

AUXILIAR 2:

CC3102 TEORÍA DE LA COMPUTACIÓN

PROFESOR: GONZALO NAVARRO

AUXILIAR: PABLO MUÑOZ

13 DE AGOSTO DEL 2014

P1. Responda las siguientes preguntas

- a) Sea M un AFD. ¿Qué condición es necesaria y suficiente para que $\varepsilon \in \mathcal{L}(M)$?
- b) Si $M(K, \Sigma, \delta, s, F)$ es un AFD y $F = K$ entonces $\mathcal{L}(M) = \Sigma^*$.
- c) Si $M(K, \Sigma, \delta, s, F)$ es un AFND y $F = K$ entonces $\mathcal{L}(M) = \Sigma^*$.

P2. Construya AFDs que reconozcan los siguientes lenguajes sobre $\Sigma = \{a, b\}$.

- a) Las palabras en que cada a es precedido por b .
- b) Las palabras que contienen a $abab$ como substring.
- c) Las palabras que no contienen ni a aa ni a bb como substring.
- d) Las palabras con una cantidad impar de a 's y una cantidad par de b 's.

P3. Un estado $q \in K$ de un AFD $M(K, \Sigma, \delta, s, F)$ se dice *alcanzable* si existe una palabra en Σ^* tal que $(q_o, w) \vdash (q, \varepsilon)$. Muestre que un automata M' obtenido de M eliminando un estado no alcanzable reconoce el mismo lenguaje que M . ¿Cómo puede encontrar de manera eficiente todos los estados alcanzables de un AFD dado?

P4. Dé diagramas de AFNDs que reconozcan los siguientes lenguajes

- a) $(ab)^*(ba)^*|aa^*$
- b) $((ab|aab)^*a^*)^*$
- c) $((a^*b^*a^*)^*b)^*$
- d) $(ba|b)^*|(bb|a)^*$

P5. Considera una definición alternativa de AFNDs dada por $M(K, \Sigma, \Delta, S, F)$, donde ahora en vez de un estado inicial, M puede elegir no determinísticamente un estado de $S \subseteq K$ para iniciar la operación.

- a) ¿Cómo se compara esta definición con la vista en clase?
- b) Usando esta definición, construya un AFND que reconozca el lenguaje $L \subseteq \{a_1, \dots, a_n\}^*$ de palabras que carecen de al menos un símbolo del alfabeto.

P6. [Control 1 '98] Es fácil determinar pertenencia de una palabra a un lenguaje utilizando un AFD. Por otro lado, es fácil describir un lenguaje regular construyendo un AFND. Sin embargo, al construir un AFD a partir de un AFND la cantidad de estados puede aumentar exponencialmente, aumentando el espacio necesario. Una solución alternativa es simular un AFND.

Escriba en pseudo-código una función $Acepta(M, w)$ que dado un AFND $M(K, \Sigma, \Delta, s, F)$ y una palabra $w \in \Sigma^*$ retorne V si $w \in \mathcal{L}(M)$ y F sino. Puede usar la notación $M.K$, $M.s$, etc., y contar con las operaciones básicas necesarias (eg. operaciones de conjuntos).