

Algoritmo de McNaughton-Yamada-Thompson

Alan Reyes-Figueroa
Teoría de la Computación

(Aula 06a) 31.julio.2024

Equivalencia de AFNs y
expresiones regulares (Parte 1)

Equivalencia entre AFDs y AFNs

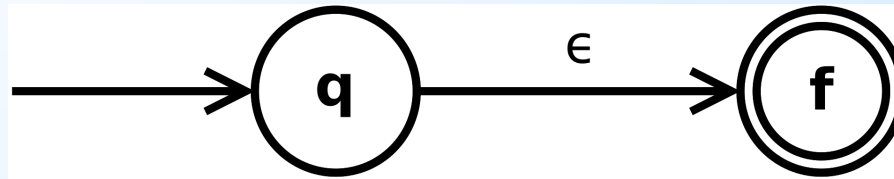
- ◆ Los autómatas finitos no-deterministas (AFNs) son, en apariencia, más generales que los AFD.
- ◆ 1) Todo AFD es un AFN.
 $\text{AFD} \subset \text{AFN}$
- ◆ 2) Vamos a mostrar que para cada AFN digamos M , existe un AFD equivalente M' , esto es, M' que acepta el mismo lenguaje que M : $L(M) = L(M')$.
Esto de alguna manera indica que $\text{AFN} \subset \text{AFD}$,
- ◆ Portanto, $\text{AFN} = \text{AFD}$.

Equivalencia entre AFDNs y expresiones regulares

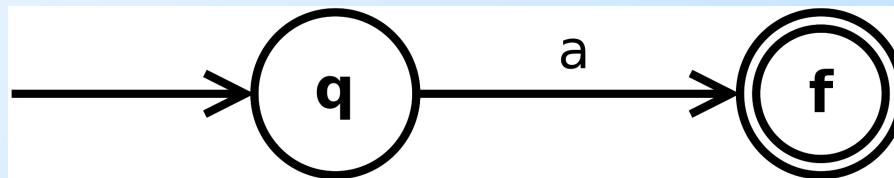
- ◆ Vamos a mostrar en las próximas clases que hay una correspondencia entre autómatas finitos y lenguajes regulares.
 1. para cada lenguaje regular L , hay un AFD M que genera L , esto es $L(M) = L$.
 2. para cada AFD, $L(M)$ es un lenguaje regular.
- ◆ 1) Específicamente, hoy vamos a mostrar que para cada expresión regular α , existe un autómata finito no-determinista que genera exactamente $L(\alpha)$.
- ◆ La construcción es vía el algoritmo de McNaughton-Yamada-Thompson.

Representación de expresiones regulares mediante autómatas

- ◆ Convertimos la regexp ϵ al autómata

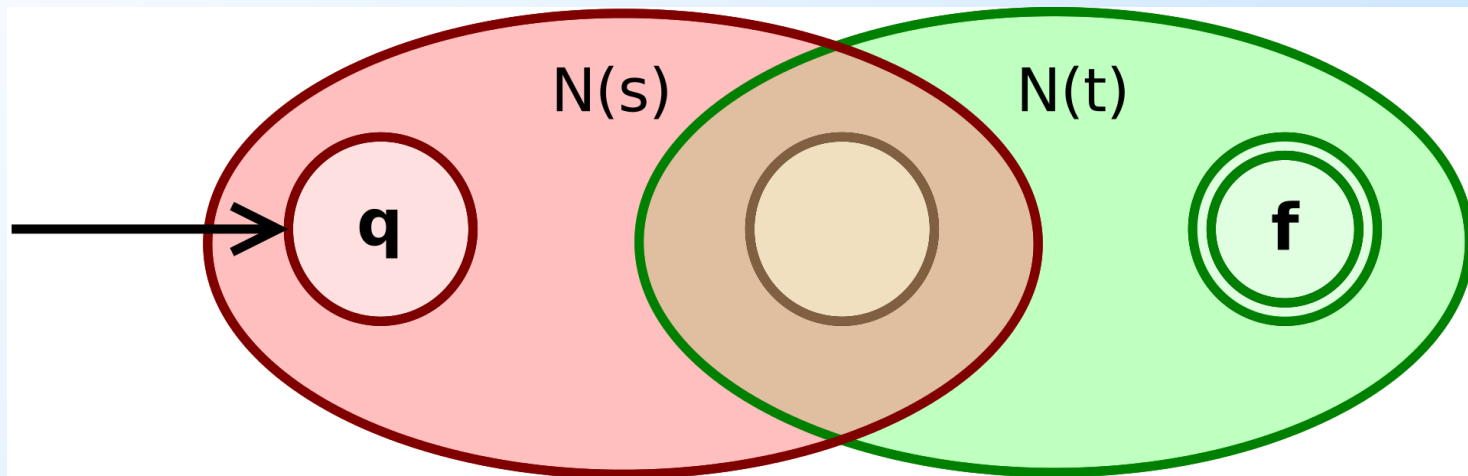


- ◆ Para cada $a \in \Sigma$, convertimos la regex a al autómata



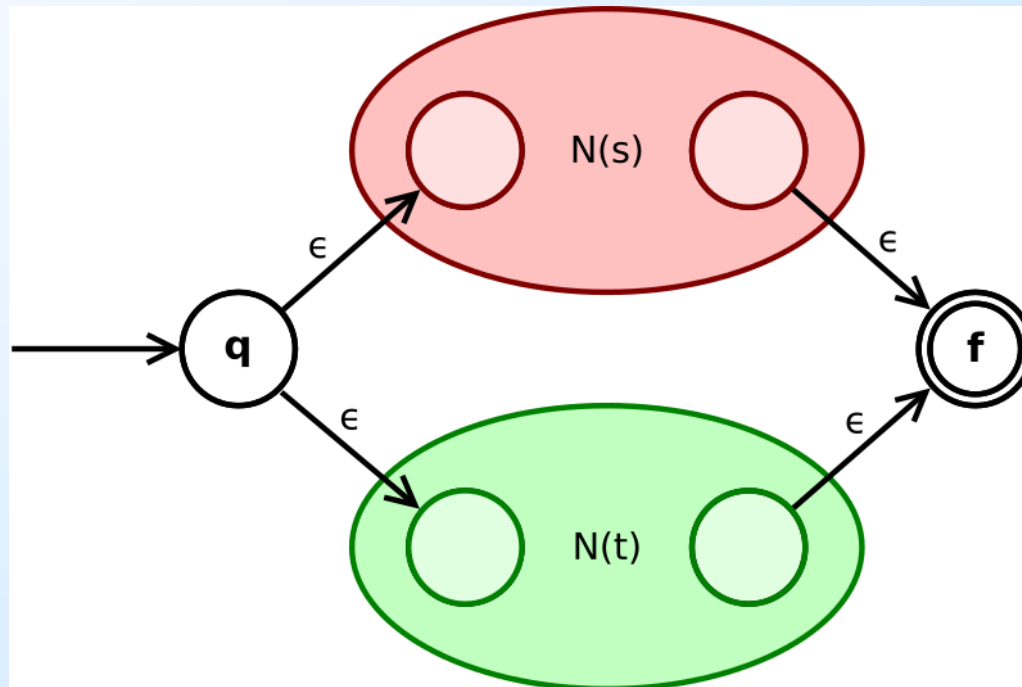
Representación de expresiones regulares mediante autómatas

- ◆ Si s, t son regexp, convertimos la concatenación st al autómata



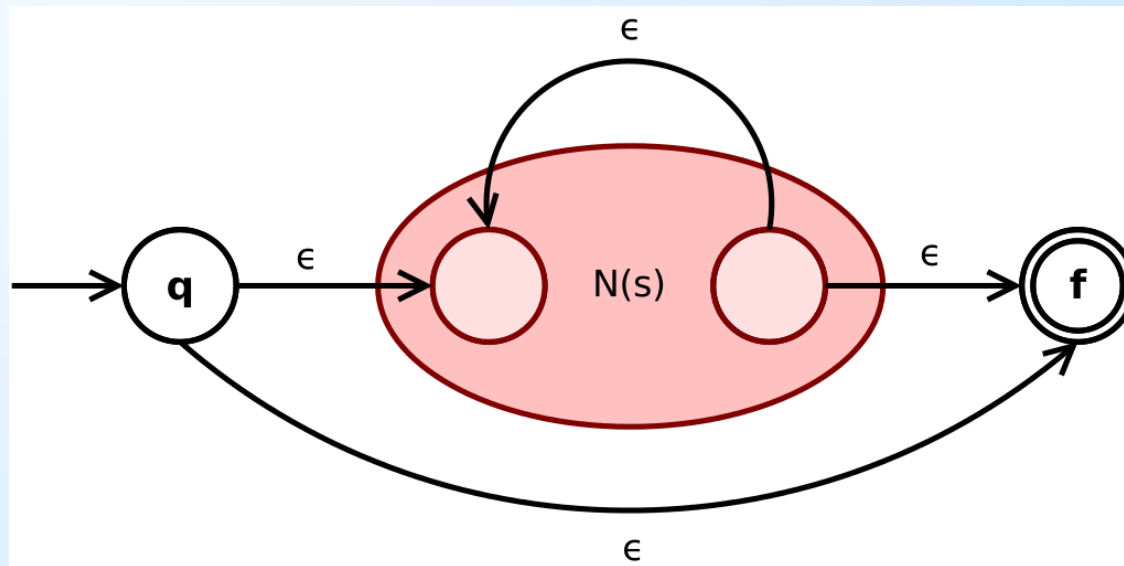
Representación de expresiones regulares mediante autómatas

- ◆ Si s, t son regexp, convertimos la unión $s \mid t$ al autómata



Representación de expresiones regulares mediante autómatas

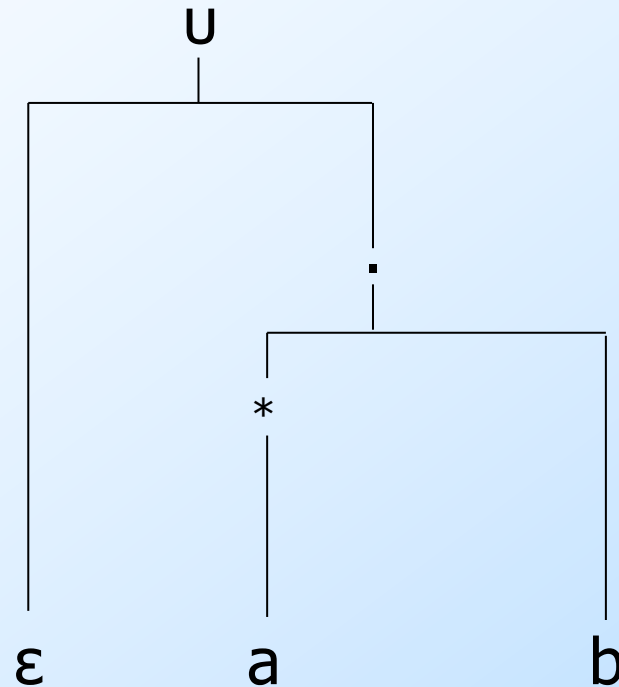
- ◆ Si s es regexp, convertimos la cerradura de Kleene s^* al autómata



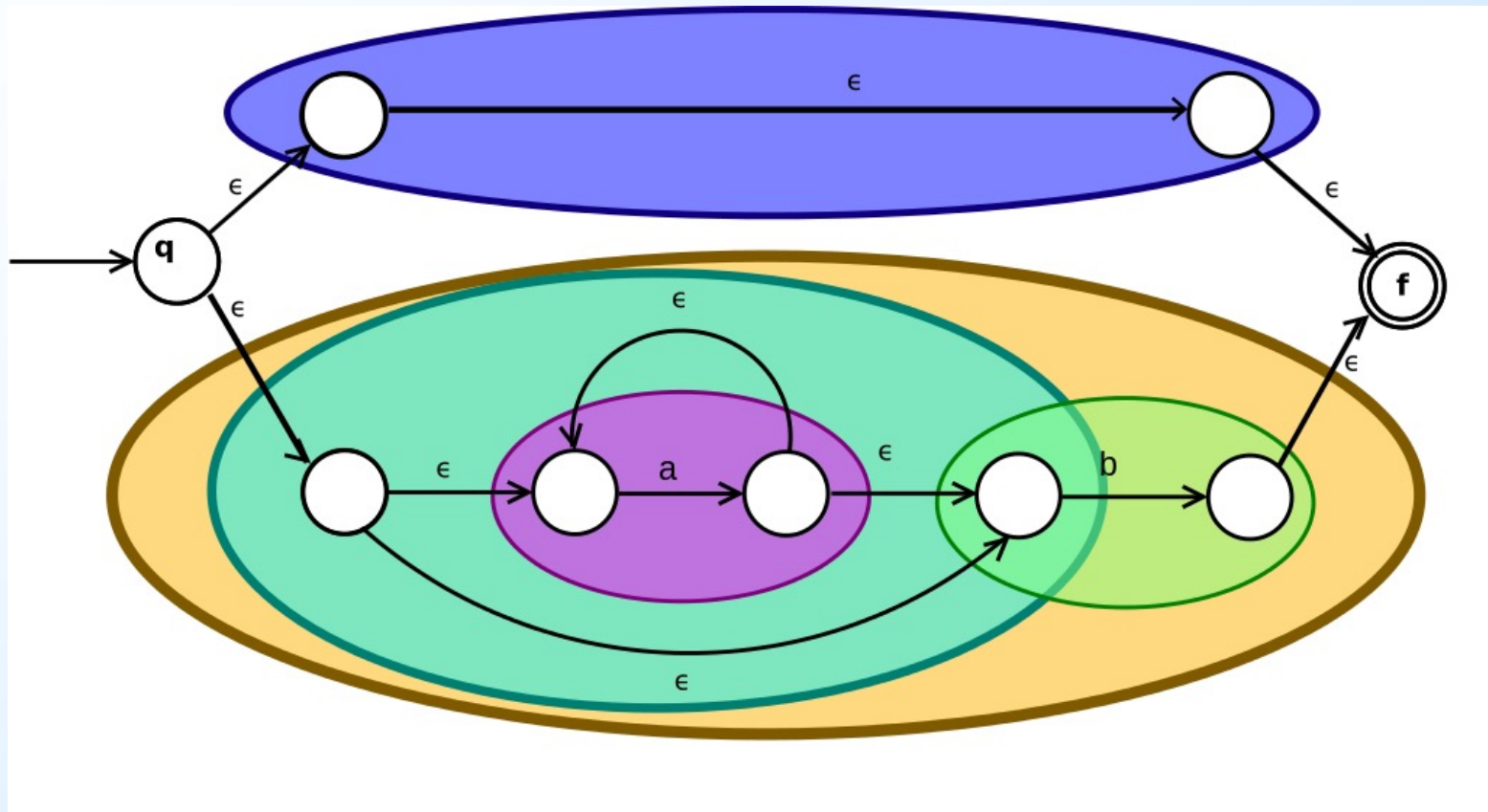
Pasos para construir el AFD

- ◆ 1) Construir el árbol de jerarquía de la expresión regular:
 - las hojas del árbol son los símbolos base (ϵ y $a \in \Sigma$)
 - Cada vez que un operador aplica a una expresión, dicho operador genera un nodo en el árbol.
- ◆ 2) Sobre cada nodo del árbol, comenzando desde las hojas hacia arriba hasta la raíz del árbol (bottom-up), convertimos cada expresión a su autómata.
- ◆ 3) Pegamos todas las partes en un sólo AFN.

Ejemplo: $w = (\epsilon|a^*b)$



Ejemplo:



Ejemplo: $a = (0|(1(01^*(00)^*0)^*1)^*)$.

