DFA Minimization and Closure Operations

Closure operations

Automata

Reverse

Star Closure

Homomorphisms

Automata Minimization and operations with regular languages.

DSIC - UPV

Closure Operations

DFA Minimization and Closure Operations

Closure operations

Automata Boolean operations Reverse Concatenation

Star Closure Homomorphisms

- A set C is closed under an operation \cdot iff for any elements $x, y \in C$, $x \cdot y \in C$.
- Examples
 - Let $C = \{L \subseteq \Sigma^* \mid L \text{ es finite } \}$. The union and the intersection are closed in C, whereas the complement is not.

Automata

DFA Minimization and Closure Operations

Closure operation:

Automata Boolean opera

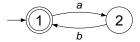
Reverse

Concatenatio

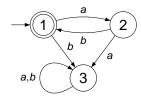
Star Closure Homomorphism

Automata Minimizatior Automaton A_1 (not complete)

$$L(A_1) = \{(ab)^n \mid n \ge 0\} = \{ab\}^*$$



Automaton A_2 (complete). Note that $L(A_2) = L(A_1)$



Automata

DFA Minimization and Closure Operations

Closure operation:

Automata

Boolean operation

Reverse

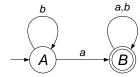
Star Closure

Homomorphism

Automata Minimizatio

Automaton A₃

$$L(A_3) = \{x \in \{a, b\}^* \mid |x|_a > 0\} = \{a, b\}^* \{a\} \{a, b\}^* = \{b\}^* \{a\} \{a, b\}^*$$



Boolean operations Intersection

Minimization and Closure Operations

Regular languages are closed under intersection:

Let $L_1, L_2 \in \mathcal{L}_3$, then there exist two automata A_1, A_2 such that $L_1 = L(A_1), L_2 = L(A_2)$, where

$$A_i = (Q_i, \Sigma, \delta_i, q_i, F_i), i = 1, 2$$

We build $A' = (Q, \Sigma, \delta, q_0, F)$ where:

$$\blacksquare \ Q = Q_1 \times Q_2$$

$$q0 = [q1, q2]$$

$$\blacksquare$$
 $F = F1 \times F2$

■
$$\delta([p_1, p_2], a) = [\delta_1(p_1, a), \delta_2(p_2, a)], p_1 \in Q1, p_2 \in Q2, a \in \Sigma$$

$$\Rightarrow L(A') = L(A_1) \cap L(A_2)$$

Boolean operations Intersection

DFA Minimization and Closure Operations

Closure operation:

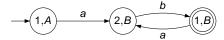
Boolean operations

Doolean operation

Concatenation

Star Closure Homomorphism

Automata Minimization Automaton for $L(A_1) \cap L(A_3)$.



Boolean operations Union

Minimization and Closure Operations

Regular languages are closed under **Union**:
Let
$$L_1, L_2 \in \mathcal{L}_2$$
, then there exist two *complet*:

Let $L_1, L_2 \in \mathcal{L}_3$, then there exist two *complete* automata A_1, A_2 such that $L_1 = L(A_1), L_2 = L(A_2)$, where

$$A_i = (Q_i, \Sigma, \delta_i, q_i, F_i), i = 1, 2$$

We build $A' = (Q, \Sigma, \delta, q_0, F)$ where:

$$\blacksquare \ Q = Q_1 \times Q_2$$

$$q0 = [q1, q2]$$

$$\blacksquare$$
 $F = F1 \times Q \times \cup Q \times F2$

■
$$\delta([p_1, p_2], a) = [\delta_1(p_1, a), \delta_2(p_2, a)], p_1 \in Q1, p_2 \in Q2, a \in \Sigma$$

$$\Rightarrow L(A') = L(A_1) \cup L(A_2)$$

Boolean operations Union

DFA Minimization and Closure Operations

Closure operations

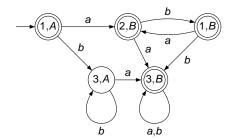
Boolean operations

Boolean operation

Concatenation

Star Closure Homomorphism

Automata Minimization Automaton for $L(A_2) \cup L(A_3)$.



Boolean operations Complement (and Difference)

DFA Minimization and Closure Operations

Closure
pperations
Automata
Boolean operations
Reverse
Concatenation
Star Closure
Homographisms

- Regular languages are closed under **Complement**. Let $L \in \mathcal{L}_3$, then there exists a complete automaton A such that L = L(A) where $A = (Q, \Sigma, \delta, q_0, F)$. Automaton $A' = (Q, \Sigma, \delta, q_0, Q - F)$ accepts L^c
- Regular languages are closed under **Difference**. Let $L_1, L_2 \in \mathcal{L}_3$. Note that L_1 - $L_2 = L_1 \cap L_2^c$.

Boolean operations Complement

DFA Minimization and Closure Operations

Closure operation:

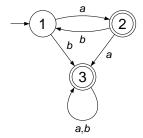
Roolean operations

Doolean operation

Concatenation

Homomorphism

Automata Minimization Automaton for $L(A_2)^c$.



Reverse

DFA Minimization and Closure Operations

Closure operations Automata Boolean operations

Reverse Concatenation Star Closure

Automata Minimization Regular languages are closed under the operation **Reverse**.

Let $L \in \mathcal{L}_3$, then there exists an automaton

 $A = (Q, \Sigma, \delta, q_0, q_f)$ If |F| > 1, A can be modified to have one final state (How?).

We build $A' = (Q, \Sigma, \delta', q_f, q_0)$ where:

$$q \in \delta(p, a) \leftrightarrow p \in \delta'(q, a).$$

$$\Rightarrow L(A') = L(A')$$

Reverse

DFA Minimization and Closure Operations

Closure operations

Automata

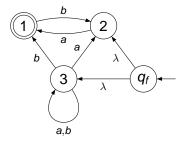
Boolean operati

Concatenatio

Star Closure Homomorphisms

Automata Minimization

Automaton for $(L(A_2)^c)^r$.



Concatenation

DFA Minimization and Closure Operations

Closure
operations
Automata
Boolean operation
Reverse

Concatenation
Star Closure
Homomorphisms

Homomorphisms

Automata Minimizatior Regular languages are closed under Concatenation. Let $I_A I_B \in \mathcal{L}_B$, then there exist two automata $A_A A_B$ si

Let $L_1, L_2 \in \mathcal{L}_3$, then there exist two automata A_1, A_2 such that $L_1 = L(A_1), L_2 = L(A_2)$, where

 $A_i = (Q_i, \Sigma, \delta_i, q_i, F_i), (i = 1, 2)$ and such that $Q1 \cap Q2 = \emptyset$

We build $A' = (Q, \Sigma, \delta', q_1, F_2)$ donde:

 $\blacksquare \ Q = Q_1 \cup Q_2$

 \bullet $\delta' = \delta_1 \cup \delta_2 \cup \delta''$ where $q_2 \in \delta''(p, \lambda)$, for any $p \in F_1$

$$\Rightarrow L(A') = L(A_1) \cdot L(A_2)$$

Concatenation

DFA Minimization and Closure Operations

Closure operations

Automata

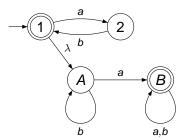
Boolean oper

oncatenatio

Star Closure

Automorphisms

Automaton for $L(A_1) \cdot L(A_3)$.



Star Closure

Minimization and Closure Operations

Regular languages are closed under Star Closure. Let $L \in \mathcal{L}_3$, then there exists an automaton A such that

L = L(A) where $A = (Q, \Sigma, \delta_0, q_0, F)$ We build

$$A' = (Q', \Sigma, \delta', q_n, F)$$
 where:

$$\blacksquare F = F \cup \{q_n\}$$

■
$$\delta'(p, a) = \delta(p, a)$$
, for every $p \in Q$ and every $a \in \Sigma$

$$\blacksquare q_n \in \delta'(p, \lambda), \text{ for every } p \in F$$

Star Closure

DFA Minimization and Closure Operations

Closure operations

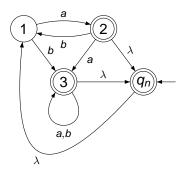
operations Automata

Boolean ope

Concatenat

Homomorphisn

Automata Minimization Automaton for $(L(A_2)^c)^*$.



Homomorphisms

DFA Minimization and Closure Operations

Closure
operations
Automata
Boolean operations
Reverse
Concatenation
Star Closure

Automata Minimization ■ Regular languages are closed under **Homomorphisms**.

Regular languages are closed under Inverse Homomorphisms.

Let $h: \Sigma \to \Delta^*$ and $L \in \mathcal{L}_3$, there exists an automaton A such that L = L(A), where $A = (Q, \Sigma, \delta, q_0, F)$. We build $A' = (Q, \Sigma, \delta', q_0, F)$ with:

$$\delta'(p, a) = \begin{cases} \delta(p, h(a)) & \text{if } \delta(p, h(a)) \neq \emptyset \\ \emptyset & \text{otherwise} \end{cases}$$

Homomorphism

DFA Minimization and Closure Operations

Closure operations

Automoto

Boolean ope

Reverse

Concatenation

Star Closure

Homomorphism

Homomorphism

$$\Sigma = \{a, b\}, \Delta = \{0, 1, 2\}$$

■
$$h(a) = 0, h(b) = 12$$

Inverse Homomorphism

DFA Minimization and Closure Operations

Closure operations

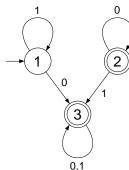
Automata

Boolean operations Reverse

Reverse Concatenation

Homomorphism

Automata Minimization $\Sigma = \{0, 1\}, \Delta = \{a, b\}. \ g(0) = ab, \ h(1) = ba.$ Automaton for $g^{-1}(L(A_2)^c)$.



DFA Minimization and Closure Operations

Closure operations
Automata
Boolean operations
Reverse
Concatenation
Star Closure
Homomorphisms

Automata Minimization ■ A DFA $A = (Q, \Sigma, \delta, q_0, F)$ is reachable if for every $q \in Q$ there exists a word $x \in \Sigma$ such that $\delta(q_0, x) = q$

■ Let $A = (Q, \Sigma, \delta, q_0, F)$ be a complete and reachable DFA. The indistinguishability relation \sim en Q is defined $\forall q, q' \in Q$:

$$(q \sim q' \leftrightarrow \forall x \in \Sigma^*(\delta(q, x) \in F \leftrightarrow \delta(q', x) \in F))$$

- Let $A = (Q, \Sigma, \delta, q_0, F)$ be a complete and reachable DFA and let \sim be the indistinguishability relation. We define the quotient automaton $A/\sim=(Q, \Sigma, \delta, q_0, F)$ as:
 - \square $Q = [q]_{\sim} \mid q \in Q$
 - $q_0 = [q_0]_{\sim}$
 - $F = \{[q] \mid q \in F\}$
 - $\delta([q]_{\sim}, a) = [\delta(q, a)]_{\sim}$

DFA Minimization and Closure Operations

operations
Automata
Boolean operation
Reverse
Concatenation
Star Closure

- Sea $A = (Q, \Sigma, \delta, q_0, F)$ be a complete and reachable DFA and let \sim be the indistinguishability relation. The automaton A/\sim is the minimum DFA accepting L(A)
- Let $A = (Q, \Sigma, \delta, q_0, F)$ be a complete and reachable DFA and let $k \ge 0$ be an integer. The k-indistinguishability relation \sim_k is defined:

$$orall q, q' \in \mathsf{Q} : (q \sim_k q \leftrightarrow \forall x \in \Sigma^*, |x| \le k, (\delta(q, x) \in \mathsf{F} \leftrightarrow \delta(q, x) \in \mathsf{F}))$$

- Properties of \sim_k :

 - $\blacksquare \ \forall k \geq 0, p \sim_{k+1} q \leftrightarrow (p \sim_k q \land \forall a \in \Sigma, \delta(p, a) \sim_k \delta(q, a))$

DFA Minimization and Closure Operations

Closure operations

Boolean operation Reverse Concatenation

Star Closure Homomorphism

Automata Minimization

Minimization Algorithm:

- 1. $\pi_0 = \{Q F, F\}$
- 2. Obtain π_{k+1} from π_k $B(p, \pi_{k+1}) == B(q, \pi_{k+1})$ if and only if
 - $\blacksquare B(p, \pi_k) == B(q, \pi_k)$
 - y For every $a \in \Sigma$, $B(\delta(p, a), \pi_k) == B(\delta(q, a))$
- 3. If π_{k+1} is different from π_k go to 2

DFA

Minimization and Closure Operations

Closure operation:

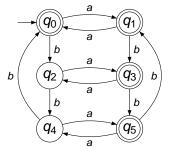
Automata

Boolean operation

Concatona

Stor Closu

Homomorphism



DFA Minimization and Closure Operations

Closure operations

Automata

Boolean ope

Concatena

Star Closur

Homomorphism

			а	b
	B_0	q_0	B_0	B_1
		q_1	B_0	B_0
π_0 :		q_3	B_1	B_0
		q ₅	B_1	B_0
	<i>B</i> ₁	q_2	B_0	B_1
		q_4	B_0	B_0

DFA Minimization and Closure Operations

operations

Automata

Boolean operation

Star Closure

			а	b
'	B_0	q_0	B_1	B_3
•	<i>B</i> 1	q_1	B_0	B_2
π_1 :	B2	q ₃	B_3	B_2
		q 5	B_4	B_1
•	<i>B</i> ₃	q_2	B_2	B_4
	B_4	q_4	B_2	B_0

DFA Minimization and Closure Operations

Closure operations

Automata

Dovices of

Concatena

Star Closure

			а	b
•	B_0	q_0	B_1	B_4
•	<i>B</i> 1	q_1	B_0	B_2
π_2 :	B2	q_3	B_4	B_3
•	<i>B</i> 3	q ₅	B_5	B_1
•	B_4	q_2	B_2	B_5
•	B_5	q_4	B_3	B_0

DFA

Minimization and Closure Operations

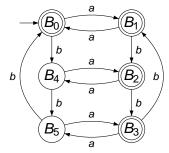
Closure operation:

Automata Boolean operations

Reverse .

Star Closure Homomorphism

Automata Minimization



 $\pi_3 = \pi_2$

DFA

Minimization and Closure Operations

Closure operation

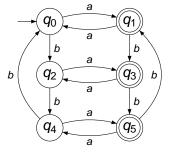
Automata

Boolean operatio

Concatena

01-- 01----

Homomorphism



DFA Minimization and Closure Operations

Closure perations

Automata

Reverse

Concatena

Star Closure

			а	b
	B_0	q_1	B_1	B_0
		q ₃	B_1	B_0 B_0
π_0 :		q 5	B_1	B_0
	<i>B</i> ₁	q_0	B_0	<i>B</i> ₁
		q_2	B_0	B_1
		q_4	B_0	B_1

DFA

Minimization and Closure Operations

Closure operations

Automata

Boolean op

Reverse

Concatenati

Star Closure

Automata Minimization $\pi_1 = \pi_0$

