UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
Facultad de Ciencias

Autor: Adrián Aguilera Moreno

Autómatas y Lenguajes Formales

Tarea 2

1. Demuestra que el lenguaje

$$\{\mathbf{a} \in \{a, b, c\}^* \mid \text{la longitud de } \mathbf{a} \text{ es el cuadrado de un natural.}\}$$

no es regular. Usa tanto el teorema del bombeo como el de Myhill-Nerode.

Demostración: Antes de iniciar con las demostraciones por los teoremas requeridos, analicemos la estructura de las cadenas generadas por

$$L(A) = \{\mathbf{a} \in \{a, b, c\}^* \mid \text{la longitud de } \mathbf{a} \text{ es el cuadrado de un natural.}\}$$

Suponiendo que existe el autómata $A \in DFA$ tal que reconoce el lenguaje anterior.

Nótese que $\Sigma = \{a, b, c\}$ y $L(A) = \Sigma^*$, entonces $\mathbf{a}$ debe ser una cadena finita y supondremos que $L(A)$ es un lenguaje "muy grande" pero finito.

Además, para nuestras demostraciones nos fijaremos en la longitud de las cadenas generadas por nuestro lenguaje y no necesariamente en la estructura de estas cadenas, entonces

$$|abababcbacbabc| = |aaaaabbbbbbccc| = |a^5b^6c^3|$$

Lo anterior se puede generalizar usando inducción. Por tanto, en las demostraciones supondré, sin pérdida de generalidad, que

$$\mathbf{a} = a^ib^jc^r$$

para $i, j, r \in \mathbb{N}$.

Obs. El que $\mathbf{a}$ tenga la estructura anterior no indica que b siempre debe estar en esa posición, incluso podriamos precindir de b o de algún término en la cadena, al suponer esa estructura solo simplificamos el considerar casos (que por ser un conjunto infinito o muy grande, tendriamos muchas combinaciones para concatenar elementos del alfabeto).

Para este ejercicio analicemos 2 versiones en la demostración, *i.e.*,

- **Demostración por Teorema del Bombeo.** Procedamos por reducción al absurdo, supongamos que el lenguaje dado es regular y que $A = \langle Q_A, \Sigma, \delta_A, s, \mathbb{F}_A\rangle$ es un autómata determinista finito [en caso de no serlo, siempre se puede reducir a un DFA] con k estados y $L(A)$ como se definió previamente.
  Sea $\mathbf{a} = a^ib^jc^r$ una cadena aceptada por nuestro autómata con

  $$i + j + r > k \text{ y al menos } j > k.$$

  de lo anterior, podemos deducir que durante la lectura de la cadena $\mathbf{a}$, al menos en 2 ocasiones se pasa por algún estado q en A (es claro que q debe ser estado de A). Así, para la función δ_A de transición en A, tenemos que

  $$\begin{array}{rcl} \delta_A^*(s, a^ib^m) & = & q \\ \delta_A^*(q, b^n) & = & q \\ \delta_A^*(q, b^lc^r) & = & f \in \mathbb{F}_A. \end{array}$$

  $\forall_{m,n,l} \in \mathbb{N}$. Donde $m + n + l = j$ y en particular $n \neq 0$ [b^n no se colapsa a λ].
  Como A es determinista, tenemos que para todo $p \in \mathbb{N}$ se cumple que

  $$\delta_A^*(q, b^{np}) = q$$

como consecuencia, directa se tiene que

$$\delta_A^*(s, a^i b^m b^{np} b^l c^r) = f \in \mathbb{F}_A.$$

pero $m + np + l \neq j$ cuando $p \neq 1$. Por tanto $a^i b^m b^{np} b^l c^r \in L(A)$, lo que implica

$$i + m + np + l + r = t^2$$

con $t \in \mathbb{N}$. Sin embargo,

$$i + m + np + l + r = t^2 \;\not\Rightarrow\; i + m + n(p+1) + l + r = t^2 + n = x^2 \text{ !!!}$$

Contraejemplo: Con $i = 6, m = 3, n = 4, l = 2, r = 1$.

$$\begin{array}{rcll} 16 & = & 6+3+4+2+1 & \text{Con } p = 1. \\ 20 & = & 6+3+4(2)+2+1 = 6+3+8+2+1 & \text{Con } p = 2. \end{array}$$

Claramente 20 no es el cuadrado de un natural.

lo cual es una contradicción, pues esto debe funcionar para cualquier p natural.

$\therefore$ $\{\mathbf{a} \in \{a, b, c\}^* \mid$ la longitud de $\mathbf{a}$ es el cuadrado de un natural.$\}$ **NO** es regular.

- **Demostración por Teorema de Myhill-Nerode.** Por refutación, supongamos que L es regular, entonces tenemos que si

$$\begin{array}{rcl} \alpha & = & (a^i b^j c^r)^{w^2} \text{ tal que } (i+j+r)^{w^2} = x^{w^2} \Rightarrow \alpha \in L. \\ \beta & = & (a^m b^n c^k)^{v^2} \text{ tal que } (m+n+k)^{v^2} = (y)^{v^2} \Rightarrow \beta \in L. \\ \varphi & = & \alpha. \end{array}$$

Donde $v \neq w$. Así,

$$\begin{array}{rcll} \alpha\varphi & = & (a^i b^j c^r)^{w^2} \cdot (a^i b^j c^r)^{w^2} & \\ & = & (a^i b^j c^r)^{w^2 + w^2} & \text{Aplicando el supuesto del inicio.} \\ & \Rightarrow & |\alpha\varphi| = (i+j+r)^{w^2+w^2} & \\ & & \quad = (i+j+r)^{2w^2} & \\ & & \quad = \left((i+j+r)^{w^2}\right)^2 & \end{array}$$

Luego, por el Teorema de Myhill-Nerode tenemos que

$$\alpha \equiv_L \beta \quad \Leftrightarrow \quad \forall_{n \in \mathbb{N}}\, \alpha\varphi^{n^2} \in L \Leftrightarrow \beta\varphi^{n^2} \in L$$

Ahora, verifiquemos que $\beta\varphi \in L$, *i.e.*,

$$\begin{array}{rcl} \beta\varphi & = & (a^m b^n c^k)^{v^2} \cdot (a^i b^j c^r)^{w^2} \\ & \Rightarrow & (y)^{v^2} \cdot (x)^{w^2} \end{array}$$

  - Si $v > w$, entonces

$$\begin{array}{rcll} y^{v^2 - w^2 + w^2} \cdot x^{w^2} & = & y^{v^2 - w^2} \cdot (xy)^{w^2 + w^2} & \\ & = & y^{v^2 - w^2} \cdot \left((xy)^{w^2}\right)^2 & \text{Se usa el supuesto del principio.} \end{array}$$

  sabemos que $v^2 - w^2 = z^2$ solo para $\{v, w, z\}$ una terna pitagórica, y no es cierto para cualesquiera naturales, esto contradice que $\beta\varphi \in L$, sin embargo nos falta analizar otro caso.

– Si $w > v$, entonces

$$y^{w^2-v^2+v^2} \cdot x^{w^2} \quad = \quad y^{w^2-v^2} \cdot (xy)^{v^2+w^2}$$

como podemos observar, sucede lo mismo que en el caso anterior, pues $w^2-v^2 = z^2$ y $v^2+w^2 = t^2$ solo se cumplen para ternas pitagóricas específicas y no para cualesquiera naturales.

De lo anterior, tenemos que $\alpha \not\equiv_L \beta$ (pues no hay afinación en L), lo que implica que existe una cantidad infinita de clases de equivalencias, esto contradice el punto 3 del Teorema de Myhill-Nerode (el tener índice finito), luego

$$\therefore\ \{\mathbf{a} \in \{a,b,c\}^* \mid \text{la longitud de } \mathbf{a} \text{ es el cuadrado de un natural.}\} \ \textbf{NO} \text{ es regular.}$$

En todos los casos se obvia que la demostración para $\mathbf{a} = \lambda$ [cadena vacía] es por vacuidad.

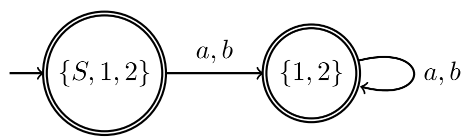
De los analisis anteriores (tanto por Teo. del Bombeo como del Teo. de Myhill-Nerode) concluimos que

$$\therefore\ \{\mathbf{a} \in \{a,b,c\}^* \mid \text{la longitud de } \mathbf{a} \text{ es el cuadrado de un natural.}\} \ \textbf{NO} \text{ es regular.}$$

□

2. Minimiza el autómata de tu respuesta al ejercicio 2 de la tarea 1.

$\triangledown$ **Solución:** El autómata del ejercicio 2 en la tarea 1 se presenta a continuación:



Al que llameremos A. Para la simplicidad de la solución consideremos

$$\begin{aligned} \mathbf{S} &= \{S,1,2\} \\ \mathbf{q} &= \{1,2\} \end{aligned}$$

así, como $\mathbf{S}$ y $\mathbf{q}$ son estados finales, tenemos que la matriz que les representa se ve como la que se muestra a continuación:

$$\begin{array}{cc} & \begin{matrix} \mathbf{S} & \mathbf{q} \end{matrix} \\ \begin{matrix} \mathbf{S} \\ \mathbf{q} \end{matrix} & \begin{pmatrix} n & n \\ n & n \end{pmatrix} \end{array}$$

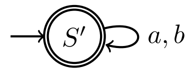
Obs. Podemos omitir analizar la diagonal y la parte (triangular) inferior de la matriz, pues nuestra matriz es simétrica, y por el algoritmo de Minimalización, esto se cumplirá al inicio de las iteraciones y en cuanto termine.

Ahora analicemos la $(2,2)$-pocisión en la matriz con respecto al algoritmo de Minimización, esto es

$$\begin{aligned} (s,q) &= n. \\ (\delta_A(\mathbf{S},a), \delta_A(\mathbf{q},a)) &= (\mathbf{q},\mathbf{q}) \\ &= n. \\ (\delta_A(\mathbf{S},b), \delta_A(\mathbf{q},b)) &= (\mathbf{q},\mathbf{q}) \\ &= n. \end{aligned}$$

por el algoritmo de Minimalización tenemos que si $(q_i,q_j) = n$, entonces $q_i \approx q_j$. Por tanto $\mathbf{S} \approx \mathbf{q}$ [esto es, que formen parte de la misma clase de equivalencia].

Así, el autómata cociente de A es el que a continuación se presenta:



◁

3. Da una CFG para el lenguaje
$$\{a^n b^{2n} c^k | 1 \leq k, n\}.$$

▽ **Solución:**

Sea $G = \langle \{a,b,c\}, \{S,X,Y,Z\}, S, \rightarrow_G \rangle$, donde las reglas de producción estan dadas por

$$\begin{array}{lll} S & \rightarrow_G & XYYZ \\ X & \rightarrow_G & a \mid aXYY \\ Y & \rightarrow_G & b \\ Z & \rightarrow_G & c \mid ZZ \end{array}$$

Para mayor practicidad en los ejercicios siguientes, definimos

$$\begin{array}{lll} \Sigma_G & = & \{a,b,c\} \\ \Gamma_G & = & \{S,X,Y,Z\} \end{array}$$

de lo anterior, se muestra un ejemplo de derivación con base a la gramática propuesta

$$\begin{array}{lll} S & \Rightarrow_G & XYYZ \\ & \Rightarrow_G & (aXYY)YYZ \\ & \Rightarrow_G & a(a)YYYYZ \\ & \Rightarrow_G & aa(b)YYYZ \\ & \Rightarrow_G & aab(b)YYZ \\ & \Rightarrow_G & aabb(b)YZ \\ & \Rightarrow_G & aabbb(b)Z \\ & \Rightarrow_G & aabbbb(ZZ) \\ & \Rightarrow_G & aabbbb(ZZZ) \\ & \Rightarrow_G & aabbbb(c)ZZ \\ & \Rightarrow_G & aabbbbc(c)Z \\ & \Rightarrow_G & aabbbbcc(c). \end{array}$$

Los paréntesis solo son para ayudar a identificar la sustitución en cada iteración. ◁

4. Transforma la gramática de 3 a forma normal de Chomsky.

▽ **Solución:** Sea G' la gramática a construir. Aplicando la regla 1 del algoritmo de transformación a CNF, tenemos que

$$\begin{array}{lll} A_a & \rightarrow & a \\ A_b & \rightarrow & b \\ A_c & \rightarrow & c \end{array}$$

Así, las reglas de producción se ven modificadas

$$\begin{array}{rcl} S & \rightarrow & XYYZ \\ X & \rightarrow & a \mid A_a XYY \\ Y & \rightarrow & b \\ Z & \rightarrow & c \mid ZZ \\ A_a & \rightarrow & a \\ A_b & \rightarrow & b \\ A_c & \rightarrow & c \end{array}$$

Ahora tenemos símbolos no terminales, pero no cumplen ser CNF, los únicos que de momento cumplen estar en CNF son los símbolos del alfabeto. Luego, por la regla 3 del algoritmo de transformación a CNF, tenemos que

$$\begin{array}{rcl} C_1 & \rightarrow & XY \\ C_2 & \rightarrow & YZ \\ C_3 & \rightarrow & A_a X \\ C_4 & \rightarrow & YY \end{array}$$

Ejecutando 4 y 5 en el algoritmo de transformación a CNF, obtenemos

$$\begin{array}{rcl} S & \rightarrow_{G'} & C_1 C_2 \\ X & \rightarrow_{G'} & a \mid C_3 C_4 \\ Y & \rightarrow_{G'} & b \\ Z & \rightarrow_{G'} & c \mid ZZ \\ A_a & \rightarrow_{G'} & a \\ A_b & \rightarrow_{G'} & b \\ A_c & \rightarrow_{G'} & c \\ C_1 & \rightarrow_{G'} & XY \\ C_2 & \rightarrow_{G'} & YZ \\ C_3 & \rightarrow_{G'} & A_a X \\ C_4 & \rightarrow_{G'} & YY \end{array}$$

Eliminando símbolos inútiles tenemos que

$$\begin{array}{rcl} S & \rightarrow_{G'} & C_1 C_2 \\ X & \rightarrow_{G'} & a \mid C_3 C_4 \\ Y & \rightarrow_{G'} & b \\ Z & \rightarrow_{G'} & c \mid ZZ \\ A_a & \rightarrow_{G'} & a \\ C_1 & \rightarrow_{G'} & XY \\ C_2 & \rightarrow_{G'} & YZ \\ C_3 & \rightarrow_{G'} & A_a X \\ C_4 & \rightarrow_{G'} & YY \end{array}$$

De lo anterior, es claro que G' esta en CNF con:

$$G' = \langle \{a, b, c\}, \{S, C_1, C_2, C_3, X, Y, Z, A_a\}, S, \rightarrow_{G'} \rangle$$

$\triangleleft$

5. Transforma la gramática de 3 a forma normal de Greibach (basta con que presentes algunas reglas de producción representativas).

   ▽ **Solución:** Empecemos encontrando las expresiones regulares (o conjuntos) $R_{A,a}$ que indica la iteración 1 del algoritmo para transformar a GNF, esto es

Para $R_{S,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{S,a} &= YC_2 + XC_4YC_2 \\ R_{S,b} = & \{\} = R_{S,c} \end{aligned}$$

pues

$$\begin{aligned} S \Rightarrow C_1C_2 \Rightarrow XYC_2 &\Rightarrow aYC_2. \\ &\Rightarrow C_3C_4YC_2 \Rightarrow A_aXC_4YC_2 \Rightarrow aXC_4YC_2. \end{aligned}$$

Para $R_{X,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{X,a} &= \lambda + XC_4 \\ R_{X,b} = & \{\} = R_{X,c} \end{aligned}$$

pues

$$X \Rightarrow C_3C_4 \Rightarrow A_aXC_4 \Rightarrow aXC_4.$$

Para $R_{Y,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{Y,b} &= \lambda \\ R_{Y,a} = & \{\} = R_{Y,c} \end{aligned}$$

pues $Y \Rightarrow b$.

Para $R_{Z,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{Z,c} &= \lambda + Z \\ R_{Z,a} = & \{\} = R_{Z,b} \end{aligned}$$

pues

$$\begin{aligned} Z &\Rightarrow C. \\ &\Rightarrow ZZ \Rightarrow cZ. \end{aligned}$$

en este caso en particular podemos producir $Z \Rightarrow Z \cdots Z$ un número a lo más de k veces, sin embargo nuestra definición de Z es suficiente[a] y es por esto que no se incluyen en la expresión regular de $R_{Z,c}$.

[a] pues k es un natural, por tanto la cadena $Z \Rightarrow Z \cdots Z$ es finita.

Para $R_{C_1,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{C_1,a} &= Y + XC_4Y \\ R_{C_1,b} = \ \{\} &= R_{C_1,c} \end{aligned}$$

pues

$$\begin{aligned} C_1 \Rightarrow XY &\Rightarrow aY. \\ &\Rightarrow C_3C_4Y \Rightarrow A_aXC_4Y \Rightarrow aXC_4Y. \end{aligned}$$

Para $R_{C_2,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{C_2,b} &= Z \\ R_{C_2,a} = \ \{\} &= R_{C_2,c} \end{aligned}$$

pues $C_2 \Rightarrow YZ \Rightarrow bZ$.

Para $R_{C_3,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{C_3,a} &= X \\ R_{C_3,b} = \ \{\} &= R_{C_3,c} \end{aligned}$$

pues $C_3 \Rightarrow A_aX \Rightarrow aX$.

Para $R_{C_4,\alpha}$ con $\alpha \in \Sigma_G$ tenemos que

$$\begin{aligned} R_{C_4,a} &= X \\ R_{C_4,b} = \ \{\} &= R_{C_4,c} \end{aligned}$$

pues $C_4 \Rightarrow YY \Rightarrow bY$.

Ahora, encontremos las gramáticas lineales por la izquierda que indica la iteración 2 del algoritmo de transformación a GNF, esto es

$$\begin{aligned} T_{S,a} &\rightarrow YC_2 \mid XC_4YC_2 \\ T_{X,a} &\rightarrow \lambda \mid XC_4 \\ T_{Y,b} &\rightarrow \lambda \\ T_{Z,c} &\rightarrow \lambda \mid Z \\ T_{C_1,a} &\rightarrow Y \mid XC_4Y \\ T_{C_2,b} &\rightarrow Z \\ T_{C_3,a} &\rightarrow X \\ T_{C_4,a} &\rightarrow Y \end{aligned}$$

Bajo la misma iteración, sustituimos las reglas para llevarlas a la forma GNF

$$
\begin{array}{rcll}
Y & \rightarrow & b & \\
A_a & \rightarrow & a & \\
S & \rightarrow & aT_{C_1,a}C_2 & \text{Sustituimos en } S \text{ de } G'. \\
X & \rightarrow & a \mid aT_{C_3,a}C_4 & \text{Sustituimos en } X \text{ de } G'. \\
Z & \rightarrow & c \mid cZ & \text{Sustituimos en } Z \text{ de } G'. \\
C_1 & \rightarrow & aY \mid aXC_4Y & \text{Sustituimos en } C_1 \text{ de } G'. \\
C_2 & \rightarrow & bZ & \text{Sustituimos en } C_2 \text{ de } G'. \\
C_3 & \rightarrow & aX & \text{Sustituimos en } C_3 \text{ de } G'. \\
C_4 & \rightarrow & bY & \text{Sustituimos en } C_4 \text{ de } G'. \\
T_{S,a} & \rightarrow & bC_2 \mid aC_4YC_2 \mid aT_{c_3,a}C_4YC_2 & \\
T_{X,a} & \rightarrow & \lambda \mid aC_4 \mid aT_{C_3,a}C_4^2 & \\
T_{Y,b} & \rightarrow & \lambda & \\
T_{Z,c} & \rightarrow & \lambda \mid bZ & \\
T_{C_1,a} & \rightarrow & b \mid aC_4Y \mid aT_{C_3,a}C_4^2Y & \\
T_{C_2,b} & \rightarrow & c|cZ & \\
T_{C_3,a} & \rightarrow & a \mid aT_{C_3,a}C_4 & \\
T_{C_4,a} & \rightarrow & b &
\end{array}
$$

Para todas las $T_{i,j}$ se hacen modificaciones con respecto a la gramática lineal generada con anterioridad.

Ahora, por la iteración 3 del algoritmo de transformación a GNF, procedamos a eliminar las producciones-ε. Así,

$$
\begin{array}{rcl}
Y & \rightarrow & b \\
A_a & \rightarrow & a \\
S & \rightarrow & aT_{C_1,a}C_2 \\
X & \rightarrow & a \mid aT_{C_3,a}C_4 \\
Z & \rightarrow & c \mid cZ \\
C_1 & \rightarrow & aY \mid aXC_4Y \\
C_2 & \rightarrow & bZ \\
C_3 & \rightarrow & aX \\
C_4 & \rightarrow & bY \\
T_{S,a} & \rightarrow & bC_2 \mid aC_4YC_2 \mid aT_{c_3,a}C_4YC_2 \\
T_{X,a} & \rightarrow & aC_4 \mid aT_{C_3,a}C_4^2 \\
T_{Z,c} & \rightarrow & bZ \\
T_{C_1,a} & \rightarrow & b \mid aC_4Y \mid aT_{C_3,a}C_4^2Y \\
T_{C_2,b} & \rightarrow & c|cZ \\
T_{C_3,a} & \rightarrow & a \mid aT_{C_3,a}C_4 \\
T_{C_4,a} & \rightarrow & b
\end{array}
$$

por último eliminemos los símbolos inaccesibles, *i.e.*,

$$
\begin{array}{rcl}
S & \to & aT_{C_1,a}C_2 \\
Y & \to & b \\
A_a & \to & a \\
X & \to & a \mid aT_{C_3,a}C_4 \\
Z & \to & c \mid cZ \\
C_2 & \to & bZ \\
C_4 & \to & bY \\
T_{C_2,b} & \to & c|cZ \\
T_{C_3,a} & \to & a \mid aT_{C_3,a}C_4
\end{array}
$$

Se puede comprobar por fuerza bruta (y por inducción) que la gramática anterior se encuentra en GNF. $\triangleleft$

6. Demuestra que el conjunto
$$\{a^n b^m c^k \mid n = k \wedge k = 3m\}$$
no es un CFL.

Demostración: Veamos que nuestro lenguaje es equivalente a

$$\{a^{3m} b^m c^{3m}\}$$

Así, analicemos 6 posibles casos:

- Supongamos, sin pérdida de generalidad, que bombearemos a^i [un extremo[1]] en nuestro lenguaje. Esto es,

$$a^{3m-i}\underbrace{a\cdots a}_{\alpha}b^m c^{3m}$$

Obs. Se puede bombear en cualquier parte de la cadena de a's.

con $|\alpha| = ri \in \mathbb{N}$. Esto se cumple solo cuando $r = 1$, en otro caso

  - $r > 1$. Entonces,

$$
\begin{array}{rcl}
(3m-i)+i & < & (3m-i)+ri \\
\Rightarrow 3m & < & (3m-i)+ri.
\end{array}
$$

lo que implica que $|a^{3m-i}\alpha| > |c^{3m}|$, ahora es claro que esa cadena no esta en L.

  - $r = 0$. Entonces,

$$
\begin{array}{rcl}
(3m-i)+i & > & (3m-i)+ri \\
 & > & (3m-i)+0 \\
\Rightarrow 3m & > & 3m-i.
\end{array}
$$

lo que implica que $|a^{3m-i}\alpha| < |c^{3m}|$, ahora es claro que esa cadena no esta en L.

- Supongamos, sin pérdida de generalidad, que bombearemos a^i y c^j [Extremos] en nuestro lenguaje. Esto es,

$$a^{3m-i}\underbrace{a\cdots a}_{\alpha}b^m\underbrace{c\cdots c}_{\varphi}c^{3m-j}$$

[1] Nótese que esto es análogo a bombear c^i.

Obs. Se puede bombear en cualquier parte de las cadenas de a's y c's respectivamente.

con

- $|\alpha| = ri = rj = |\beta| \neq 1$. Entonces,
  Caso 1: $ri = 0$.

$$\begin{aligned} 3m &> (3m-i)+ri = (3m-j)+pj \\ &> 3m-i = 3m-j. \end{aligned}$$

  lo que implica que $3|b^m| > |a^{3m-i}\alpha| = |c^{3m-j}\beta|$, ahora es claro que esa cadena no esta en L.
  Caso 2: $ri > 1$.

$$\begin{aligned} 3m &< (3m-i)+ri = (3m-j)+pj \\ &< 3m+(r-1)i = 3m+(p-1)j. \end{aligned}$$

  el caso mínimo es cuando $r = 2 = p$, entonces nos quedan $3m+i = 3m+j$ como longitudes de α y β respectivamente, luego descartamos la cadena de L por no cumplir que $|a^{3m-i}\alpha| = 3|b^m| = |c^{3m-j}\beta|$.
- $|\alpha| \neq |\beta|$ y $|\alpha| = ri$, $|\beta| = pj$. Entonces,

$$\begin{aligned} 3m &\neq (3m-i)+ri \\ &\neq 3m+(r-1)i. \end{aligned} \quad \text{y} \quad \begin{aligned} 3m &\neq (3m-j)+pj \\ &\neq 3m+(p-1)j. \end{aligned}$$

  lo cual es claro por los ejemplos más desarrollados arriba. Así, esas cadenas no pertenecen a L.

- Supongamos, sin pérdida de generalidad, que bombearemos b^i [Medio] en nuestro lenguaje. Esto es,

$$a^{3m}b^{m-(i+j)}\underbrace{b\cdots b}_{\alpha}b^jc^{3m}$$

Obs. Se puede bombear en cualquier parte de la cadena de b's.

con $|\alpha| = ri$. Entonces la cadena solo pertenece a L cuando $r = 1$, en otro caso

$$m-(i+j)+i+j = m \neq m-(i+j)+ri+j$$

Así, las cadenas de esa forma no son parte de L.

- Supongamos, sin pérdida de generalidad, que bombearemos las subcadenas a^i y b^j [Medio y un extremo [2]] en nuestro lenguaje. Esto es,

$$a^{3m-i}\underbrace{a\cdots a}_{\alpha}b^{m-j}\underbrace{b\cdots b}_{\beta}c^{3m}$$

Obs. Se puede bombear en cualquier parte de las cadenas de a's y b's respectivamente.

con $|\alpha| = ri$ y $|\beta| = pj$. De lo anterior, la única cadena que esta en el lenguaje es cuando $r = 1 = p$. En otro caso, pasa el primer caso y el anterior, con lo cual ya vimos que las cadenas generadas no pertenecen a L.

[2] Esto es análogo a bombear b^i y c^j.

- Otro caso que vale la pena analizar y es más obvio de notar (que no pertenece a L), es cuando se bombea $a^i b^j$, pues el único caso que es aceptado por L es cuando se bombea una solo vez, en otro caso tendriamos algo como lo siguiente

$$a^{3m-i} a^i b^j \cdots a^i b^j b^{m-j} c^{3m}$$

  o como lo siguiente

$$a^{3m-i} b^{m-j} c^{3m}$$

  y como $|a^{3m-i}| < |c^{3m}|$ (o por el caso 1), deducimos que no pertenece a L.

- Un último caso a analizar (y que también es fácil de ver, el porque no pertenece a L) es cuando se bombea $a^i b^m c^j$, pues se tendría que cuando se bombea una cantidad de veces distinta de 1 se tiene que

$$a^{3m-i} a^i b^m c^j \cdots a^i b^m c^j c^{3m-j}.$$

Después del análisis anterior, podemos concluir que $L \notin$ CFL. $\square$