Equivalencia y Minimización de Autómatas Finitos Deterministas

Matemáticas Computacionales (TC2020)

M.C. Xavier Sánchez Díaz
sax@itesm.mx

Tabla de contenidos

Definición de Equivalencia

Equivalencia de AFDs

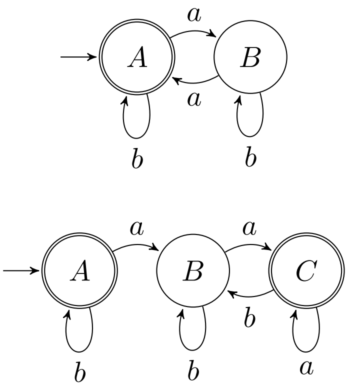
Definición 1

Dos autómatas M_1 y M_2 son equivalentes, $M_1 \equiv M_2$, cuando aceptan exactamente el mismo lenguaje.

Equivalencia

Equivalencia de AFDs

¿Son estos dos autómatas equivalentes?



¿Cómo podemos probarlo?

Equivalencia de AFDs

Sistemáticamente, probando con las palabras de $\sigma^* = \{\varepsilon, a, b, aa, ab, \ldots\}$

¿Qué pasa si no son equivalentes? Simplemente nunca acabaremos.

Podemos probar todas las posibilidades mediante un **árbol de estados incompatibles**.

¿Cómo podemos probarlo?

Equivalencia de AFDs

Sistemáticamente, probando con las palabras de $\sigma^* = \{\varepsilon, a, b, aa, ab, \dots\}$
¿Qué pasa si no son equivalentes? Simplemente nunca acabaremos.

Podemos probar todas las posibilidades mediante un **árbol de estados incompatibles**.

¿Cómo podemos probarlo?

Equivalencia de AFDs

Sistemáticamente, probando con las palabras de $\sigma^* = \{\varepsilon, a, b, aa, ab, \ldots\}$
¿Qué pasa si no son equivalentes? Simplemente nunca acabaremos.

Podemos probar todas las posibilidades mediante un **árbol de estados incompatibles**.

¿Cómo podemos probarlo?

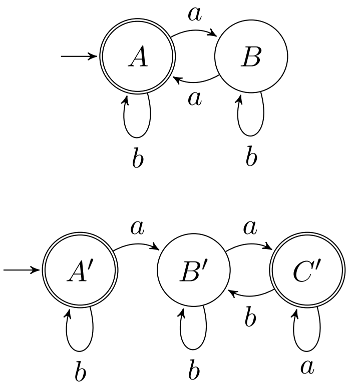
Equivalencia de AFDs

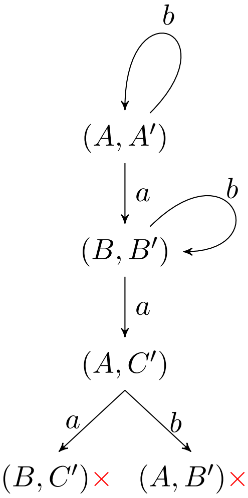
Sistemáticamente, probando con las palabras de $\sigma^* = \{\varepsilon, a, b, aa, ab, \dots\}$
¿Qué pasa si no son equivalentes? Simplemente nunca acabaremos.

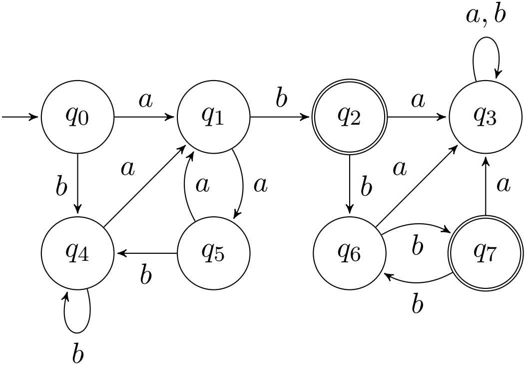
Podemos probar todas las posibilidades mediante un **árbol de estados incompatibles**.

Árbol de comparación

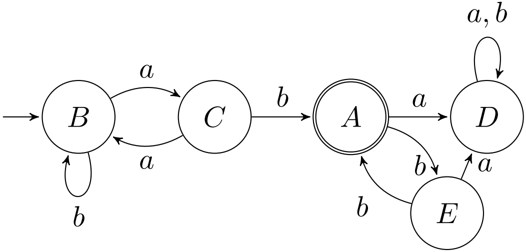
Equivalencia de AFDs







q_0
q_1
q_2
q_3
q_4
q_5
q_6
q_7
a
b
a, b



B
C
A
D
E
a
b
a, b

¿Por qué simplificar AFDs?

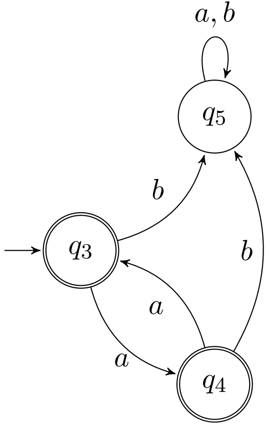
Simplificación de AFDs

Una máquina M puede tener estados redundantes.

¿Por qué simplificar AFDs?

Simplificación de AFDs

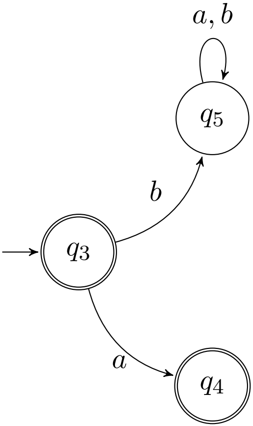
Una máquina M puede tener estados redundantes.



Eliminación de estados equivalentes

Simplificación de AFDs

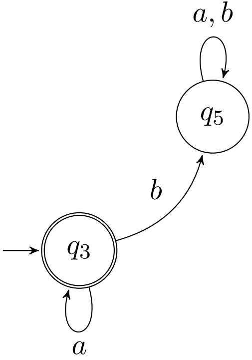
Borrar transiciones:



Eliminación de estados equivalentes

Simplificación de AFDs

Redirigir transiciones:



Deducción de estados distinguibles

Simplificación de AFDs

- Dos estados son distinguibles si son **incompatibles**: uno es final y el otro es no final.

- Si tenemos transiciones $\delta(p_0, \sigma) = p$ y $\delta(q_0, \sigma) = q$, donde p, q son distinguibles, entonces también p_0 y q_0 son distinguibles.

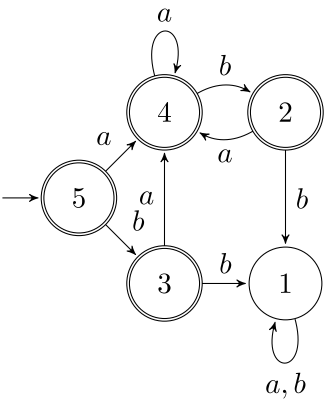
Deducción de estados distinguibles

Simplificación de AFDs

- Dos estados son distinguibles si son **incompatibles**: uno es final y el otro es no final.
- Si tenemos transiciones $\delta(p_0, \sigma) = p$ y $\delta(q_0, \sigma) = q$, donde p, q son distinguibles, entonces también p_0 y q_0 son distinguibles.

Deducción de estados distinguibles

Simplificación de AFDs



2				
3				
4				
5				
	1	2	3	4

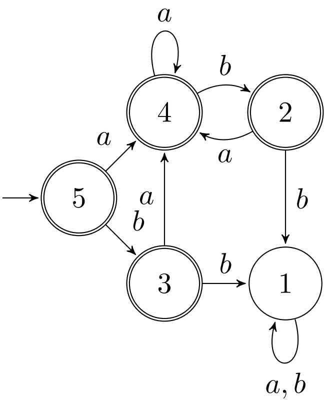
2				
3				
4				
5				
	1	2	3	4

2				
3				
4				
5				
	1	2	3	4

2				
3				
4				
5				
	1	2	3	4

Deducción de estados distinguibles

Simplificación de AFDs



2				
3				
4				
5				
	1	2	3	4

2	×			
3	×			
4	×			
5	×			
	1	2	3	4

2	×			
3	×			
4	×			
5	×	×		
	1	2	3	4

2	×			
3	×			
4	×	×	×	
5	×	×	×	
	1	2	3	4

Simplificación por clases de equivalencia

Simplificación de AFDs

Formar clases de estados de un autómata que pudieran ser equivalentes.

Al seguir examinando las clases, podremos percatarnos de si es necesario volver a **dividirlas**.

Si las clases ya no pueden dividirse más, entonces hemos encontrado el autómata más pequeño.

Simplificación por clases de equivalencia

Simplificación de AFDs

Formar clases de estados de un autómata que pudieran ser equivalentes.

Al seguir examinando las clases, podremos percatarnos de si es necesario volver a **dividirlas**.

Si las clases ya no pueden dividirse más, entonces hemos encontrado el autómata más pequeño.

Simplificación por clases de equivalencia

Simplificación de AFDs

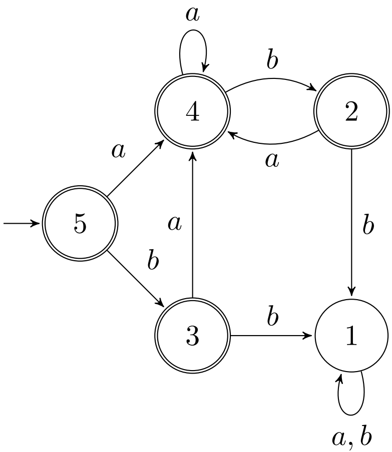
Formar clases de estados de un autómata que pudieran ser equivalentes.

Al seguir examinando las clases, podremos percatarnos de si es necesario volver a **dividirlas**.

Si las clases ya no pueden dividirse más, entonces hemos encontrado el autómata más pequeño.

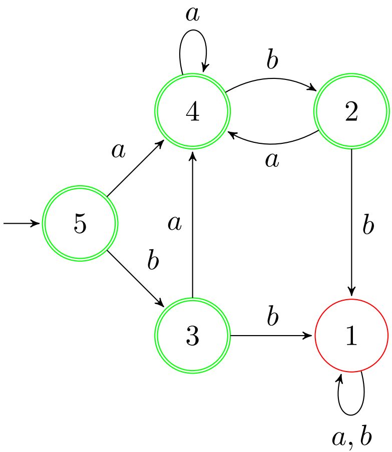
Simplificación por clases de equivalencia

Simplificación de AFDs



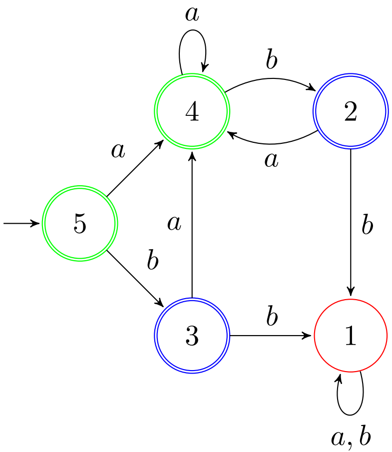
Simplificación por clases de equivalencia

Simplificación de AFDs



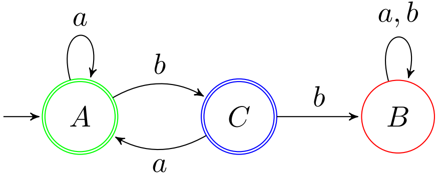
Simplificación por clases de equivalencia

Simplificación de AFDs



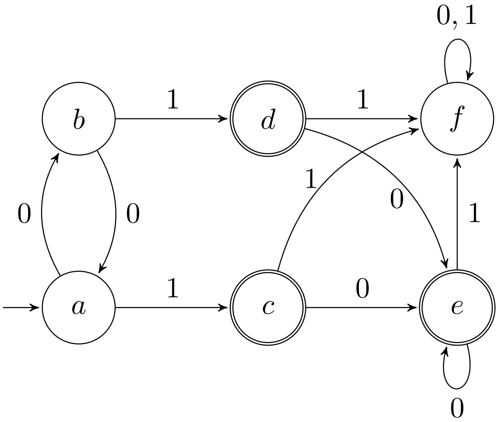
Simplificación por clases de equivalencia

Simplificación de AFDs



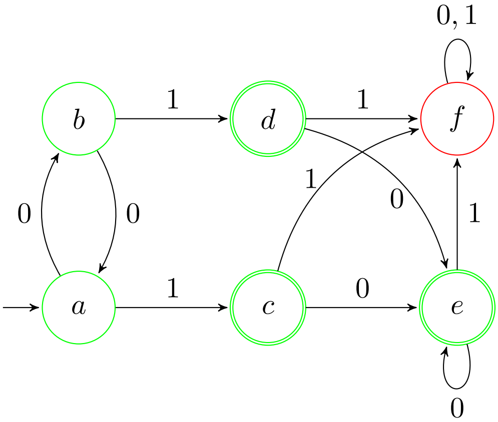
Simplificación por clases de equivalencia

Simplificación de AFDs



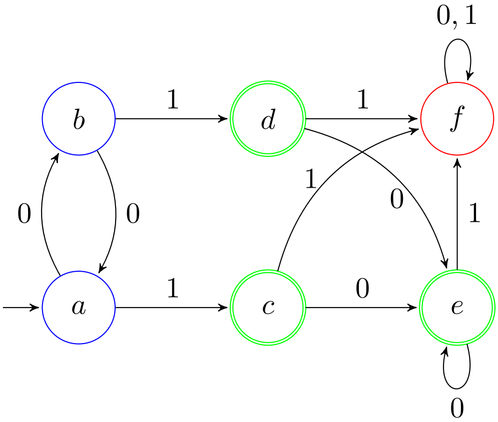
Simplificación por clases de equivalencia

Simplificación de AFDs



Simplificación por clases de equivalencia

Simplificación de AFDs



Simplificación por clases de equivalencia

Simplificación de AFDs