Autómatas y Lenguajes Formales 2019-II
Facultad de Ciencias UNAM*
Nota de Clase 5: El Teorema de Kleene

Favio E. Miranda Perea A. Liliana Reyes Cabello Lourdes González Huesca

16 de marzo de 2019

Como hemos visto, la relación entre autómatas finitos y lenguajes regulares parece de equivalencia, las nociones detrás de ambos conceptos la suguieren. En esta nota demostraremos que ese es el caso utilizando el método propuesto por Kleene [1]

1. Teorema de Kleene

Teorema 1. *Un lenguaje es regular si y sólo si es aceptado por un autómata finito.*
Demostración. La prueba es en dos partes:

I Síntesis: Dado un lenguaje regular L, existe un autómata finito M tal que $L = L(M)$.

II Análisis: Dado un autómata finito M, existe una expresión regular α tal que $L(M) = L(\alpha)$. Es decir, $L(M)$ es regular.

□

A continuación se abordarán dos teoremas que demuestran al anterior. Para ello recordemos que las expresiones regulares están en correspondencia con los lenguajes regulares y de esta forma podremos empatar también a las expresiones regulares con los autómatas finitos.

1.1. Teorema de Síntesis de Kleene

En esta sección se demostrará una parte de la doble implicación del teorema de Kleene: se pasará de expresiones regulares a autómatas finitos.

Este teorema se denomina de síntesis ya que se proporcionará una máquina para reconocer un lenguaje regular dado. Es decir, se sintetizará un autómata finito analizando la forma de una expresión regular que genera al lenguaje dado.

*Material elaborado en el marco del proyecto PAPIME PE102117

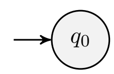
[1] Stephen Cole Kleene fue un matemático estadounidense, alumno de Alonzo Church. También es conocido por iniciar la teoría de la recursión que fue usada para los fundamentos de la Teoría de la Computación, como la noción de computabilidad.

Teorema 2. *Dada una expresión regular α existe un autómata finito M tal que $L(\alpha) = L(M)$.*

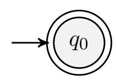
Demostración. La demostración es *constructiva* y se hará mediante inducción sobre las expresiones regulares, es decir proporcionando un autómata que *reconzca* cada caso de una expresión regular.

Base de la Inducción

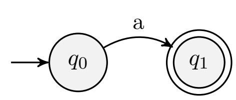
El caso en que $\alpha = \varnothing$, el siguiente autómata reconoce a $L(\alpha)$:



Caso en que $\alpha = \boldsymbol{\varepsilon}$. Entonces el siguiente autómata reconoce a $L(\alpha)$:



Para $\alpha = a$, con $a \in \Sigma$ se tiene el siguiente autómata que reconoce a $L(\alpha)$:

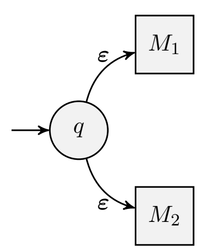


Hipótesis de inducción

Sean M_1, M_2 dos autómatas que reconocen a los lenguajes $L(\alpha_1)$ y $L(\alpha_2)$ respectivamente.

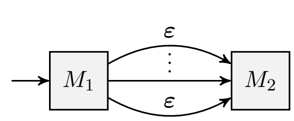
Paso Inductivo

Caso en que $\alpha = \alpha_1 + \alpha_2$. El siguiente autómata reconoce a $L(\alpha)$:



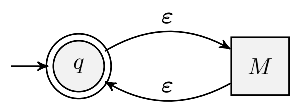
donde M_1, M_2 son autómatas dados por la hipótesis de inducción y las transiciones $\boldsymbol{\varepsilon}$ van hacia los estados iniciales de cada uno de ellos.

Para el caso en que $\alpha = \alpha_1\alpha_2$ entonces el siguiente autómata reconoce a $L(\alpha)$:



donde M_1, M_2 son autómatas que reconocen a $L(\alpha_1), L(\alpha_2)$ dados por la hipótesis de inducción, el estado inicial es el de M_1 y las transiciones $\boldsymbol{\varepsilon}$ van de los estados finales de M_1 hacia el inicial en M_2.

Finalmente el caso $\alpha = \alpha_1^\star$ tiene al siguiente autómata que reconoce a $L(\alpha)$:



donde M_1 es M más q_0 (el estado inicial de M_1) y M_1 es un autómata que reconoce a $L(\alpha_1)$ dado por la hipótesis de inducción y las transiciones ε conectan los estados finales de M_1 con el estado inicial q_0, el cúal se vuelve final, si no lo era en M_1.

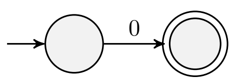
Se deja como ejercicio en todos los casos verificar que el autómata recien construido efectivamente acepta al lenguaje deseado. $\square$

[START] Ejemplo

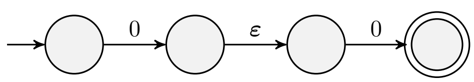
Ejemplo Crear un AFN con ε-transiciones para la siguiente expresión regular:

$$(00+1)^\star(10)^\star$$

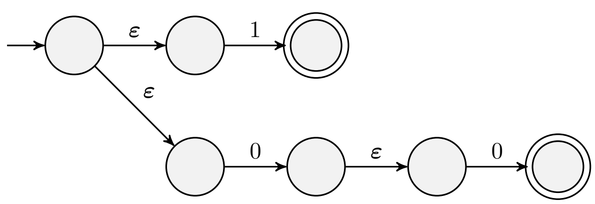
1. $(00+1)^\star(10)^\star$



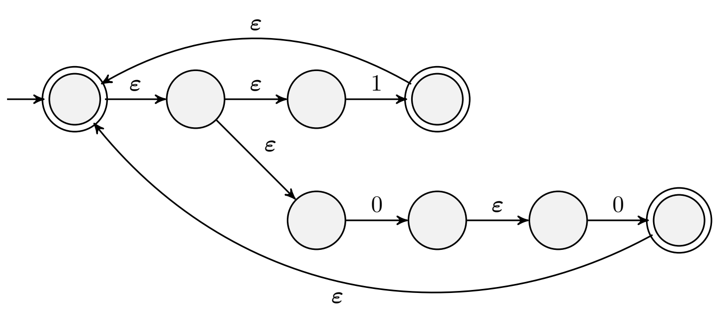
2. $(00+1)^\star(10)^\star$



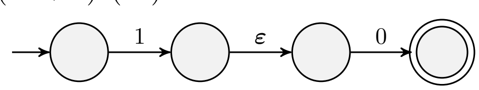
3. $(00+1)^\star(10)^\star$



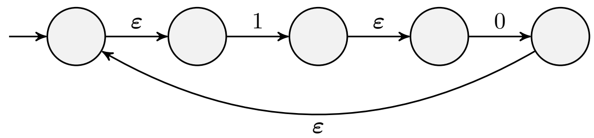
4. $(00+1^\star)(10)^\star$



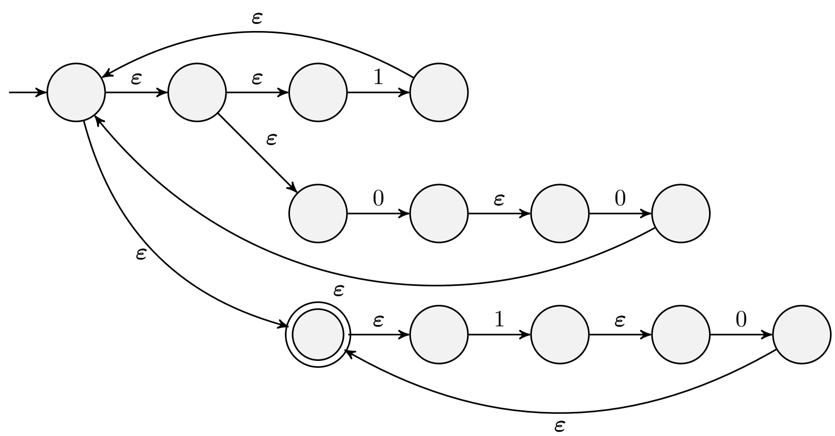
5. $(00+1)^\star(10)^\star$



6. $(00+1)^{\star}(10)^{\star}$



7. $(00+1)^{\star}(10)^{\star}$



[END] Ejemplo

1.2. Teorema de Análisis de Kleene

Ahora demostraremos la segunda parte en donde un autómata finito implica una expresión regular, la noción detrás de esta parte es que analizaremos los lenguajes acumulados de cada estado para generar una expresión regular.

Teorema 3. *Dado un autómata finito M existe una expresión regular α tal que $L(M) = L(\alpha)$. Es decir, $L(M)$ es regular.*

Demostración. Existen diversas demostraciones, nosotros usaremos el método de ecuaciones características usando el Lema de Arden. □

1.2.1. Lema de Arden

Este lema extrae un conjunto de ecuaciones para determinar el lenguaje de aceptación de una máquina. Primero veamos la definicón de dichas ecuaciones y después el método para obtener las ecuaciones dado un autómata.

Definición 1. *Sean $A, B \subseteq \Sigma^{\star}$ y X una variable:*

- *Una ecuación lineal derecha para X es una expresión de la forma:*

$$X = AX + B$$

- *Análogamente, una ecuación lineal izquierda es una expresión de la forma:*

$$X = XA + B$$

Donde el símbolo + denota a la unión de lenguajes.

Lema 1.*: Lema de Arden* *Sean $A, B \subseteq \Sigma^\star$ dos lenguajes y $X = AX + B$ una ecuación lineal derecha. Entonces*

1. *$A^\star B$ es una solución de la ecuación, es decir, $A^\star B = A(A^\star B) + B$.*
2. *Si L es una solución entonces $A^\star B \subseteq L$, es decir, $A^\star B$ es la solución mínima.*
3. *Si $\varepsilon \notin A$ entonces $A^\star B$ es la única solución.*
4. *Si $\varepsilon \in A$ entonces existe un infinidad de soluciones.*

Demostración.

1. Por propiedades de lenguajes regulares se tiene que:

$$A(A^\star B) + B = (AA^\star)B + \boldsymbol{\varepsilon} B = (AA^\star + \boldsymbol{\varepsilon})B = A^\star B$$

2. Sea L una solución, es decir, $L = AL + B$. Sea $w \in A^\star B$ y demostremos que $w \in L$. Como $w \in A^\star B$, existen $u \in A^\star$ y $v \in B$ tales que $w = uv$. Veamos que $uv \in L$ mediante inducción sobre $u \in A^\star$
   - Base, $u = \boldsymbol{\varepsilon}$. En este caso $w = v \in B \subseteq AL + B = L \quad \therefore w \in L$.
   - H. I. Si $u_2 \in A^\star$ entonces $u_2 v \in L$.
   - Paso Inductivo: sea $u = u_1 u_2$ donde $u_1 \in A$ y $u_2 \in A^\star$. Queremos demostrar que $w = uv = u_1 u_2 v \in L$. Por la H.I. tenemos que $u_2 v \in L$, así $u_1(u_2 v) \in AL \subseteq AL + B = L \quad \therefore w = uv = (u_1 u_2)v \in L$.

3. Supongamos que $\boldsymbol{\varepsilon} \notin A$. Sea $L = AL + B$ una solución, veamos que $L = A^\star B$. Por la parte 2 se tiene que $A^\star B \subseteq L$. Si la contención no es propia hemos terminado. Por lo tanto supongamos que $A^\star B \subset L$. En tal caso existe un C tal que $L = A^\star B + C$ y $A^\star B \cap C = \varnothing$. Afirmamos que $C = \varnothing$. Como $L = AL + B$ entonces $A^\star B + C = A^\star B + AC$ de donde al intersectar con C en ambos lados y simplificando se sigue que $(A^\star B \cap C) + C = (A^\star B \cap C) + (AC \cap C)$. Luego entonces $C = AC \cap C$, puesto que $A^\star B \cap C = \varnothing$. De aquí se sigue que $C \subseteq AC$, pero como $\boldsymbol{\varepsilon} \notin A$, esto sólo puede pasar[2] si $C = \varnothing$. Así que $L = A^\star B + C = A^\star B + \varnothing = A^\star B$. Luego entonces $A^\star B$ es la única solución.

4. Se deja como ejercicio.

□

[2] Tómese una cadena $w \in C$ de longitud mínima, entonces $w \in AC$ de donde $w = uv$ con $u \in A$ y $v \in C$, pero como $u \neq \boldsymbol{\varepsilon}$ entonces $v \in C$ tendría menor longitud que w contradiciendo la minimalidad de w.

Esta parte mostrará que un autómata finito implica una expresión regular, esto se hará por medio de un sistema de ecuaciones a partir de un AFN, para ello se abstraerá la noción del lenguaje aceptado desde un estado particular del autómata y no necesariamente del inicial.

Definición 2. *Dado un AFN $M = \langle Q, \Sigma, \delta, q_0, F \rangle$ tal que $Q = \{q_0, \ldots, q_n\}$. Definimos los siguientes conjuntos:*

- *El conjunto de cadenas que se aceptan desde el estado q_i, para cualquier $1 \leq i \leq n$:*

$$L_i = \{w \in \Sigma^\star \mid \delta^\star(q_i, w) \cap F \neq \varnothing\}$$

- *L_0 es el lenguaje aceptado por M, es decir, $L_0 = L(M)$.*

En general no es sencillo calcular directamente los conjuntos L_i. Para obtener una expresión regular completa respecto al autómata, se obtendrá un sistema de ecuaciones a partir de un AFN. Al resolverlo, L_0 será el lenguaje que reconoce el autómata como se mencionó arriba.
El sistema de ecuaciones se define usando:

1. El conjunto de símbolos de Σ tal que existe una transición del estado q_i al estado q_j, para cualesquiera $1 \leq i, j \leq n$, con n el total de estados:

$$X_{i,j} = \{a \in \Sigma \mid q_j \in \delta(q_i, a)\}$$

2. El conjunto auxiliar Y_i que indica si ε es aceptada desde q_i

$$Y_i = \begin{cases} \{\varepsilon\} & \text{si } q_i \in F \\ \varnothing & \text{en otro caso} \end{cases}$$

A partir de estos conjuntos podemos hallar una caracterización de cada lenguaje L_i mediante ecuaciones

$$L_i = \sum_{j=0}^{n} X_{i,j} L_j + Y_i$$

Ésta es la llamada ecuación característica para L_i, dichas ecuaciones general el llamado sistema de ecuaciones características para un AFN.

Finalmente, el Lema de Arden nos indica cómo calcular los conjuntos L_i. Esta será la idea a seguir para demostrar el Teorema de Análisis de Kleene:

1. Dado el autómata M construir los conjuntos $X_{i,j}$, Y_i y plantear las ecuaciones características.

2. Resolver el sistema de ecuaciones características, empleando el Lema de Arden donde se requiera.

3. La solución para L_0 genera una expresión regular para $L(M)$.

Es importante remarcar que la solución del sistema de ecuaciones raramente es única, puesto que éste puede resolverse de distintas maneras. Por supuesto que todas las soluciones resultan equivalentes.