\mathcal{A}_q) = \mathcal{L}(\mathcal{A}_{q'}).$$

Then $\sim_{\mathcal{A}}$ is an equivalence relation. Now identify equivalent states wrt $\sim_{\mathcal{A}}$.

NERODE RIGHT CONGRUENCE

For $\mathcal{L} \subseteq \Sigma^*$ define

$$u \rho_{\mathcal{L}} v \quad \text{iff} \quad \forall w \in \Sigma^*: (uw \in \mathcal{L} \text{ iff } vw \in \mathcal{L})$$

Then $\rho_{\mathcal{L}}$ is an equivalence relation, and in addition

$$u \rho_{\mathcal{L}} v \Rightarrow uw \rho_{\mathcal{L}} vw \qquad (w \in \Sigma^*)$$

NERODE'S THEOREM

$\mathcal{L}$ is regular iff $\rho_{\mathcal{L}}$ has finite index, ie,

$$\Sigma^*/\rho_{\mathcal{L}} := \{ [u]_{\rho_{\mathcal{L}}} \mid u \in \Sigma^* \}$$

is finite, where

$$[u]_{\rho_{\mathcal{L}}} := \{ v \in \Sigma^* \mid u \rho_{\mathcal{L}} v \}.$$

MINIMAL AUTOMATON

Define $\mathcal{A}_{\mathcal{L}} := (Q, \Sigma, q_0, \delta, F)$, where

- $Q := \Sigma^* / \rho_{\mathcal{L}}$ (finite if $\mathcal{L}$ is regular)
- $q_0 := [\varepsilon]_{\rho_{\mathcal{L}}}$
- $\delta([u]_{\rho_{\mathcal{L}}}, a) := [ua]_{\rho_{\mathcal{L}}}$ (!)
- $F := \{ [u]_{\rho_{\mathcal{L}}} \mid u \in \mathcal{L} \}$ (!)

If $\mathcal{L}$ is regular, then $\mathcal{A}_{\mathcal{L}}$ is the minimal deterministic automaton for $\mathcal{L}$.