Properties of equivalence relation on Σ^*

Definition (right congruence)

An equivalence relation $\equiv_A$ defined on Σ^* is said to be **a right congruence** if $\forall x, y \in \Sigma^*$ and $\forall a \in \Sigma$, $x \equiv y \implies x \cdot a \equiv y \cdot a$.

Definition (Refinement)

An equivalence relation $\equiv$ is said to **refine** a language L, if $x \equiv y$ then $(x \in L \iff y \in L)$.

Definition (Finite index)

An equivalence relation is said to have **finite index** if the number of equivalence classes defined by $\equiv$ is finite.

Lemma

For a DFA A, the equivalence relation $\equiv_A$ defined as before is

is a right congruence, refines $L(A)$, has finite index.

Properties of $\equiv_A$

Lemma

For a DFA A, the equivalence relation $\equiv_A$ defined as before is

is a right congruence, refines $L(A)$, has finite index.

Proof.

right congruence

$$\begin{aligned}\tilde{\delta}(q_0, x \cdot a) &= \delta(\tilde{\delta}(q_0, x), a) \\ &= \delta(\tilde{\delta}(q_0, y), a) \because x \equiv_A y \\ &= \tilde{\delta}(q_0, y \cdot a)\end{aligned}$$

finite index

For $q \in Q$,

$[q] := \{w \in \Sigma^* \mid \tilde{\delta}(q_0, w) = q\}$

$\#$ equivalence classes $\leq |Q|$.

refinement

If $x \equiv_A y$

then $\tilde{\delta}(q_0, x) = \tilde{\delta}(q_0, y)$

$\therefore x, y$ both accepted or

both rejected.

$\square$

Myhill-Nerode relation

Definition

An equivalence relation $\equiv$ on Σ^* is said to be **a Myhill-Nerode relation** for a language L if

- it is a right congruence
- refining L
- and has a finite index.

Lemma (Regular language $\implies$ Myhill-Nerode relation)

For any regular language there is a Myhill-Nerode relation.

What about the converse?

Non-regular languages

Let $L_{a,b} = \{a^n b^n \mid n \geq 0\}$.

Consider any relation $\equiv$ on $\{a, b\}^*$.

Assume that it is a right congruence and refines L.

Now we will show that it does not have finite index.

For $n \neq m$, can $a^n \equiv a^m$? NO!

$\because a^n b^n \in L$ but $a^m b^n \notin L$.

Let $FACTORIAL = \{a^{n!} \mid n \geq 0\}$.

Consider any relation $\equiv$ on $\{a\}^*$.

Assume that it is a right congruence and refines L.

Now we will show that it does not have finite index.

Can $a^{n!} \equiv a^{n+1!}$? NO!

$\because a^{n!} \cdot a^{n \cdot n!} \in L$ but $a^{n+1!} \cdot a^{n \cdot n!} \notin L$.

Converse also holds

Lemma

Let $L \subseteq \Sigma^*$. If there is a Myhill-Nerode relation for L then L is regular.

Proof idea

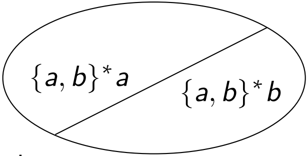
Using the relation, construct a finite state automaton.

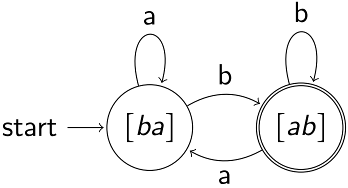
Let each equivalence class of the relation be a state of the automaton.

Define transitions naturally.

Converse: Myhill-Nerode relation $\implies$ regulararity

Example





$L = \{\text{words ending with } b\}$

Converse also holds

Lemma

Let $L \subseteq \Sigma^*$. If there is a Myhill-Nerode relation for L then L is regular.

Proof.

Construction

Let $\equiv$ be a Myhill-Nerode relation.

Let $[x] = \{y \mid y \equiv x\}$.

Let $A_{\equiv} = (Q, \Sigma, \delta, q_0, F)$ be defined as follows:

$Q = \{[x] \mid x \in \Sigma^*\}$,

$q_0 = [\epsilon]$, $F = \{[x] \mid x \in L\}$,

$\delta([x], a) = [xa]$.

Correctness: DIY. $\square$

Decision problems on regular languages

Acceptance problem (for fixed Σ)

Given: DFA A, input string $w \in \Sigma^*$

Output: "yes" iff A accepts w.

Construct a graph from an automaton:

Let $Q = \{q_0, \ldots, q_{m-1}\}$, q_0 be the start state,
$F \subseteq Q$ be the set of final states.

Create a layered graph $G_{A,n}$, where $|w| = n$, as follows:

Make $n+1$ copies of Q: $Q_0, Q_1, \ldots, Q_n$, where $Q_i = \{q_{i,0}, \ldots, q_{i,m-1}\}$.

Add edge $(q_{i,u}, q_{i+1,v})$ with label $a \in \Sigma$
if $\delta(q_u, a) = q_v$.

Lemma

There is a path from $q_{0,0}$ to $q_{n,u}$ labelled by a string w in $G_{A,|w|}$ if and only if $\tilde{\delta}(q_0, w) = q_u$ in A.

Decision problems on regular languages

Nonemptiness problem (for fixed Σ)

Given: DFA A

Output: "yes" iff $\exists w : A$ accepts w.

Lemma

If a DFA $A = (Q, \Sigma, \delta, q_0, F)$ accepts some string then it accepts a string of length $\leq |Q|$.

Minimization problem

Minimization problem (for fixed Σ)

Given: DFA A

Output: DFA B s.t. $L(A) = L(B)$ and B has the smallest number of states possible for recognizing $L(A)$

Definition

Let $A = (Q, \Sigma, \delta, q_0, F)$. We call states p, q indistinguishable if $\forall w \in \Sigma^*, \tilde{\delta}(p, w) \Leftrightarrow \tilde{\delta}(q, w)$.

Minimization algorithm.

Identify indistinguishable states.

Collapse them.

Minimization problem

Minimization problem (for fixed Σ)

Given:	DFA A
Output:	DFA B s.t. $L(A) = L(B)$ and B has the smallest number of states possible for recognizing $L(A)$

Example

	0	1	2	3	4	5
a	1	2	3	4	5	0

(Red color indicates final states.)

0					
–	1				
–	–	2			
–	–	–	3		
–	–	–	–	4	
–	–	–	–	–	5

0					
✓	1				
✓	✓	2			
–	✓	✓	3		
✓	–	✓	✓	4	
✓	✓	–	✓	✓	5

Minimization problem

Minimization problem (for fixed Σ)

Given:	DFA A
Output:	DFA B s.t. $L(A) = L(B)$ and B has the smallest number of states possible for recognizing $L(A)$

Example

	0	1	2	3	4	5
a	1	3	4	5	5	5
b	2	4	3	5	5	5

(Red color indicates final states.)