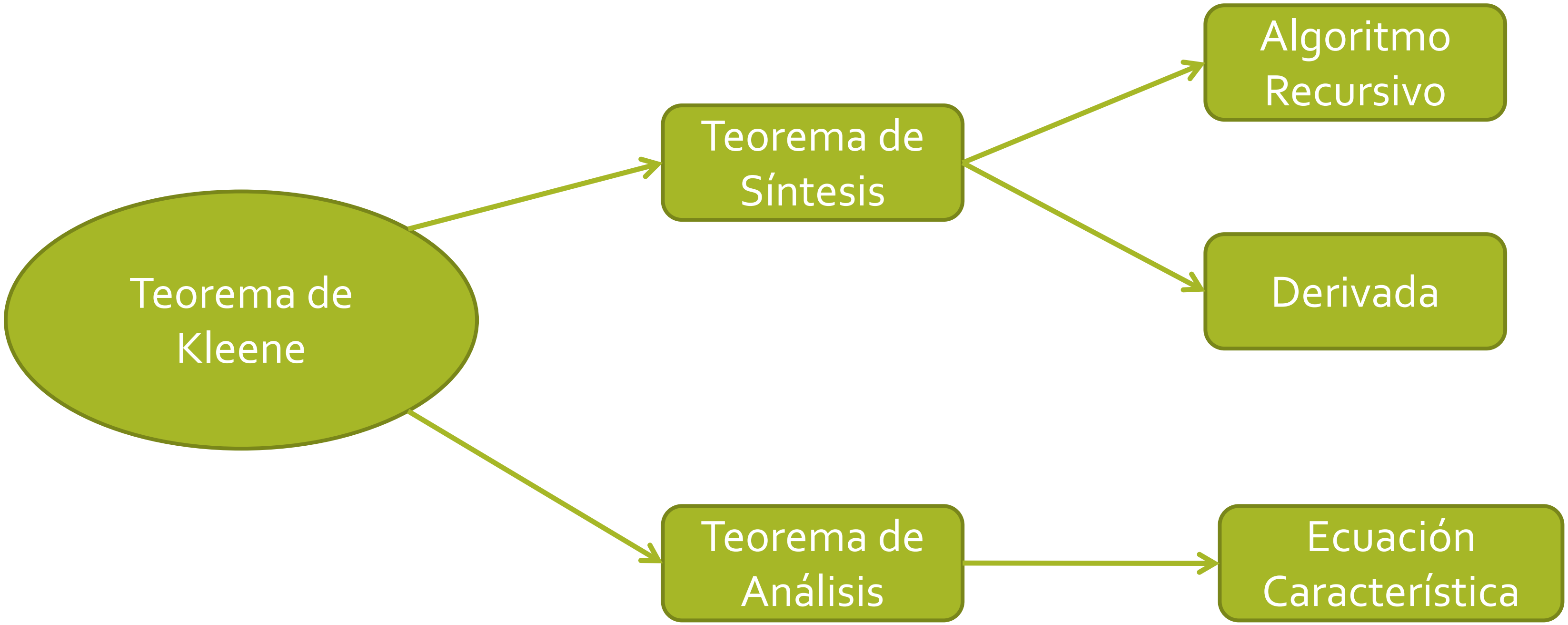
Lenguajes Formales

TEOREMA DE KLEENE

DERIVADAS

Ing. Fabiana Aragón

Teorema de Kleene



Teorema de Sintesis: Derivada

Definicion de Derivada: permite resolver el problema de sintesis, generando el automata minimo que acepta el lenguaje descrito por cualquier expresion regular.

El calculo se realiza en 3 fases:

- Se calculan las derivada de la E.R
- Se obtiene la gramatica que genera el lenguaje representado por la E.R.
- Se transforma la Gramatica de Tipo 3 al A.F

Derivada: Definición

Derivada de una ER α respecto a un símbolo a $\in \Sigma$, se define como $D_a(\alpha)=\{x|a.x \in \alpha\}$. (conjunto de palabras x que aparecen en α)

Calculo: para cada a $\in \Sigma$, cada ER tiene asociada una forma de calcular su derivada

1. $D_a(\emptyset)= \emptyset$
2. $D_a(\lambda)= \emptyset$
3. $D_a(a)= \lambda$
4. $D_a(b)= \emptyset, \forall b \in \Sigma, b \neq a$
5. $D_a(\alpha+\beta)=D_a(\alpha)+D_a(\beta)$
6. $D_a(\alpha.\beta)=D_a(\alpha) . \beta + \delta(\alpha) . D_a(\beta)$
7. $D_a(\alpha^*)=D_a(\alpha) . \alpha^*$

$$\delta(\alpha)= \begin{cases} \lambda & \lambda \in \alpha \\ \emptyset & \lambda \notin \alpha \end{cases}$$

Derivada. Ejemplo

- Ejemplo: se tiene la ER 1*(0+1)*, se calculan sus derivadas respecto a 0 y a 1.

$$
\begin{aligned}
D_0(1^*(0+1)^*) = & \\
& = D_0(1^*)(0+1)^* + \delta(1^*)D_0((0+1)^*) \\
& = D_0(1)\ 1^*(0+1)^* + \lambda D_0((0+1)^*) \\
& = \varnothing\ 1^*(0+1)^* + D_0(0+1)(0+1)^* \\
& = \varnothing + [D_0(0)+D_0(1)\ ](0+1)^* \\
& = [\lambda+\varnothing](0+1)^* \\
& = \lambda(0+1)^* \\
& = \mathbf{(0+1)^*}
\end{aligned}
$$

Propiedades:

1. $D_a(\varnothing)=\varnothing$
2. $D_a(\lambda)=\varnothing$
3. $D_a(a)=\lambda$
4. $D_a(b)=\varnothing,\ \forall b\in\Sigma,\ b\neq a$
5. $D_a(\alpha+\beta)=D_a(\alpha)+D_a(\beta)$
6. $D_a(\alpha.\beta)=D_a(\alpha)\ .\ \beta + \delta(\alpha)\ .\ D_a(\beta)$
7. $D_a(\alpha^*)=D_a(\alpha)\ .\ \alpha^*$

$$
\delta(\alpha)=\begin{cases} \lambda & \lambda\in\alpha \\ \varnothing & \lambda\notin\alpha \end{cases}
$$

Derivada. Ejemplo

- Ejemplo: se tiene la ER 1*(0+1)*, se calculan sus derivadas respecto a 0 y a 1.

$$
\begin{aligned}
D_1(1^*(0+1)^*) = & \\
&= D_1(1^*)(0+1)^* \quad + \delta(1^*)D_1((0+1)^*) \\
&= D_1(1)\, 1^*(0+1)^* + \lambda D_1(0+1)\, (0+1)^* \\
&= \lambda\, 1^*(0+1)^* \quad + [D_1(0) \;\; + D_1(1)](0+1)^* \\
&= 1^*(0+1)^* \quad + [\varnothing + \lambda]\, (0+1)^* \\
&= 1^*(0+1)^* \quad + \lambda(0+1)^* \\
&= (1^* + \lambda)\, (0+1)^* \\
&= \mathbf{1^*(0+1)^*}
\end{aligned}
$$

Propiedades:

1. $D_a(\varnothing)= \varnothing$
2. $D_a(\lambda)= \varnothing$
3. $D_a(a)= \lambda$
4. $D_a(b)= \varnothing, \forall b \in \Sigma, b \neq a$
5. $D_a(\alpha+\beta)=D_a(\alpha)+D_a(\beta)$
6. $D_a(\alpha.\beta)=D_a(\alpha) . \beta + \delta(\alpha) . D_a(\beta)$
7. $D_a(\alpha^*)=D_a(\alpha) . \alpha^*$

$$\delta(\alpha)= \begin{cases} \lambda & \lambda \in \alpha \\ \varnothing & \lambda \notin \alpha \end{cases}$$

Composición de Derivadas

- Se puede realizar la composición de derivada de la siguiente forma: $\mathbf{D_{ab}(\alpha) = D_b(D_a(\alpha))}$

- Ejemplo: Para la expresión del ejemplo anterior

$D_{10}(1^*(0+1)^*) =$

$= D_0(D_1(1^*(0+1)^*))$

$= D_0(1^*(0+1)^*)$

$= \mathbf{(0+1)^*}$

Creación de la Gramática

- A partir de una ER que define un LR, se puede construir la gramática que genera dicho lenguaje. Esta gramática tiene la estructura:

$$\mathbf{G=(\Sigma, \{\alpha_o\} \cup D_i, \alpha_o, \boldsymbol{P})}$$

Donde:

$\boldsymbol{\alpha_o}$ es la ER de la que se parte

$\mathbf{D_i}$ son todas las ER distintas que se obtienen por derivación compuesta con respecto a todos los símbolos de entrada.

$\boldsymbol{P}$ **:**

- Si $D_a(\alpha) = \beta$, donde $\beta \neq \lambda$, $\beta \neq \varnothing$ $\rightarrow$ crear una regla $\alpha ::= a\beta$
- Si $\lambda \in D_a(\alpha)$ $\rightarrow$ crear una regla $\alpha ::= a$
- Si $\lambda \in \alpha_o$ $\rightarrow$ crear una regla $\alpha_o ::= \lambda$
- Si $D_a(\alpha) = \varnothing$ $\rightarrow$ No crear ninguna regla de los tipos $\alpha ::= \beta a$ y $\alpha ::= a\beta$, donde $\beta \in \Sigma \cup \{\lambda\}$

Creación de la Gramática: Ejemplo

- Dada la ER anterior

$$\alpha_0 = 1^*(0 + 1)^*$$

- Se obtiene la gramática que genera el lenguaje asociado, calculando todas las derivadas distintas.
- Se tiene ya calculadas
  - $D_0(\alpha_0) = (0+1)^* = \alpha_1$
  - $D_1(\alpha_0) = \alpha_0$

  - $D_0(\alpha_1) = D_{00}(\alpha_0) = D_0((0+1)^*) = D_0(0+1)(0+1)^* = \lambda(0+1)^* = (0+1)^* = \alpha_1$
  - $D_1(\alpha_1) = D_{01}(\alpha_0) = D_1((0+1)^*) = D_1(0+1)(0+1)^* = \lambda(0+1)^* = (0+1)^* = \alpha_1$

Creación de la Gramática: Ejemplo

Esto da lugar a la gramática

$$\mathbf{G = (\{ 0, 1 \}, \{ \alpha_0, \alpha_1 \}, \alpha_0, P)}$$

y P:

$$\alpha_0 ::= \lambda \mid 0\,\alpha_1 \mid 0 \mid 1\,\alpha_0 \mid 1$$

$$\alpha_1 ::= 0\,\alpha_1 \mid 0 \mid 1\,\alpha_1 \mid 1$$

Intuitivamente, se puede observar que existe una gramática equivalente mas sencilla, que seria:

$$\mathbf{G' = (\{ 0, 1 \}, \{ \alpha_0 \}, \alpha_0, P')}$$

Donde P' seria:

$$\alpha_0 ::= \lambda \mid 0\,\alpha_0 \mid 1\,\alpha_0$$

Esto es debido a que la ER de la que se ha partido, $\alpha_0 = 1^*(0+1)^*$, es equivalente a otra ER mas sencilla $\alpha'_0 = (0+1)^*$

Obtencion del AF

$AF = (\Sigma, D_i \cup \{ F \}, f, \alpha_o, \{ F \})$

Donde f se forma:

Si $D_a(\alpha) = \beta, \beta \neq \lambda, \beta \neq \varnothing$: $\beta \in f(\alpha,a)$

Si $\lambda \in D_a(\alpha)$: $F \in f(\alpha,a)$

Si $\lambda \in \alpha_o$: $F \in f(\alpha_o, \lambda)$

Si $D_a(\alpha) = \varnothing$: $f(\alpha,a) = \varnothing$

Obtención del AF: Ejemplo

- **Ejemplo**: a partir de las derivadas calculadas anteriormente, se puede generar el siguiente AF que reconoce el lenguaje que representa α_0

$$\mathbf{A = (\{ 0, 1 \}, \{ \alpha_0 , F \}, f, \alpha_0 , \{ F \})}$$

Donde f sería.

f	0	1	λ
$\rightarrow q_0$	$\{ q_0, F \}$	$\{ q_0, F \}$	$\{ F \}$
*F			