



Automata and Logic

Aart Middeldorp and Johannes Niederhauser

Outline

1. Introduction

Organisation AI Policy Contents

2. Basic Definitions

3. Deterministic Finite Automata

4. Intermezzo

5. Closure Properties

6. Further Reading

Initial Remarks

- ▶ **Automata and Logic** is elective module 1 in master program Computer Science
- ▶ master students must select 3 out of 6 elective modules:
 - ① Automata and Logic
 - ② Constraint Solving (offered in 27S)
 - ③ Cryptography (offered in 27S)
 - ④ High-Performance Computing
 - ⑤ Optimisation and Numerical Computation
 - ⑥ Signal Processing and Algorithmic Geometry
- ▶ other master modules with theory content (**Logic and Learning** specialization):
 - ▶ Program and Resource Analysis (WM 8)
 - ▶ Computability Theory (WM 9)
 - ▶ Advanced Functional Programming (WM 20)
 - ▶ **The Lambda-Calculus, Typed and Untyped** (WM 20)



with session ID **3017 2066** for anonymous questions



Important Information

- ▶ LVA 703302 (VO 2) + 703303 (PS 2)
- ▶ <https://cl-teaching.uibk.ac.at/26w/al>
- ▶ online registration for VO required
- ▶ OLAT links for VO and PS

- ▶ LVA 703302 (VO 2) + 703303 (PS 2)
- ▶ <https://cl-teaching.uibk.ac.at/26w/al>
- ▶ online registration for VO required
- ▶ OLAT links for VO and PS

Time and Place				
VO	Monday	8:30–10:00	HSB 9	(AM)
PS	Friday	8:30–10:00	SR 12	(JN)

VO	Monday	8:30–10:00	HSB 9	(AM)
PS	Friday	8:30–10:00	SR 12	(JN)

Consultation Hours			
Aart Middeldorp	3M07	Monday	11:30–13:00
Johannes Niederhauser	3M03	Wednesday	10:00–11:30

Aart Middeldorp	3M07	Monday	11:30–13:00
Johannes Niederhauser	3M03	Wednesday	10:00–11:30

Schedule			
week 1	05.10 & 09.10	week 7	30.11 & 04.12
week 2	12.10 & 23.10	week 8	07.12 & 11.12
week 3	19.10 & 30.10 & 06.11	week 9	14.12 & 18.12 & 08.01
week 4	09.11 & 13.11	week 10	11.01 & 15.01
week 5	16.11 & 20.11	week 11	18.01 & 22.01
week 6	23.11 & 27.11	week 12	25.01

week 1	05.10 & 09.10	week 7	30.11 & 04.12
week 2	12.10 & 23.10	week 8	07.12 & 11.12
week 3	19.10 & 30.10 & 06.11	week 9	14.12 & 18.12 & 08.01
week 4	09.11 & 13.11	week 10	11.01 & 15.01
week 5	16.11 & 20.11	week 11	18.01 & 22.01
week 6	23.11 & 27.11	week 12	25.01

Grading — Vorlesung

- ▶ first exam on January 25
- ▶ registration starts 5 weeks and ends 2 weeks before exam
- ▶ de-registration is possible until 10:00 on January 22
- ▶ second exam on February 22; third exam on May 21 (on demand)

- ▶ first exam on January 25
- ▶ registration starts 5 weeks and ends 2 weeks before exam
- ▶ de-registration is possible until 10:00 on January 22
- ▶ second exam on February 22; third exam on May 21 (on demand)

Grading — PS

$\text{score} = \min\left(\frac{2}{3}(E + P) + B, 100\right)$

- E : points for solved **exercises** (at most 130)
- P : points for **presentations** of solutions (at most 20)
- B : points for **bonus presentations**

grade : $[0, 50) \rightarrow \mathbf{5}$ $[50, 63) \rightarrow \mathbf{4}$ $[63, 75) \rightarrow \mathbf{3}$ $[75, 88) \rightarrow \mathbf{2}$ $[88, 100] \rightarrow \mathbf{1}$

- ▶ homework exercises are given on course web site
- ▶ solved exercises must be marked and solutions must be uploaded (**PDF**) in **OLAT**
- ▶ strict deadline: 7 am on Friday
- ▶ 10 points per PS
- ▶ two presentations of solutions are mandatory
- ▶ 20 points for two presentations; additional presentations give bonus points
- ▶ attendance is compulsory; unexcused absence is allowed twice (resulting in 0 points)

score = $\min(\frac{2}{3}(E + P) + B, 100)$

- E : points for solved **exercises** (at most 130)
- P : points for **presentations** of solutions (at most 20)
- B : points for **bonus presentations**

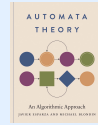
grade : $[0, 50) \rightarrow \mathbf{5}$ $[50, 63) \rightarrow \mathbf{4}$ $[63, 75) \rightarrow \mathbf{3}$ $[75, 88) \rightarrow \mathbf{2}$ $[88, 100] \rightarrow \mathbf{1}$

- ▶ homework exercises are given on course web site
- ▶ solved exercises must be marked and solutions must be uploaded (PDF) in OLAT
- ▶ strict deadline: 7 am on Friday
- ▶ 10 points per PS
- ▶ two presentations of solutions are mandatory
- ▶ 20 points for two presentations; additional presentations give bonus points
- ▶ attendance is compulsory; unexcused absence is allowed twice (resulting in 0 points)

evaluation 25W

Literature

- ▶ Dexter C Kozen
Automata and Computability
Springer–Verlag, 1997
- ▶ Javier Esparza and Michael Blondin
Automata Theory: An Algorithmic Approach
MIT Press, 2023
- ▶ Christel Baier and Joost–Pieter Katoen
Principles of Model Checking
MIT Press, 2008

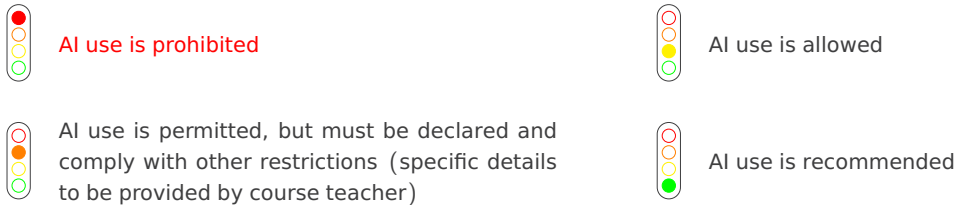


Online Material

- ▶ solutions to selected exercises are available after they have been discussed in PS

AI Policy — Automata and Logic (PS)

- ▶ students are responsible for all content they deliver (spoken, in writing, or code)
- ▶ **traffic light** system recognizes that policies differ between courses; its purpose is to provide students with certainty
- ▶ teachers have authority to redefine any aspect of policy and its enforcement
- ▶ failure to comply with policy will result in failing course



Outline

1. Introduction

Organisation AI Policy Contents

2. Basic Definitions

3. Deterministic Finite Automata

4. Intermezzo

5. Closure Properties

6. Further Reading

Outline

1. Introduction

Organisation AI Policy Contents

2. Basic Definitions

3. Deterministic Finite Automata

4. Intermezzo

5. Closure Properties

6. Further Reading

Automata

- ▶ (**deterministic**, nondeterministic, alternating) **finite automata**
- ▶ regular expressions
- ▶ (alternating) Büchi automata

Logic

- ▶ (weak) monadic second-order logic
- ▶ Presburger arithmetic
- ▶ linear-time temporal logic

Outline

1. Introduction
2. **Basic Definitions**
3. Deterministic Finite Automata
4. Intermezzo
5. Closure Properties
6. Further Reading

Definitions

- ▶ **alphabet** is finite set; its elements are called **symbols** or **letters**
- ▶ **string** over alphabet Σ is finite sequence of elements of Σ
- ▶ **length** $|x|$ of string x is number of symbols in x
- ▶ **empty string** is unique string of length 0 and denoted by ϵ
- ▶ Σ^* is set of all strings over Σ ($\emptyset^* = \{\epsilon\}$)
- ▶ **language** over Σ is subset of Σ^*

Examples

strings over $\Sigma = \{0, 1\}$: 0 0110 ϵ

languages over Σ :

- ▶ $\{\epsilon, 0, 1, 00, 01, 10, 11\}$ (all strings having at most two symbols)
- ▶ $\{x \mid x \text{ is valid program in some machine language}\}$

Definitions

- ▶ **string concatenation** $x, y \in \Sigma^* \implies xy \in \Sigma^*$ is associative:
 $(xy)z = x(yz)$ for all $x, y, z \in \Sigma^*$

- ▶ empty string is **identity** for concatenation:

$$\epsilon x = x\epsilon = x \quad \text{for all } x \in \Sigma^*$$

- ▶ x is **substring** (**prefix**, **suffix**) of y if $y = uxv$ ($y = xv$, $y = ux$)

- ▶ x^n ($x \in \Sigma^*$, $n \in \mathbb{N}$):

$$\begin{aligned} x^0 &= \epsilon \\ x^{n+1} &= x^n x \end{aligned}$$

- ▶ $\#a(x)$ ($a \in \Sigma$, $x \in \Sigma^*$) denotes number of a 's in x

Definitions

for $A, B \subseteq \Sigma^*$

- ▶ **union** $A \cup B = \{x \mid x \in A \text{ or } x \in B\}$
- ▶ **intersection** $A \cap B = \{x \mid x \in A \text{ and } x \in B\}$
- ▶ **complement** $\sim A = \Sigma^* - A = \{x \in \Sigma^* \mid x \notin A\}$
- ▶ **set concatenation** $AB = \{xy \mid x \in A \text{ and } y \in B\}$
- ▶ **powers** A^n ($n \in \mathbb{N}$) $A^0 = \{\epsilon\}$ $A^{n+1} = AA^n$
- ▶ **asterate** A^* is union of all finite powers of A

$$A^* = \bigcup_{n \geq 0} A^n = \{x_1 x_2 \cdots x_n \mid n \geq 0 \text{ and } x_i \in A \text{ for all } 1 \leq i \leq n\}$$

$$\text{▶ } A^+ = AA^* = \bigcup_{n \geq 1} A^n$$

$$\text{▶ } \text{power set} \quad 2^A = \{Q \mid Q \subseteq A\}$$

Examples

- ▶ substrings of 011: 0, 1, 01, 11, 011, ϵ
- ▶ prefixes of 011: 0, 01, 011, ϵ
- ▶ suffixes of 011: 1, 11, 011, ϵ
- ▶ $(011)^3 = 011011011 \neq 011^3$
- ▶ $\#1(011011011) = 6$ $\#0(\epsilon) = 0$
- ▶ $\{0, 10, 111\}\{1, 11\} = \{01, 101, 1111, 011, 1011, 11111\}$
- ▶ $\{0, 01, 111\}\{1, 11\} = \{01, 011, 1111, 0111, 11111\}$
- ▶ $\{1, 01\}^3 = \{111, 0111, 1011, 01011, 1101, 01101, 10101, 010101\}$
- ▶ $\{1, 01\}^* = \{\epsilon, 1, 01, 11, 011, 101, 0101, 111, 0111, 1011, 01011, \dots\}$
- ▶ $2^{\{1, 01\}} = \{\emptyset, \{1\}, \{01\}, \{1, 01\}\}$

Some Useful Properties

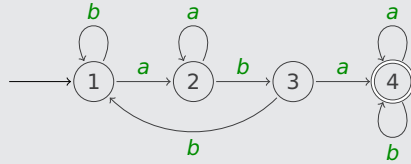
- ▶ $\{\epsilon\}A = A\{\epsilon\} = A$
- ▶ $\emptyset A = A\emptyset = \emptyset$
- ▶ $\sim(A \cup B) = (\sim A) \cap (\sim B)$
- ▶ $\sim(A \cap B) = (\sim A) \cup (\sim B)$
- ▶ $A^{m+n} = A^m A^n$
- ▶ $A^* A^* = A^*$
- ▶ $A^{**} = A^*$
- ▶ $A^* = \{\epsilon\} \cup AA^* = \{\epsilon\} \cup A^* A$
- ▶ $\emptyset^* = \{\epsilon\}$

Outline

1. Introduction
2. Basic Definitions
3. Deterministic Finite Automata
4. Intermezzo
5. Closure Properties
6. Further Reading

Example

DFA $M = (Q, \Sigma, \delta, s, F)$



1 $Q = \{1, 2, 3, 4\}$

2 $\Sigma = \{a, b\}$

3 $\delta: Q \times \Sigma \rightarrow Q$

4 $s = 1$

5 $F = \{4\}$

δ	a	b
1	2	1
2	2	3
3	4	1
4	4	4

$b a b a a \in L(M)$

1 1 2 3 4 4

$a a b b b \notin L(M)$

1 2 2 3 1 1

$L(M) = \{x \in \Sigma^* \mid x \text{ contains } aba \text{ as substring}\}$

Definitions

► **deterministic finite automaton (DFA)** is quintuple $M = (Q, \Sigma, \delta, s, F)$ with

- ① Q : finite set of **states**
- ② Σ : **input alphabet**
- ③ $\delta: Q \times \Sigma \rightarrow Q$: **transition function**
- ④ $s \in Q$: **start state**
- ⑤ $F \subseteq Q$: **final (accept) states**

► $\hat{\delta}: Q \times \Sigma^* \rightarrow Q$ is inductively defined by

$$\hat{\delta}(q, \epsilon) = q$$

$$\hat{\delta}(q, xa) = \delta(\hat{\delta}(q, x), a)$$

► string $x \in \Sigma^*$ is **accepted** by M if $\hat{\delta}(s, x) \in F$

► string $x \in \Sigma^*$ is **rejected** by M if $\hat{\delta}(s, x) \notin F$

► language accepted by M : $L(M) = \{x \in \Sigma^* \mid \hat{\delta}(s, x) \in F\}$

► set $A \subseteq \Sigma^*$ is **regular** if $A = L(M)$ for some DFA M

Outline

1. Introduction

2. Basic Definitions

3. Deterministic Finite Automata

4. Intermezzo

5. Closure Properties

6. Further Reading

 with session ID **3017 2066**

Question

What is the language accepted by the DFA given by the following transition table ?

	a	b
→ 1 F	2	1
2 F	3	1
3	3	3

Here the arrow indicates the start state and F marks the final states.

A $\{(ab)^n \mid n \in \mathbb{N}\}$

B $\sim(\{a, b\}^* \{aa\})$

☒ the set of all strings over $\{a, b\}$ not containing two consecutive a 's

D the set of all strings over $\{a, b\}$ with an odd number of a 's and an even number of b 's



Outline

1. Introduction
2. Basic Definitions
3. Deterministic Finite Automata
4. Intermezzo
5. Closure Properties
6. Further Reading

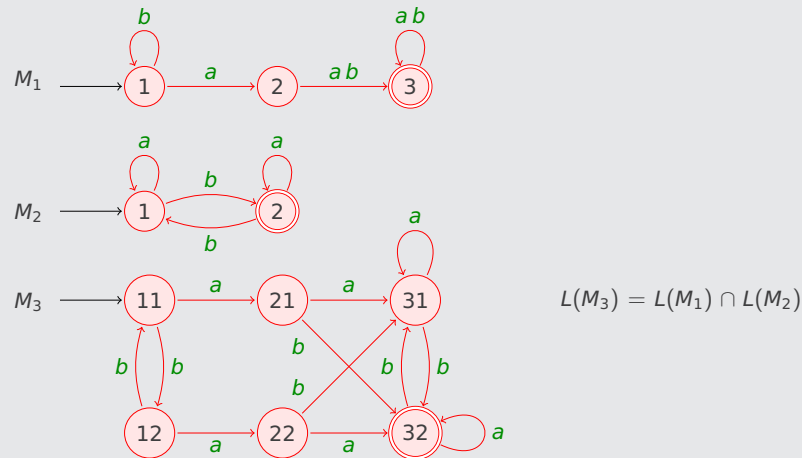
Theorem

regular sets are **effectively** closed under **intersection**

Proof (product construction)

- ▶ $A = L(M_1)$ for DFA $M_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$
 $B = L(M_2)$ for DFA $M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$
- ▶ $A \cap B = L(M_3)$ for DFA $M_3 = (Q_3, \Sigma, \delta_3, s_3, F_3)$ with
 - ① $Q_3 = Q_1 \times Q_2 = \{(p, q) \mid p \in Q_1 \text{ and } q \in Q_2\}$
 - ② $F_3 = F_1 \times F_2$
 - ③ $s_3 = (s_1, s_2)$
 - ④ $\delta_3((p, q), a) = (\delta_1(p, a), \delta_2(q, a))$ for all $p \in Q_1, q \in Q_2, a \in \Sigma$
- ▶ claim: $\hat{\delta}_3((p, q), x) = (\hat{\delta}_1(p, x), \hat{\delta}_2(q, x))$ for all $x \in \Sigma^*$
 proof of claim: easy induction on $|x|$ (on next slide)

Example



Proof of Claim

claim: $\hat{\delta}_3((p, q), x) = (\hat{\delta}_1(p, x), \hat{\delta}_2(q, x))$ for all $x \in \Sigma^*$

- ▶ base case: $|x| = 0$ and thus $x = \epsilon$
 $\hat{\delta}_3((p, q), x) = (p, q) = (\hat{\delta}_1(p, x), \hat{\delta}_2(q, x))$

- ▶ induction step: $|x| > 0$ and thus $x = ya$ with $|y| = |x| - 1$

$$\begin{aligned}
 \hat{\delta}_3((p, q), x) &= \delta_3(\hat{\delta}_3((p, q), y), a) && \text{(definition of } \hat{\delta}_3) \\
 &= \delta_3((\hat{\delta}_1(p, y), \hat{\delta}_2(q, y)), a) && \text{(induction hypothesis)} \\
 &= (\delta_1(\hat{\delta}_1(p, y), a), \delta_2(\hat{\delta}_2(q, y), a)) && \text{(definition of } \delta_3) \\
 &= (\hat{\delta}_1(p, x), \hat{\delta}_2(q, x)) && \text{(definition of } \hat{\delta}_1 \text{ and } \hat{\delta}_2)
 \end{aligned}$$

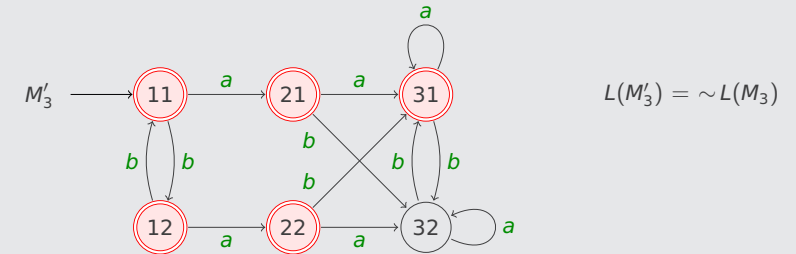
Theorem

regular sets are **effectively** closed under **complement**

Proof

- ▶ $A = L(M_1)$ for DFA $M_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$
- ▶ $\sim A = \Sigma^* - A = L(M_2)$ for DFA $M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$ with
 - ① $Q_2 = Q_1$
 - ② $\delta_2(q, a) = \delta_1(q, a)$ for all $q \in Q_2$ and $a \in \Sigma$
 - ③ $s_2 = s_1$
 - ④ $F_2 = Q_1 - F_1$

Example



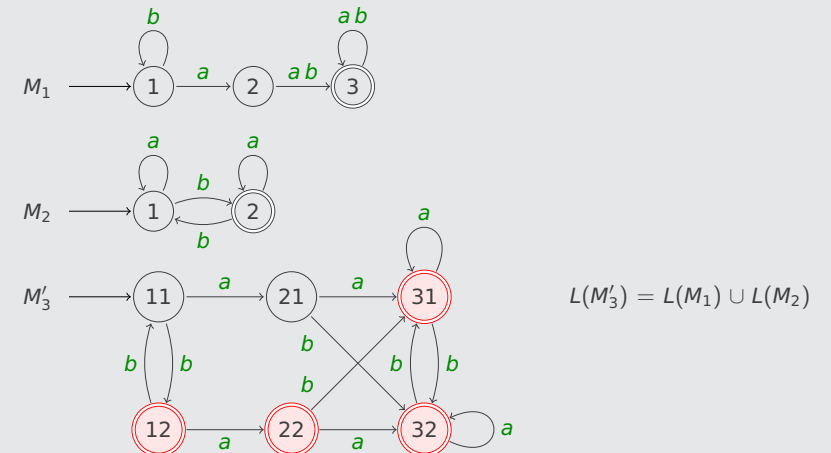
Theorem

regular sets are **effectively** closed under **union**

Proof (explicit construction)

- ▶ $A = L(M_1)$ for DFA $M_1 = (Q_1, \Sigma, \delta_1, s_1, F_1)$
- ▶ $B = L(M_2)$ for DFA $M_2 = (Q_2, \Sigma, \delta_2, s_2, F_2)$
- ▶ $A \cup B = L(M_3)$ for DFA $M_3 = (Q_3, \Sigma, \delta_3, s_3, F_3)$ with
 - ① $Q_3 = Q_1 \times Q_2 = \{(p, q) \mid p \in Q_1 \text{ and } q \in Q_2\}$
 - ② $F_3 = (F_1 \times Q_2) \cup (Q_1 \times F_2)$
 - ③ $s_3 = (s_1, s_2)$
 - ④ $\delta_3((p, q), a) = (\delta_1(p, a), \delta_2(q, a))$ for all $p \in Q_1, q \in Q_2, a \in \Sigma$

Example



Outline

- 1. Introduction
- 2. Basic Definitions
- 3. Deterministic Finite Automata
- 4. Intermezzo
- 5. Closure Properties
- 6. Further Reading

Kozen

- ▶ Lectures 1 – 4

Important Concepts

- | | | |
|----------------------|------------------------|---------------|
| ▶ alphabet | ▶ language | ▶ regular set |
| ▶ closure properties | ▶ product construction | ▶ string |
| ▶ DFA | | |

homework for October 9

Solutions

- ... must be uploaded (PDF format) in OLAT before 7 am on Friday
- ... bonus exercises give bonus points