# Autômatos e computabilidade

## Propriedades de Linguagens Regulares 1

Pedro A D Rezende
UnB – IE – CIC

# A & C: Linguagens Regulares

Lema do bombeamento (AH L3.1: *pumping lemma*)

$$L \in \mathcal{L}_{\text{Reg}} \Rightarrow \exists n \in \mathbb{N} \mid \forall \alpha \in L \; [\, |\alpha| \geq n \Rightarrow \exists u,v,w \in \Sigma^* \mid$$

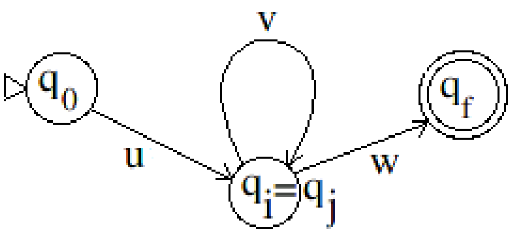
$$0 < |v| \leq n \wedge \alpha = uvw \wedge \forall k \in \mathbb{N} \; [uv^k w \in L] \,]$$

# A & C: Linguagens Regulares

Lema do bombeamento (AH L3.1: *pumping lemma*)

$L \in \mathcal{L}_{Reg} \Rightarrow \exists n \in \mathbb{N} \mid \forall \alpha \in L \; [\, |\alpha| \geq n \Rightarrow \exists u,v,w \in \Sigma^* \mid$

$0 < |v| \leq n \wedge \alpha = uvw \wedge \forall k \in \mathbb{N} \; [uv^k w \in L] \,]$




$\delta^*(q_0,u) = q_i \; ; \quad \delta^*(q_i,v) = q_j = q_i \; ; \quad \delta^*(q_j,w) = q_f$

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{\text{Reg}}$ (HU T3.1-4)

1. $\mathcal{L}_{\text{reg}}$ é fechada em relação a $\cup$, $\circ$, $^*$

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{\text{Reg}}$ (HU T3.1-4)

1. $\mathcal{L}_{\text{reg}}$ é fechada em relação a $\cup$, $\circ$, $^*$
2. Complemento:

$$L \in \mathcal{L}_{\text{Reg}} \Rightarrow \Sigma^* - L = {}^{c}L \in \mathcal{L}_{\text{Reg}}$$

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{Reg}$ (HU T3.1-4)

1. $\mathcal{L}_{reg}$ é fechada em relação a $\cup$, $\circ$, $^*$
2. Complemento:

$$L \in \mathcal{L}_{Reg} \Rightarrow \Sigma^* - L = {}^cL \in \mathcal{L}_{Reg} \; ({}^cL = L(Q, \Sigma, q_0, \delta, {}^cF))$$

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{Reg}$ (HU T3.1-4)

1. $\mathcal{L}_{reg}$ é fechada em relação a $\cup$, $\circ$, $^*$
2. Complemento:

   $L \in \mathcal{L}_{Reg} \Rightarrow \Sigma^* - L = {}^cL \in \mathcal{L}_{Reg}$ $({}^cL = L(Q, \Sigma, q_0, \delta, {}^cF))$
3. $\cap$:

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{\text{Reg}}$ (HU T3.1-4)

1. $\mathcal{L}_{\text{reg}}$ é fechada em relação a $\cup$, $\circ$, $^*$
2. Complemento:

   $L \in \mathcal{L}_{\text{Reg}} \Rightarrow \Sigma^* - L = {}^{c}L \in \mathcal{L}_{\text{Reg}}$ $({}^{c}L = L(Q, \Sigma, q_0, \delta, {}^{c}F))$
3. $\cap$: $L \cap L' = {}^{c}({}^{c}L \cup {}^{c}L')$ (ou $L(M \times M')$)

# A & C: Linguagens Regulares

Propriedades de fechamento de $\mathcal{L}_{Reg}$ (HU T3.1-4)

1. $\mathcal{L}_{reg}$ é fechada em relação a $\cup$, $\circ$, $^*$
2. Complemento:

   $L \in \mathcal{L}_{Reg} \Rightarrow \Sigma^* - L = {}^cL \in \mathcal{L}_{Reg}$ $({}^cL = L(Q, \Sigma, q_0, \delta, {}^cF))$
3. $\cap$: $L \cap L' = {}^c({}^cL \cup {}^cL')$ (ou $L(M \times M')$)
4. Substituição:

   $a \to L_a$ em cadeias de L, $a \in \Sigma$, $L \subseteq \Sigma^*$, $L_a \in \mathcal{L}_{Reg}$

# A & C: Linguagens Regulares

## Congruências sintáticas em $\mathcal{L}_{Reg}$ (HU T3.9-10)

Um DFA M = $(Q, \Sigma, q_0, \delta, F)$ define $\equiv_M$ sobre $\Sigma^*$:

$$u \equiv_M v \Leftrightarrow_{def} \delta^*(q_0,u) = \delta^*(q_0,v)$$

# A & C: Linguagens Regulares

Congruências sintáticas em $\mathcal{L}_{Reg}$ (HU T3.9-10)

Um DFA M = (Q, $\Sigma$, $q_0$, $\delta$, F) define $\equiv_M$ sobre $\Sigma^*$:

$$u \equiv_M v \Leftrightarrow_{def} \delta^*(q_0,u) = \delta^*(q_0,v)$$

$\equiv_M$ é rel. de equivalência e congruêcia à direita:

$$u \equiv_M v \Rightarrow \forall z \in \Sigma^* [uz \equiv_M vz]$$

# A & C: Linguagens Regulares

Congruências sintáticas em $\mathcal{L}_{Reg}$ (HU T3.9-10)

Um DFA M = (Q, $\Sigma$, $q_0$, $\delta$, F) define $\equiv_M$ sobre $\Sigma^*$:

$$u \equiv_M v \Leftrightarrow_{def} \delta^*(q_0,u) = \delta^*(q_0,v)$$

$\equiv_M$ é rel. de equivalência e congruêcia à direita:

$$u \equiv_M v \Rightarrow \forall z \in \Sigma^* [uz \equiv_M vz]$$

Uma $L \in \mathcal{L}_{Reg}$ define $\equiv_L$ sobre $\Sigma^*$:

$$u \equiv_L v \Leftrightarrow_{def} \forall z \in \Sigma^* [uz \in L \Leftrightarrow vz \in L]$$

# A & C: Linguagens Regulares

**Teorema Mihill-Nerode** (HU T3.9)

1. L $\in \mathcal{L}_{\text{Reg}} \Leftrightarrow$
2. L = $\cup$ classes de alguma congruência à direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 1. $\Rightarrow$ 2.

$$\exists\, M \mid L = L(M) \wedge L = \bigcup_{\delta^*(q_0,v)\in F} [v]_{\equiv_M}$$

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. L $\in \mathcal{L}_{\text{Reg}}$ $\Leftrightarrow$
2. L = $\cup$ classes de alguma congruência à Direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. $L \in \mathcal{L}_{Reg} \Leftrightarrow$
2. $L = \cup$ classes de alguma congruência à Direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 2. $\Rightarrow$ 3.

Dado R congruência-D de índice finito, $L = \bigcup_{1 \leq i \leq n} [v_i]_R$ ,

$u \; R \; v \Rightarrow uz \; R \; vz$

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. $L \in \mathcal{L}_{Reg} \Leftrightarrow$
2. $L = \cup$ classes de alguma congruência à Direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 2. $\Rightarrow$ 3.

Dado R congruência-D de índice finito, $L = \bigcup_{1 \leq i \leq n} [v_i]_R$ ,

$u \; R \; v \Rightarrow uz \; R \; vz \Rightarrow [uz \in L \Leftrightarrow vz \in L]$

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. $L \in \mathcal{L}_{Reg} \Leftrightarrow$
2. $L = \cup$ classes de alguma congruência à Direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 2. $\Rightarrow$ 3.

Dado R congruência-D de índice finito, $L = \bigcup_{1 \leq i \leq n} [v_i]_R$ ,

$u \; R \; v \Rightarrow uz \; R \; vz \Rightarrow [uz \in L \Leftrightarrow vz \in L] \Rightarrow u \equiv_L v$

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. $L \in \mathcal{L}_{\text{Reg}} \Leftrightarrow$
2. $L = \cup$ classes de alguma congruência à Direita de índice $(|\{$classes de equivalência$\}|)$ finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 3. $\Rightarrow$ 1.

Seja $M = (Q=\{[v]_{\equiv_L}\}, \Sigma, q_0=[\lambda]_{\equiv_L}, \delta, F=\{[v]_{\equiv_L} | v \in L\})$

onde $\delta([v]_{\equiv_L}, a) = [va]_{\equiv_L}$.

# A & C: Linguagens Regulares

Teorema Mihill-Nerode (HU T3.9)

1. L $\in \mathcal{L}_{\text{Reg}}$ $\Leftrightarrow$
2. L = $\cup$ classes de alguma congruência à Direita de índice ($|\{$classes de equivalência$\}|$) finito $\Leftrightarrow$
3. $\equiv_L$ tem ínidice finito

Demonstração 3. $\Rightarrow$ 1.

Seja M = $(Q=\{[v]_{\equiv_L}\}, \Sigma, q_0=[\lambda]_{\equiv_L}, \delta, F=\{[v]_{\equiv_L} | v \in L\})$

onde $\delta([v]_{\equiv_L}, a) = [va]_{\equiv_L}$. Então L = L(M)

# A & C: Linguagens Regulares

Automaton minimal relativo a $|Q|$ (HU T3.9)

1. $\forall L \in \mathcal{L}_{\text{Reg}} \exists M_L \in \mathcal{M}_{\text{DFA}} | \forall M' \in \mathcal{M}_{\text{DFA}} [L = L(M') \Rightarrow |Q'| \geq |Q_L|]$
2. $M_L$ é o único minimal a menos de isomorfismo

# A & C: Linguagens Regulares

Automaton minimal relativo a $|Q|$ (HU T3.9)

1. $\forall L \in \mathcal{L}_{Reg} \exists M_L \in \mathcal{M}_{DFA} | \forall M' \in \mathcal{M}_{DFA} [L = L(M') \Rightarrow |Q'| \geq |Q_L|]$
2. $M_L$ é o único minimal a menos de isomorfismo

Demonstração:

$$M_L = (Q = \{[v]_{\equiv_L}\}, \Sigma, q_0 = [\lambda]_{\equiv_L}, \delta, F = \{[v]_{\equiv_L} | v \in L\})$$

$$\delta([v]_{\equiv_L}, a) = [va]_{\equiv_L}$$

l

# A & C: Linguagens Regulares

Automaton minimal relativo a $|Q|$ (HU T3.9)

1. $\forall L \in \mathcal{L}_{Reg} \exists M_L \in \mathcal{M}_{DFA} | \forall M' \in \mathcal{M}_{DFA} [L = L(M') \Rightarrow |Q'| \geq |Q_L|]$
2. $M_L$ é o único minimal a menos de isomorfismo

Demonstração:

$$M_L = (Q = \{[v]_{\equiv_L}\}, \Sigma, q_0 = [\lambda]_{\equiv_L}, \delta, F = \{[v]_{\equiv_L} | v \in L\})$$

$$\delta([v]_{\equiv_L}, a) = [va]_{\equiv_L}$$

Isomorfismo: Por Mihill-Nerode 2.$\Rightarrow$.3,

se $|Q'| = |Q_L|$ então $\equiv_{M'} = \equiv_L$

# A & C: Linguagens Regulares

## Algoritmo de Minimalização de FAs (HU T3.11)

Seja $\equiv_Q$ rel. de equivalência sobre Q dada por

$$p \equiv_Q q \Leftrightarrow_{def} \forall u \in \Sigma^* [\delta^*(p,u) \in F \Leftrightarrow \delta^*(q,u) \in F]$$

Então $M_L = (Q/\equiv_Q, \Sigma, [q_0], ([q],a) \rightarrow [\delta(q,a)], F/\equiv_Q)$

Demonstração (algoritmo):

Idéia: Refinamentos sucessivos para calcular $Q/\equiv_Q$

São marcados, recursivamente, pares de estados

distinguíveis, isto é, pares (p,q) para os quais $[p] \neq [q]$.

Inicialização: Marque os pares de $F \times {}^cF$

# A & C: Linguagens Regulares

## Algoritmo de Minimalização de DFAs (completos)

$\forall$ (p,q)$\in$ F$\times$F $\cup$ $^c$F$\times$$^c$F (pares inicialmente não marcados) faça {

Se $\forall a\in\Sigma$ [($\delta$(p,a),$\delta$(q,a)) não está marcado 'distinguiveis']

então {

$\forall a\in\Sigma$ faça {

Se $\delta(p,a)\neq\delta(q,a)$ então {

lista dos distinguíveis via ($\delta$(p,a),$\delta$(q,a))$\leftarrow$ (p,q)

}

}

}

senão { /* $\exists a\in\Sigma$ t.q. [$\delta$(p,a),$\delta$(q,a)) está marcado] */

marque os pares da lista dos distinguíveis via (p,q) ;

marque recursivamente os pares das listas destes

}

}