Tema 2: Lenguajes regulares

Autómatas finitos y gramáticas regulares

- Autómatas finitos deterministas
- Autómatas finitos no deterministas
- Gramáticas regulares (y lineales) a la derecha

Expresiones regulares

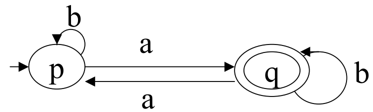
- Expresiones regulares
- Autómatas finitos no deterministas con transiciones vacías

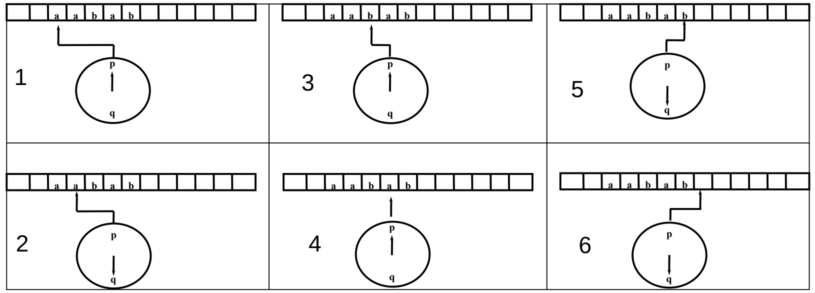
Propiedades de los lenguajes regulares

- Minimización de autómatas
- Algunas propiedades de cierre
- Lenguajes no regulares

Aplicaciones

Idea de autómata





Autómata finito determinista (AFD)

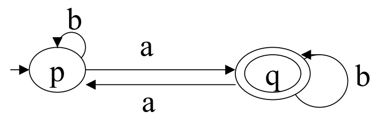
- **Elementos:**

Componentes físicos	Componentes lógicos	
Unidad de Proceso	Q	conjunto de estados
Fuente de entrada	Σ	alfabeto de entrada

- **Ciclo-máquina**:
  Consultas: estado actual y símbolo de entrada
  Acciones: avance en la entrada y cambio de estado
- **Elementos distinguidos** para la inicialización y salida:
  $q_0 \in Q$: estado inicial y $F \subseteq Q$: conjunto de estados finales
- **Definición formal: M = (Q, Σ, δ, q_0, F)** con $\delta : Q \times \Sigma \longrightarrow Q$
  (función parcial)

Ejemplo de AFD

Formalmente: **M = (Q, Σ, δ, q_0, F)** con $\delta : Q \times \Sigma \longrightarrow Q$



Este autómata es:

M = ({p,q}, {a,b}, δ, p, {q}) con δ definida:

δ(p,a) = q
δ(p,b) = p
δ(q,a) = p
δ(q,b) = q

Lenguaje aceptado por un AFD

- **Configuración:**

  (q, w) $\in Q \times \Sigma^*$

  q es el estado actual y **w** la palabra que queda por leer

- **Movimiento:** cambio de configuración en un paso ($s \in \Sigma$)

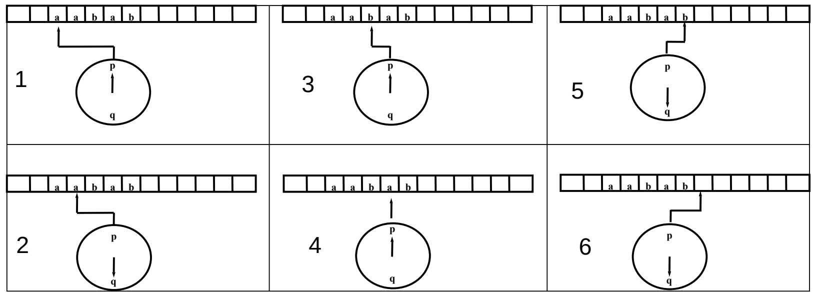
  (p, s.w) $\vdash$ (q, w) si y solo si $\delta(p,s) = q$

- **Lenguaje aceptado por el autómata M:**

  L(M) = { $\mathbf{w} \in \Sigma^*$: $(\mathbf{q_0},\mathbf{w}) \vdash^* (p,\varepsilon) \wedge p \in \mathbf{F}$ }

  ($\vdash^*$ denota 0 ó más pasos de $\vdash$)

Ejemplo de AFD



(p, aabab) $\vdash$ (q, abab) $\vdash$ (p, bab) $\vdash$ (p, ab) $\vdash$ (q, b) $\vdash$ (q,ε)

$\delta(p,a) = q$ $\delta(q,a) = p$ $\delta(p,b) = p$ $\delta(p,a) = q$ $\delta(q,b) = q$

Función de transición extendida

Función de transición extendida a palabras

δ^*: $Q \times \Sigma^* \longrightarrow Q$

- $\forall w \in \Sigma^*, \forall s \in \Sigma, \forall q \in Q$

  $\delta^*(\mathbf{q}, \mathbf{w})$ = estado en que se encuentra M tras leer la palabra **w** desde **q**

  - $\delta^*(q, \varepsilon) = q$
  - $\delta^*(q, ws) = \delta(\delta^*(q, w), s)$

- **Lenguaje aceptado por M:**

  L(M) = { $w \in \Sigma^*$: $\delta^*(q_0, w) \in F$ }

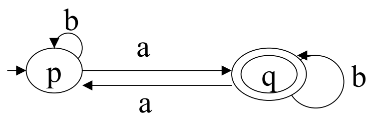
Ejemplo de lenguaje aceptado

Dado un AFD **M = (Q, Σ, δ, q_0, F)**, el lenguaje que acepta es

L(M) = { $w \in \Sigma^*$: $(q_0,w) \vdash^* (p, \varepsilon) \wedge p \in F$ }

= { $w \in \Sigma^*$: $\delta^*(q_0, w) \in F$ }

Para el AFD **M = ({p,q}, {a,b}, δ, p, {q})**



L(M) = { a, bab, aaa, aaba, aabba, abbbabab,.... }

= { $\mathbf{w} \in \{a,b\}^*$: el número de **a**'s en **w** es impar }

Autómata finito determinista totalmente especificado (AFDt)

- **Autómata totalmente especificado AFDt**

  Es un AFD cuya función de transición es <u>total</u>

  M = (Q, Σ, δ, q_0, F) con **δ : Q×Σ ⟶ Q**

- **Autómatas equivalentes**

  M y **N** son equivalentes si **L(M) = L(N)**

PROPOSICIÓN 1:

Los AFDts y los AFDs son equivalentes: L(AFDt) = L(AFD)

Dem: [⊆] Todo AFDt es un AFD

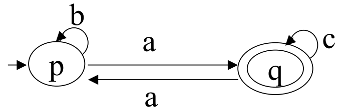
[⊇] Para cada AFD existe un AFDt (con un estado "trampa") equivalente.

Ejemplo: AFD a AFDt

AFD **M = (Q, Σ, δ, q_0, F)** con **δ : Q × Σ ⟶ Q** f. parcial

δ(p,c) = indefinido

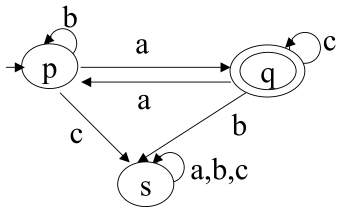
δ(q,b) = indefinido



AFDt **N = (Q∪{s}, Σ, δ', q_0, F)** con **δ' : Q∪{s} × Σ ⟶ Q∪{s}** f. total

tal que

L(M) = L(N)



Autómata finito no determinista (AFND)

- **Definición formal:**

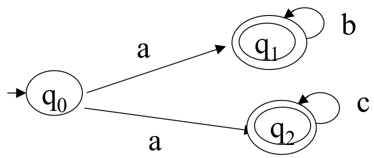
  M = (Q, Σ, δ, q0, F) con **δ : Q × Σ → ℘(Q)**

- **Función de transición extendida:** **δ*** : Q × Σ* → ℘(Q)

$$\delta^*(q,\varepsilon)=\{q\} \quad \text{y} \quad \delta^*(q,ws)=\bigcup_{p\in\delta^*(q,w)}\delta(p,s)$$

- **Configuración:** **(q, w)** ∈ Q × Σ *
- **Movimiento:** **(p, s.w) |— (q, w)** si y solo si q ∈ δ(p, s)
- **Lenguaje aceptado:** **L(M)** = { w∈Σ*: δ *(q_0, w) ∩ F ≠∅ }

Ejemplo de AFND



δ(q_0, a) = {q_1, q_2}	δ(q_0, b) = ∅	δ(q_0, c) = ∅
δ(q_1, a) = ∅	δ(q_1, b) = {q1}	δ(q_1, c) = ∅
δ(q_2, a) = ∅	δ(q_2, b) = ∅	δ(q_2, c) = {q_2}

Relación entre AFND y AFD

PROPOSICIÓN 2: L(AFD) $\subseteq$ L(AFND)

Los lenguajes reconocidos por los autómatas finitos deterministas son reconocidos por los autómatas finitos no deterministas.

DEMOSTRACIÓN:

- Dado un AFD M = (Q, Σ, δ, q_0, F), construimos el AFND
  M' = (Q, Σ, γ, q_0, F) con γ (q,s) = { δ(q,s) } $\forall q \in Q$, $\forall s \in \Sigma$
- Los autómatas M y M' son equivalentes: L(M) = L(M')

Pasos de la demostración:

1) $\gamma^*(q,x) = \{ \delta^*(q,x) \}$ $\forall q \in Q$, $\forall x \in \Sigma^*$

2) $x \in L(M) \Leftrightarrow x \in L(M')$

Gramáticas regulares a la derecha (GRD)

- **Elementos:**

Componentes	Componentes formales	
Categorias	N	conjunto de no terminales
Fuente de entrada	Σ	alfabeto de terminales

- **Elementos distinguidos para la inicialización: S∈N** símbolo inicial
- **Ciclo-generativo**:
  Búsqueda de subpalabra: lado izquierdo de una regla
  Acciones: sustituir por lado derecho.
- **Definición formal: G = (N, ∑, S, P)** con **N∩Σ=∅** y **S∈N**
  con **P (conjunto de reglas o producciones)** de la forma:

  $$\mathbf{A \rightarrow aB}$$
  $$\mathbf{A \rightarrow \varepsilon} \qquad \text{con } \mathbf{A,B} \in \mathbf{N},\ \mathbf{a} \in \Sigma$$

Gramáticas regulares a la derecha (GRD)

- **Derivación:**

$$\delta_1 \Rightarrow \delta_2 \text{ si y sólo si } \delta_1 = \sigma_1 \mathrm{A} \sigma_2,\ \delta_2 = \sigma_1 \beta \sigma_2 \text{ y } \mathrm{A} \rightarrow \beta \in \mathrm{P}$$

- **Forma sentencial:**

$$\alpha \in (\Sigma \cup N)^* \text{ tal que } S \overset{*}{\underset{G}{\Rightarrow}} \alpha \quad (\text{donde } \overset{*}{\Rightarrow} \text{ denota } 0 \text{ ó más pasos de } \Rightarrow)$$

- **Lenguaje generado por G:**

$$L(G) = \{ w \in \Sigma^* : S \overset{*}{\underset{G}{\Rightarrow}} w \}$$

Ejemplo de GRD

G = (N, ∑, S, P), con **N** = {S,A,B}, **∑** = {a,b,c} y siendo **P** el conjunto de reglas:

$$\mathrm{S} \rightarrow \mathbf{a}\mathrm{A} \mid \mathbf{a}\mathrm{B} \qquad \mathrm{A} \rightarrow \mathbf{b}\mathrm{A} \mid \boldsymbol{\varepsilon} \qquad \mathrm{B} \rightarrow \mathbf{c}\mathrm{B} \mid \boldsymbol{\varepsilon}$$

Ejemplos de derivaciones:

$$\mathrm{S} \Rightarrow \mathbf{a}\mathrm{A} \Rightarrow \mathbf{ab}\mathrm{A} \Rightarrow \mathbf{abb}\mathrm{A} \Rightarrow \mathbf{abb}$$

$$\mathrm{S} \Rightarrow \mathbf{a}\mathrm{B} \Rightarrow \mathbf{ac}\mathrm{B} \Rightarrow \mathbf{ac}$$

¿Lenguaje generado por G?

Gramáticas lineales a la derecha (GLD)

- **Definición formal: G = (N, Σ, S, P)** con $N \cap \Sigma = \emptyset$ y $S \in N$
  donde el conjunto **P de reglas o producciones** es de la forma:
  - **$A \to wB$**
  - **$A \to u$** con **$A,B \in N$**, **$w \in \Sigma^+$**, **$u \in \Sigma^*$**

PROPOSICIÓN 3:
Las GRDs y las GLDs son equivalentes: L(GRD) = L(GLD)

Dem: [$\subseteq$] Toda GRD es una GLD

[$\supseteq$] Para cada GLD se construye una GRD equivalente, mediante el siguiente algoritmo:

Relación entre GRD y GLD

Algoritmo que transforma una GLD en una GRD equivalente:

ENTRADA: GLD $G = (N, \Sigma, A_0, P)$

SALIDA: GRD $G' = (N \cup N', \Sigma, A_0, P')$

- Para cada **$A \to awB$** (ó equiv. **$A \to aw$**) con **$w \neq \varepsilon$** se crea un símbolo no terminal $Z \in N'$ y cambiamos la regla por 2 reglas:
  $A \to aZ$ y **$Z \to wB$** (ó equiv. **$Z \to w$**)
- Para cada **$A \to a$** con **$a \in \Sigma$** se crea un símbolo no terminal $Z \in N'$ y cambiamos la regla por 2 reglas:
  $A \to aZ$ y **$Z \to \varepsilon$**
- Continuar el proceso mientras haya reglas que reemplazar.

Ejemplo: GLD a GRD

Ejemplo de transformación de una GLD en una GRD equivalente:

ENTRADA: GLD G = ({A,B}, {a,b,c,d}, A, P) siendo P:

$A \to abcB$ **$B \to d \mid \varepsilon$**

SALIDA: GRD G' = ({A,B,C,D,E}, {a,b,c,d}, A, P') siendo P':

$A \to aC$ ~~**$C \to bcB$**~~

$C \to bD$ **$D \to cB$**

$B \to dE \mid \varepsilon$ **$E \to \varepsilon$**

Relación entre GRD y AFD

PROPOSICIÓN 4: L(AFD) $\subseteq$ L(GRD)

Los lenguajes reconocidos por los autómatas finitos deterministas son generados por las gramáticas regulares a la derecha.

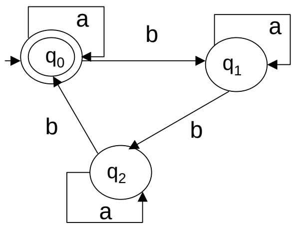
DEMOSTRACIÓN:

1) Crear un algoritmo que produzca una GRD equivalente a un AFD M:
   - ENTRADA: AFD $M = (Q, \Sigma, \delta, q_0, F)$ con $Q = \{q_0, q_1, ..., q_m\}$.
   - SALIDA: GRD $G = (N, \Sigma, A_0, P)$
2) Probar que M y G son equivalentes: L(M)=L(G).

1) Construcción del algoritmo

- Sea **N = { A_0, A_1, ..., A_m }** donde **A_i** se corresponde con el estado **q_i**
- Sea **P = { $A_i \rightarrow sA_j$: $\delta(q_i, s) = q_j$ } $\cup$ { $A_i \rightarrow \varepsilon$: $q_i \in F$ }**
- La salida es G = (N, Σ, A_0, P)

Ejemplo:



Gramática obtenida:

$A_0 \rightarrow aA_0 \mid bA_1 \mid \varepsilon$

$A_1 \rightarrow aA_1 \mid bA_2$

$A_2 \rightarrow aA_2 \mid bA_0$

2) Demostración formal de la equivalencia

a) Demostramos por inducción sobre k≥0 la siguiente propiedad:

$$(q_i, x) \vdash^{k} (q_j, \varepsilon) \Leftrightarrow A_i \Rightarrow^k xA_j \qquad (x \in \Sigma^*)$$

b) A partir de la propiedad anterior podemos demostrar la equivalencia:

$x \in L(M) \Leftrightarrow (q_0, x) \vdash^{k} (q_f, \varepsilon)$ con $q_f \in F \Leftrightarrow$

$A_0 \Rightarrow^k xA_f$ y $A_f \rightarrow \varepsilon \in P \Leftrightarrow A_0 \Rightarrow^k xA_f \Rightarrow x \Leftrightarrow$

$A_0 \Rightarrow^{k+1} x \Leftrightarrow x \in L(G)$

Relación entre GRD y AFND

PROPOSICIÓN 5: L(AFND) = L(GRD)

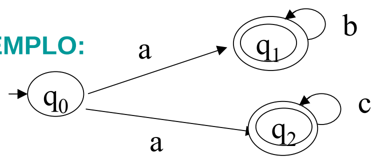
Los lenguajes reconocidos por los autómatas finitos no deterministas son generados por gramáticas regulares a la derecha. Y viceversa.

DEMOSTRACIÓN:

L(AFND) $\subseteq$ L(GRD) Similar a la demostración de la Proposición 2.

L(GRD) $\subseteq$ L(AFND) Dar la vuelta a la demostración anterior.

EJEMPLO:



$S \rightarrow \mathbf{a}A \mid \mathbf{a}B$

$A \rightarrow \mathbf{b}A \mid \varepsilon$

$B \rightarrow \mathbf{c}B \mid \varepsilon$

Relaciones estudiadas

Repaso de las proposiciones vistas hasta ahora:

L(AFDt) $=_{(1)}$ L(AFD) $\subseteq_{(2)}$ L(AFND) $=_{(5)}$ L(GRD) $=_{(3)}$ L(GLD)

$\subseteq_{(4)}$