



Automata and Logic

Aart Middeldorp and Samuel Frontull

Definitions

- ▶ **regular expression** α over alphabet Σ :

$$a \in \Sigma \quad \epsilon \quad \emptyset \quad \beta + \gamma \quad \beta\gamma \quad \beta^*$$

- ▶ set of strings $L(\alpha) \subseteq \Sigma^*$ matched by regular expression α :

$$\begin{array}{lll} L(a) = \{a\} & L(\emptyset) = \emptyset & L(\beta\gamma) = L(\beta)L(\gamma) \\ L(\epsilon) = \{\epsilon\} & L(\beta + \gamma) = L(\beta) \cup L(\gamma) & L(\beta^*) = L(\beta)^* \end{array}$$

- ▶ regular expressions α and β are **equivalent** ($\alpha \equiv \beta$) if $L(\alpha) = L(\beta)$

Theorem

finite automata and regular expressions are **equivalent**:

$$\text{for all } A \subseteq \Sigma^* \quad A \text{ is regular} \iff A = L(\alpha) \text{ for some regular expression } \alpha$$

Outline

1. Summary of Previous Lecture
2. Minimization
3. Intermezzo
4. Weak Monadic Second-Order Logic
5. Further Reading

Definitions

- ▶ **homomorphism** is mapping $h: \Sigma^* \rightarrow \Gamma^*$ such that $h(\epsilon) = \epsilon$ and $h(xy) = h(x)h(y)$
- ▶ if $A \subseteq \Sigma^*$ then $h(A) = \{h(x) \mid x \in A\} \subseteq \Gamma^*$ "image of A under h "
- ▶ if $B \subseteq \Gamma^*$ then $h^{-1}(B) = \{x \mid h(x) \in B\} \subseteq \Sigma^*$ "preimage of B under h "

Theorem

regular sets are effectively closed under **homomorphic image** and **preimage**

Theorem

problems

instance: DFA M and string x

question: $x \in L(M)$?

instance: DFA M

question: $L(M) = \emptyset$?

instance: DFAs M and N

question: $L(M) = L(N)$?

are decidable

Automata

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

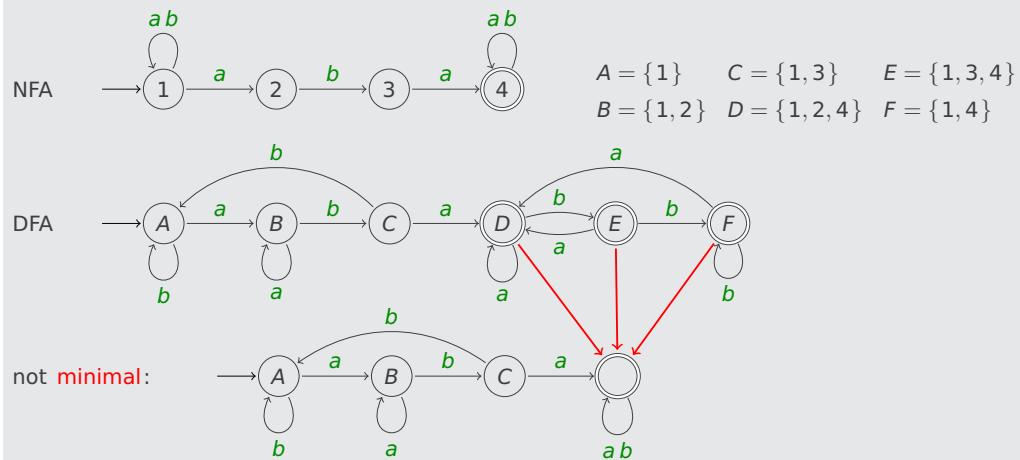
Logic

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

Outline

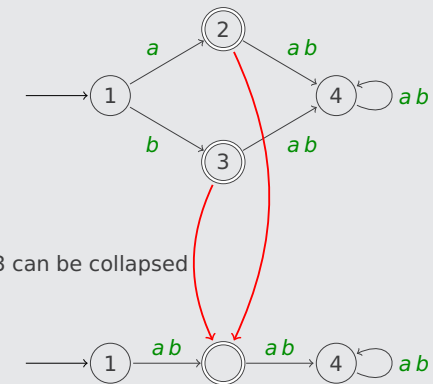
1. Summary of Previous Lecture
2. Minimization
3. Intermezzo
4. Weak Monadic Second-Order Logic
5. Further Reading

Example 1



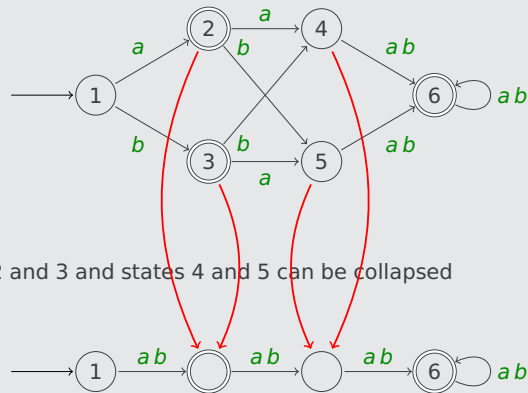
Example 2

DFA

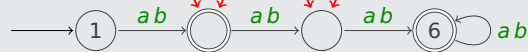


Example 3

DFA



not **minimal**: states 2 and 3 and states 4 and 5 can be collapsed



Definitions

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

- state p is **inaccessible** if $\hat{\delta}(s, x) \neq p$ for all $x \in \Sigma^*$
- states p and q are **distinguishable** if

$$(\hat{\delta}(p, x) \in F \wedge \hat{\delta}(q, x) \notin F) \vee (\hat{\delta}(p, x) \notin F \wedge \hat{\delta}(q, x) \in F)$$

for some $x \in \Sigma^*$

Minimization Algorithm

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

- remove inaccessible states
- for every two different states determine whether they are distinguishable (**marking**)
- collapse** indistinguishable states

Marking Algorithm

given DFA $M = (Q, \Sigma, \delta, s, F)$ without inaccessible states

- tabulate all unordered pairs $\{p, q\}$ with $p, q \in Q$, initially unmarked
- mark $\{p, q\}$ if $p \in F$ and $q \notin F$ or $p \notin F$ and $q \in F$
- repeat until no change:
mark $\{p, q\}$ if $\{\delta(p, a), \delta(q, a)\}$ is marked for some $a \in \Sigma$

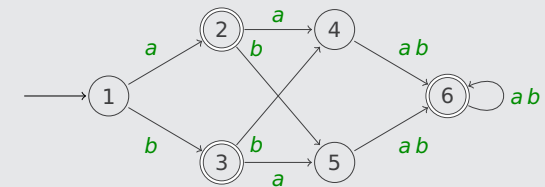
Notation

$p \approx q \iff$ states p and q are indistinguishable

Lemma

$p \approx q \iff \{p, q\}$ is unmarked

Example



1
✓ 2
✓ 3
✓ ✓ ✓ 4
✓ ✓ ✓ 5
✓ ✓ ✓ ✓ ✓ 6

- final / nonfinal** states are distinguishable
- $\{2, 6\} \xrightarrow{a} \{4, 6\}$ $\{3, 6\} \xrightarrow{a} \{5, 6\}$
- $\{1, 4\} \xrightarrow{a} \{2, 6\}$ $\{1, 5\} \xrightarrow{a} \{2, 6\}$

collapse states 2 and 3 and states 4 and 5:

Definition

states p and q of DFA $M = (Q, \Sigma, \delta, s, F)$ are **indistinguishable** ($p \approx q$) if for all $x \in \Sigma^*$

$$\widehat{\delta}(p, x) \in F \iff \widehat{\delta}(q, x) \in F$$

Lemma

$\approx$ is **equivalence relation** on Q :

- 1 $\forall p \in Q \quad p \approx p$ (reflexivity)
- 2 $\forall p, q \in Q \quad p \approx q \implies q \approx p$ (symmetry)
- 3 $\forall p, q, r \in Q \quad p \approx q \wedge q \approx r \implies p \approx r$ (transitivity)

Notation

$[p]_{\approx} = \{q \in Q \mid p \approx q\}$ denotes **equivalence class** of p

Definition (Collapsing Indistinguishable States)

DFA $M/\approx$ is defined as $(Q', \Sigma, \delta', s', F')$ with

- ▶ $Q' = \{[p]_{\approx} \mid p \in Q\}$
- ▶ $\delta'([p]_{\approx}, a) = [\delta(p, a)]_{\approx}$ **well-defined:** $p \approx q \implies \delta(p, a) \approx \delta(q, a)$
- ▶ $s' = [s]_{\approx}$
- ▶ $F' = \{[p]_{\approx} \mid p \in F\}$

Lemma

- 1 $\widehat{\delta'}([p]_{\approx}, x) = [\widehat{\delta}(p, x)]_{\approx}$ for all $x \in \Sigma^*$
 - 2 $p \in F \iff [p]_{\approx} \in F'$
- for all $p \in Q$

Theorem

$$L(M/\approx) = L(M)$$

Proof

$$x \in L(M/\approx) \iff \widehat{\delta'}([s]_{\approx}, x) \in F' \iff [\widehat{\delta}(s, x)]_{\approx} \in F' \iff \widehat{\delta}(s, x) \in F \iff x \in L(M)$$

Question

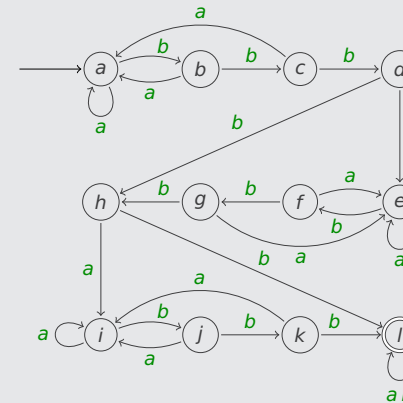
is $M/\approx$ minimum-state DFA for $L(M)$?

Lemma

$M/\approx$ cannot be collapsed further

Example

DFA for set of strings over $\{a, b\}$ containing at least three occurrences of three consecutive b 's, overlapping permitted:



a
✓ b
✓ ✓ c
✓ ✓ ✓ d
✓ ✓ ✓ ✓ e
✓ ✓ ✓ ✓ ✓ f
✓ ✓ ✓ ✓ ✓ ✓ g
✓ ✓ ✓ ✓ ✓ ✓ ✓ h
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ i
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ j
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ k
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ l

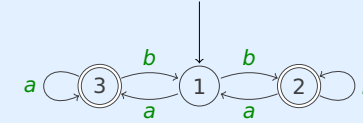
states d, g and h, k can be merged

Outline

1. Summary of Previous Lecture
2. Minimization
- 3. Intermezzo**
4. Weak Monadic Second-Order Logic
5. Further Reading

Question

Which statements about the following DFA A are true ?



- A** the DFA is minimal
- B** states 2 and 3 are distinguishable
- C** $L(A) = L(a^*ba^*)$
- D** all states can be merged



Outline

1. Summary of Previous Lecture
2. Minimization
3. Intermezzo
- 4. Weak Monadic Second-Order Logic**
5. Further Reading

Definitions

- first-order variables $V_1 = \{x, y, \dots\}$ ranging over natural numbers
- second-order variables $V_2 = \{X, Y, \dots\}$ ranging over finite sets of natural numbers
- formulas of **weak monadic second-order logic**

$$\varphi ::= \perp \mid x < y \mid X(x) \mid \neg \varphi \mid \varphi_1 \vee \varphi_2 \mid \exists x. \varphi \mid \exists X. \varphi$$

with $x, y \in V_1$ and $X \in V_2$

Abbreviations

$$\varphi \wedge \psi := \neg(\neg\varphi \vee \neg\psi)$$

$$\forall x. \varphi := \neg \exists x. \neg \varphi$$

$$x \leq y := \neg(y < x)$$

$$\top := \neg \perp$$

$$X(0) := \exists x. X(x) \wedge x = 0$$

$$\varphi \rightarrow \psi := \neg\varphi \vee \psi$$

$$\forall X. \varphi := \neg \exists X. \neg \varphi$$

$$x = y := x \leq y \wedge y \leq x$$

$$x = 0 := \neg \exists y. y < x$$

Remarks

- ▶ $X(x)$ represents $x \in X$
- ▶ MSO is weak MSO without restriction to finite sets

Examples

- ▶ $(\forall x. X(x) \rightarrow Y(x)) \wedge (\exists y. \neg X(y) \wedge Y(y))$
- ▶ $\exists X. (\forall x. x = 0 \rightarrow X(x)) \wedge (\forall x. X(x) \rightarrow \exists y. x < y \wedge X(y))$
- ▶ $\exists X. X(0) \wedge (\forall y. \forall z. z = y + 1 \wedge z \leq x \rightarrow (X(y) \leftrightarrow \neg X(z))) \wedge X(x) \iff x \text{ is even}$

Remark

$z = y + 1$ abbreviates $y < z \wedge \neg \exists x. (y < x \wedge x < z)$

Definitions

- ▶ assignment α is mapping from variables $x \in V_1$ to $\mathbb{N}$ and $X \in V_2$ to **finite** subsets of $\mathbb{N}$
- ▶ assignment α **satisfies** formula φ ($\alpha \models \varphi$):

$$\alpha \not\models \perp$$

$$\alpha \models x < y \iff \alpha(x) < \alpha(y)$$

$$\alpha \models X(x) \iff \alpha(x) \in \alpha(X)$$

$$\alpha \models \neg \varphi \iff \alpha \not\models \varphi$$

$$\alpha \models \varphi_1 \vee \varphi_2 \iff \alpha \models \varphi_1 \text{ or } \alpha \models \varphi_2$$

$$\alpha \models \exists x. \varphi \iff \alpha[x \mapsto n] \models \varphi \text{ for some } n \in \mathbb{N}$$

$$\alpha \models \exists X. \varphi \iff \alpha[X \mapsto N] \models \varphi \text{ for some finite subset } N \subset \mathbb{N}$$

Definitions

- ▶ formula φ is **satisfiable** if $\alpha \models \varphi$ for some assignment α
- ▶ formula φ is **valid** if $\alpha \models \varphi$ for all assignments α
- ▶ **model** of formula φ is assignment α such that $\alpha \models \varphi$
- ▶ **size** of model α is smallest n such that
 - ① $\alpha(x) < n$ for $x \in V_1$
 - ② $\alpha(X) \subseteq \{0, \dots, n-1\}$ for $X \in V_2$

Examples

- ▶ $(\forall x. X(x) \rightarrow Y(x)) \wedge (\exists y. \neg X(y) \wedge Y(y))$ satisfiable
- ▶ $(\forall x. x = 0 \rightarrow X(x)) \wedge (\forall x. X(x) \rightarrow \exists y. x < y \wedge X(y))$ unsatisfiable
- ▶ $(\exists x. X(x) \wedge \exists y. X(y) \wedge x \neq y) \wedge (\forall x. X(x) \rightarrow \exists y. Y(y) \wedge x < y)$ satisfiable

Definition

given alphabet Σ and string $x = a_0 \dots a_{n-1} \in \Sigma^*$

- ▶ second-order variables $V_2 = \{P_a \mid a \in \Sigma\}$
- ▶ $\alpha_x(P_a) = \{i < n \mid a_i = a\}$

Notation

$\underline{x}$ for α_x

Example

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

$$\underline{abba}(P_a) = \{0, 3\}$$

$$\underline{abba}(P_b) = \{1, 2\}$$

Example

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

- ▶ $\varphi = \forall x. \neg(P_a(x) \wedge P_b(x))$ $\underline{x} \models \varphi$ for all $x \in \Sigma^*$
- ▶ $\psi = \forall x. \forall y. (P_a(x) \wedge P_b(y)) \rightarrow x < y$ $\underline{aabb} \models \psi$ $\underline{aabab} \not\models \psi$
- ▶ $\chi = \forall x. P_b(x) \rightarrow \exists y. P_a(y) \wedge y < x$ $\underline{aaaaa} \models \chi$ $\underline{babab} \not\models \chi$

Definitions

- ▶ given alphabet Σ and WMSO formula φ with free variables (exclusively) in $\{P_a \mid a \in \Sigma\}$

$$L(\varphi) = \{x \in \Sigma^* \mid \underline{x} \models \varphi\}$$

- ▶ set $A \subseteq \Sigma^*$ is **WMSO definable** if $A = L(\varphi)$ for some WMSO formula φ

Examples

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

- ▶ regular set $L((a+b)^*ab(a+b)^*)$ is WMSO definable by formula

$$\exists x. \exists y. P_a(x) \wedge P_b(y) \wedge x < y \wedge \neg \exists z. x < z \wedge z < y$$

- ▶ WMSO formula

$$\exists x. P_a(x) \wedge \forall y. x < y \rightarrow \neg(P_a(y) \vee P_b(y))$$

defines regular set $\{xa \mid x \in \Sigma^*\}$

Theorem

set $A \subseteq \Sigma^*$ is regular if and only if A is WMSO definable

Outline

1. Summary of Previous Lecture
2. Minimization
3. Intermezzo
4. Weak Monadic Second-Order Logic
5. Further Reading

Kozen

- ▶ Lectures 13 and 14

Important Concepts

- ▶ $\alpha \models \varphi$
- ▶ indistinguishable states
- ▶ minimization algorithm
- ▶ model
- ▶ MSO
- ▶ satisfiability
- ▶ validity
- ▶ weak monadic second-order logic (WMSO)
- ▶ WMSO definability

homework for October 31