Propiedades de Cerradura de los Lenguajes Regulares

Alan Reyes-Figueroa

Teoría de la Computación (Aula 08) 14.agosto.2024

Unión, Intersección, Diferencia, Concatenación, Cerradura, Lenguaje Reverso

Propiedades de Cerradura

- Recordemos que una propiedad de cierre es un enunciado sobre cierta operación de lenguajes, que cuando se aplica a una clase de lenguajes L (por ejemplo, los lenguajes regulares), produce un resultado que también está en esa clase L.
- Para lenguajes regulares, usamos cualquiera de sus representaciones para mostrar una propiedad de cerradura.

Cerradura bajo la unión

- Si L y M son lenguajes regulares, también lo es $L \cup M$.
- Prueba: Sean L y M lenguajes representados por la expresiones regulares R y S, respectivamente.
- Entonces, R+S es una expresión regular cuyo lenguaje es $L \cup M$.

Cerradura de la Concatenación y de Kleene

- Prueba: La misma idea:
  - RS es una expresión regular cuyo lenguaje es LM.
  - R* es una expresión regular cuyo lenguaje es L*.

Producto de Autómatas

Sean $M_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ y

$M_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$

Autómatas AFD.

Definimos el *autómata producto* de M_1 y M_2 como el autómata

$$M_1 \times M_2 = (Q_1 \times Q_2, \Sigma, \delta_1 \times \delta_2, (q_{01}, q_{02}), G)$$

- $Q_1 \times Q_2$ es el nuevo conjunto de estados
- La función de trancisión es la función producto $\delta = \delta_1 \times \delta_2$ de las funciones δ_1 y δ_2

Producto de Autómatas

Los estados ahora son de la forma:

$q = [q_1, q_2], \quad \text{con } q_1 \in Q_1, q_2 \in Q_2$

Las transiciones ahora son de la forma

$\delta(q_1, q_2) = (\delta_1 \times \delta_2)[q_1, q_2] = [\ \delta_1(q_1), \delta_2(q_2)\ ]$

(ambas funciones actúan sobre el mismo alfabeto.)

Los estados de aceptación de $M_1 \times M_2$ se pueden definir de diferentes formas, según la utilidad del automata producto.

Cerradura de la Intersección

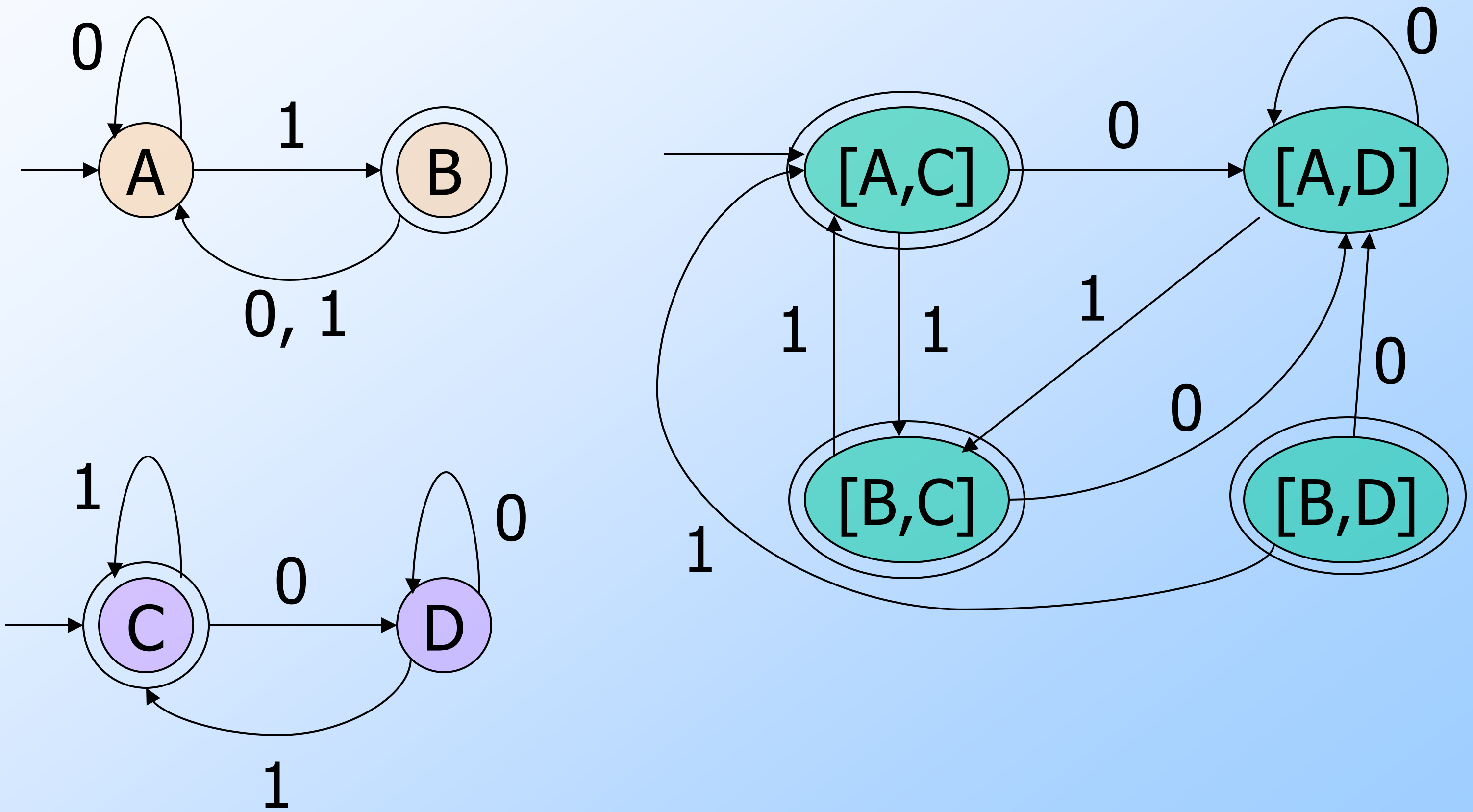
- Si L y M son lenguajes regulares, entonces también lo es $L \cap M$.
- Prueba: Sean A y B dos autómatas AFD cuyos lenguajes son L y M, resp.
- Construímos $C = A \times B$, el autómata producto de A y B.
- Hacemos los estados finales de C, aquellos pares [q, r], donde q es estado final de A, y r es estado final de B.

Cerradura bajo la Diferencia

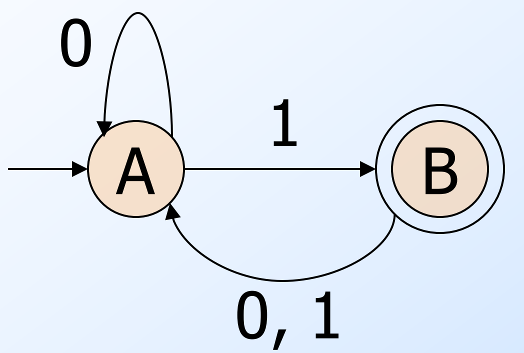
- Si L y M son lenguajes regulares, entonces también lo es $L - M$.

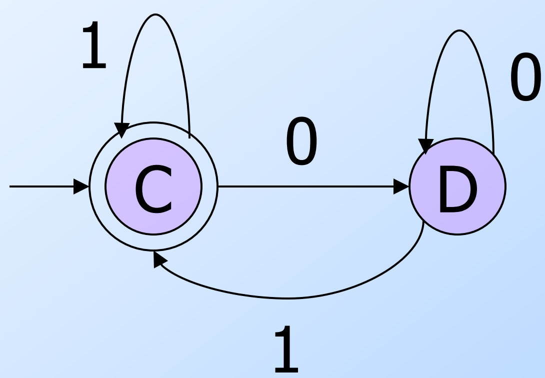
  (las cadenas en L, pero no en M).
- Prueba: Sean A y B autómatas AFD cuyos lenguajes son L y M, resp.
- Construímos C=A×B, el autómata producto.
- Hacemos los estados finales de C, aquellos pares [q, r], donde q es estado final de A, pero r no es estado final de B.

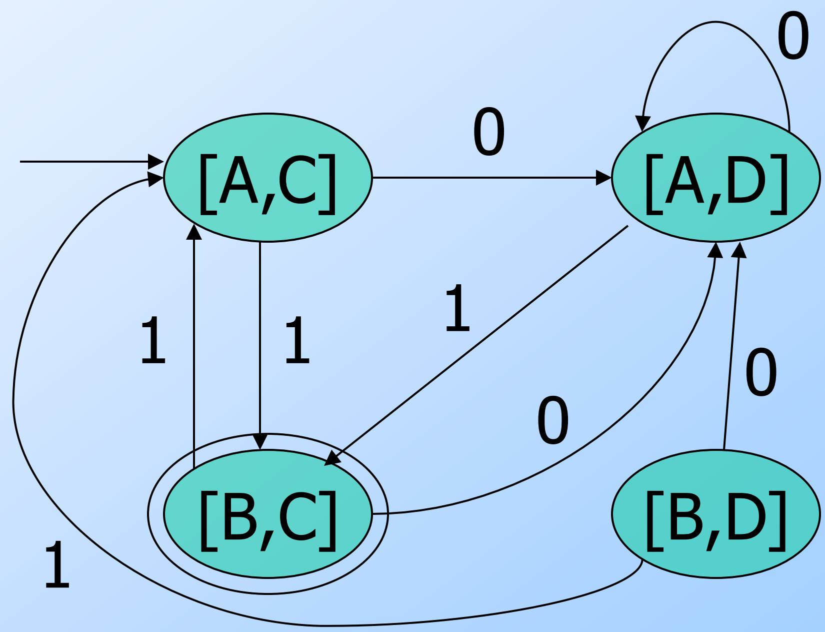
Ejemplo: DFA para L ∪ M



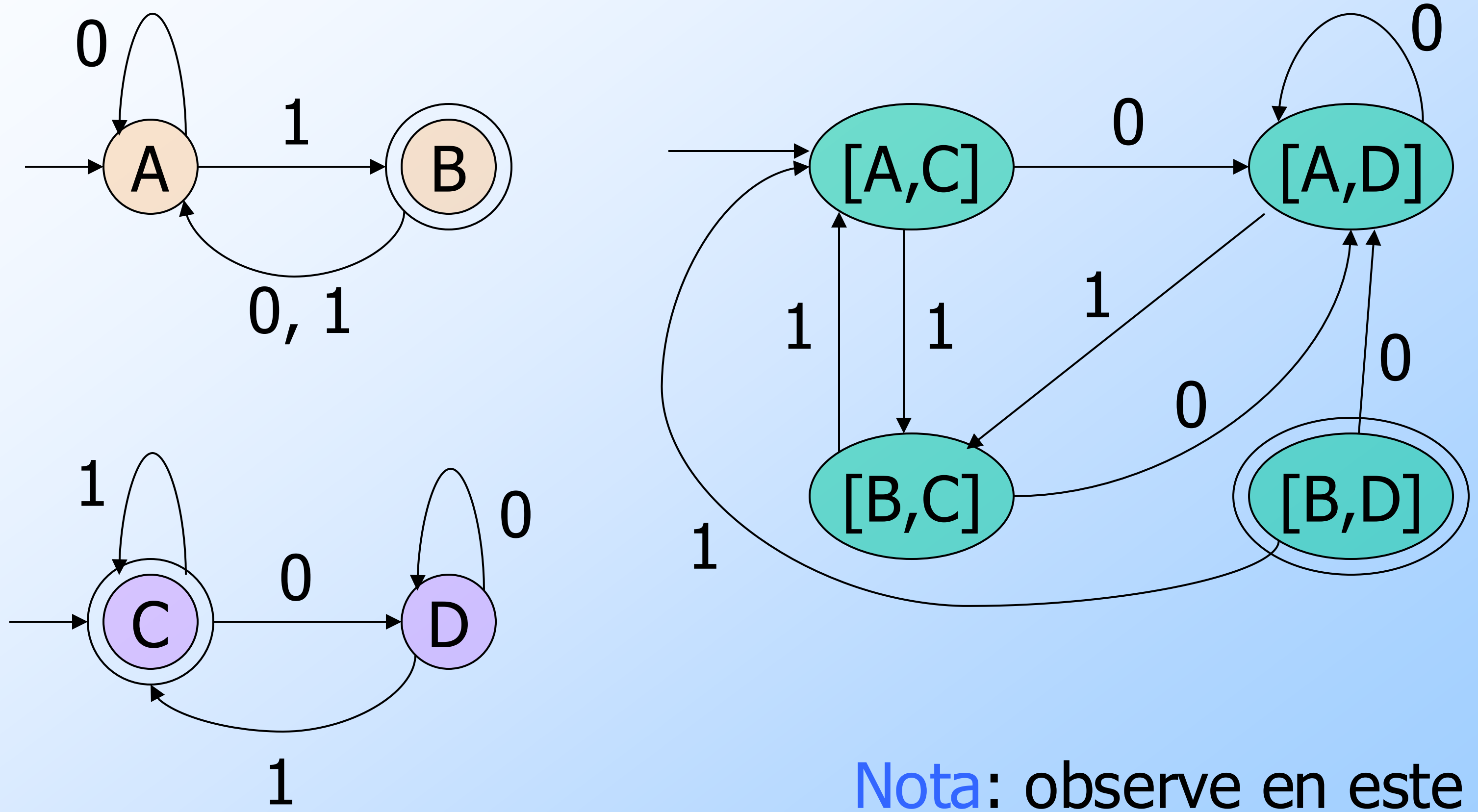
Ejemplo: DFA para L ∩ M





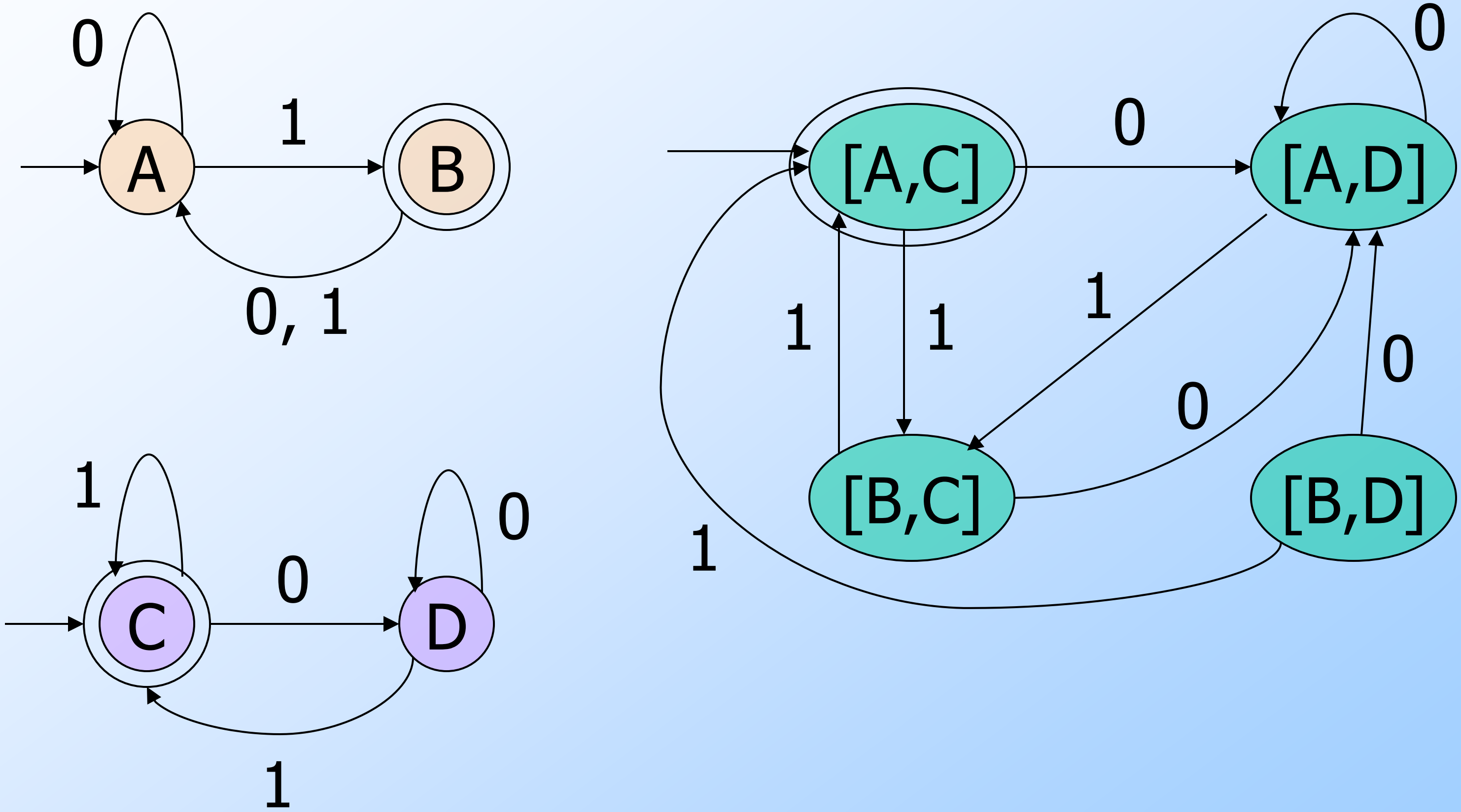


Ejemplo: DFA para L – M



Nota: observe en este ejemplo que la diff. es el lenguaje vacío.

Ejemplo: DFA para M – L



Cerradura bajo Complemento

- El *complemento* de un lenguaje L (con respecto al alfabeto Σ, con Σ* conteniendo L) es Σ* – L.
- Como Σ* es ciertamente un lenguaje regular, el complemento de un lenguaje regular L es también un lenguaje regular.

Ejercicio: Hallar un AFD para el complemento de los siguientes autómatas

Cerradura bajo Reversa

- Dado un lenguaje L, L^R es el conjunto de todas las cadenas cuya cadena reversa está en L.
- Ejemplo: L = {0, 01, 100};
  L^R = {0, 10, 001}.
- Sea E una expresión regular para L.
- Mostraremos cómo revertir E, y producir una espresión regular E^R para L^R.

Reversa de una Regexp

- Base: Si E es un símbolo a, ϵ, ó $\emptyset$, entonces $E^R = E$.
- Inducción: Si E es
  - $F + G$, entonces $E^R = F^R + G^R$.
  - FG, entonces $E^R = G^R F^R$
  - F^*, entonces $E^R = (F^R)^*$.

Ejemplo: Reversa de *regexp*

- Sea E = $\mathbf{01}^* + \mathbf{10}^*$.
- $E^R = (\mathbf{01}^* + \mathbf{10}^*)^R = (\mathbf{01}^*)^R + (\mathbf{10}^*)^R$
- $= (\mathbf{1}^*)^R\mathbf{0}^R + (\mathbf{0}^*)^R\mathbf{1}^R$
- $= (\mathbf{1}^R)^*\mathbf{0} + (\mathbf{0}^R)^*\mathbf{1}$
- $= \mathbf{1}^*\mathbf{0} + \mathbf{0}^*\mathbf{1}$.