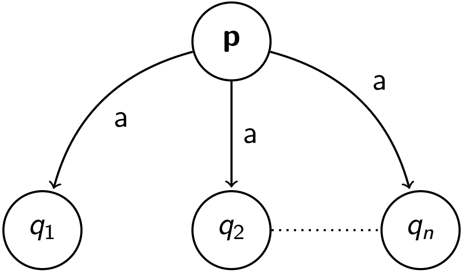
Conteúdo

AFN

AFN: Visão macro

O não-determinismo nos autômatos é expresso por uma função tal que, para $p \in Q$, o estado corrente, e um símbolo lido, esta função determina aleatoriamente um novo estado de um conjunto de estados alternativos.

Um possível recorte de um diagrama para um AFN[1]



[1] Menezes, PB.

Definição formal de um AFN

Definição

Um autômato finito não-determinístico (AFN) é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, em que

1. Q é um conjunto finito de estados;
2. Σ é um conjunto finito chamado alfabeto;
3. $\delta : Q \times \Sigma \rightarrow 2^Q$ é a função programa a qual é uma função total. Assim, para um estado p e um símbolo a $\delta(p, a) = \{q_1, q_2, \ldots, q_n\}$ é a transição do autômato;
4. $q_0 \in Q$ é o estado inicial; e
5. $F \subseteq Q$ é o conjunto de estados de aceitação.

Observem a definição de função programa acima.

Recordando função total

Função total

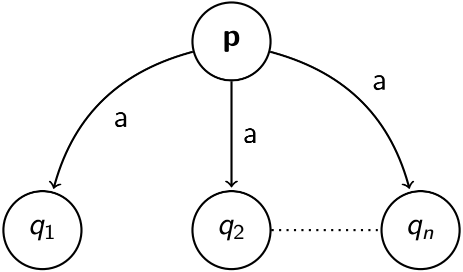
Considerando dois conjuntos, X chamado de domínio e Y de contradomínio e, dado:

$$f : X \to Y,$$

dizemos que:

- f é total: para todos $x \in X$, $\exists\ y \in Y$ tal que $y = f(x)$.

Um possível recorte de um diagrama para um AFN[2]



[2] Menezes, PB.

Função programa estendida ou COMPUTAÇÃO

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN.
A função programa estendida ou computação de M, denotada por

$$\delta^* : 2^Q \times \Sigma^* \to 2^Q$$

é a função programa $\delta : Q \times \Sigma \to 2^Q$ estendida para palavras e conjunto de estados (P) como:

$$\delta^*(P, \epsilon) = P$$
$$\delta^*(P, aw) = \delta^*(\cup_{(q \in P)} \delta(q, a), w)$$

ou seja, é a sucessiva aplicação da função programa para cada símbolo da palavra.

Função programa: mais detalhes

continuação...

Ou ainda, dados os estados $\{q_1, q_2, \ldots, q_n\}$ e um símbolo a, vale que

$$\delta^*(\{q_1, q_2, \ldots, q_n\}, a) = \delta(q_1, a) \cup \delta(q_2, a) \cup \ldots \cup \delta(q_n, a)$$

ou seja, é a união de todas as possibilidades de aplicação da função de transição sobre o símbolo lido.

Transição vazia

$\delta(p, a) = \varnothing$

Se $\delta(p, a) = \varnothing$ afirma-se que a transição é indefinida para (p, a) e, portanto, o autômato para rejeitando a entrada.

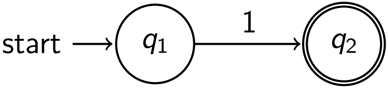
Transição vazia

$\delta(p, a) = \varnothing$

Se $\delta(p, a) = \varnothing$ afirma-se que a transição é indefinida para (p, a) e, portanto, o autômato para rejeitando a entrada.

Exemplo de transição vazia

$\Sigma = \{0, 1\}, w = 001.$



A função programa $\delta(q_1, 0) = \varnothing$, ou seja, não está definida.

Movimento vazio

$\delta(p, \epsilon) = p'$

Se $\delta(p, \epsilon) = p'$ afirma-se que o movimento do estado p para o estado p' é vazio, ou seja, acontece mesmo sem haver um símbolo lido.

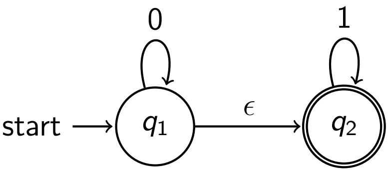
Movimento vazio

$\delta(p, \epsilon) = p'$

Se $\delta(p, \epsilon) = p'$ afirma-se que o movimento do estado p para o estado p' é vazio, ou seja, acontece mesmo sem haver um símbolo lido.

Exemplo de movimento vazio

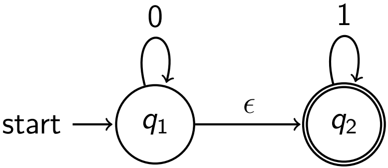
$\Sigma = \{0, 1\}, w = 001.$



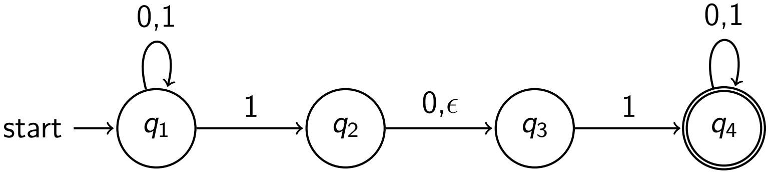
Qualquer símbolo 0 antecede o símbolo 1.

Movimento vazio: como entender

- Toda vez que o autômato estiver numa transição ϵ significa que ele está em *ambos* os estados
- Se a cadeia é ϵ o estado atual do autômato é duplo $\{q_0, q_1\}$
- Se o símbolo lido for '0' o estado atual do autômato é novamente $\{q_0, q_1\}$



Exemplo de um autômato finito não-determinístico (AFN).[3]



[3] Sipser,M.

Formalizando o autômato

Formalizando o autômato $N_1 = (Q, \Sigma, \delta, q_0, F)$ temos:

1. $Q = \{q_1, q_2, q_3, q_4\}$;
2. $\Sigma = \{0, 1\}$;
3. $\delta : Q \times \Sigma \to Q$ é descrita como:

	0	1	ϵ
q_1	q_1	$\{q_1, q_2\}$	$\varnothing$
q_2	q_3	$\varnothing$	q_3
q_3	$\varnothing$	q_4	$\varnothing$
q_4	q_4	q_4	$\varnothing$

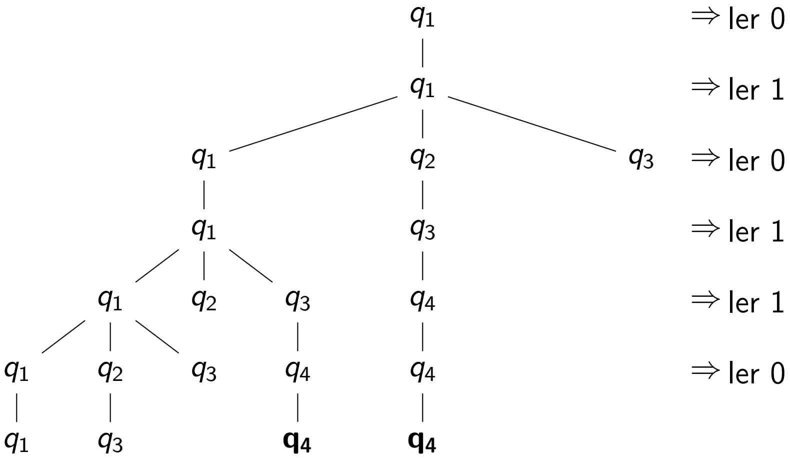
4. q_1 é o estado inicial; e
5. $F = \{q_4\}$.

Como entender o AFN em execução: modo alternativo

- A cada estado alternativo do AFN, ocorre a divisão da máquina em múltiplas cópias
- Cada cópia segue o processamento de acordo com seu estado, o símbolo lido e a função programa
- O símbolo ϵ, como símbolo lido, permite o avanço para o próximo estado do autômato

Vejamos agora uma representação da execução do AFN N_1 para a palavra **010110**

Árvore de execução do AFN N_1 para **010110**



Aceita ou rejeita

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN.
A linguagem aceita ou linguagem reconhecida por N, denotada por

$$L(N) \text{ ou } \mathrm{Aceita}(N)$$

é o conjunto de todas as palavras $\in \Sigma^*$ t.q. existe pelo menos um caminho alternativo que aceita a palavra a partir de $\{q_0\}$, ou seja,

$$L(N) = \{w \mid \delta^*(\{q_0\}, w) \cap F \neq \varnothing\}$$

Explicando...

Ou seja, dada uma palavra w, um AFN para por um dos dois motivos:

1. Aceita w
   Após processar o último símbolo lido existe pelo menos um estado final pertencente ao conjunto dos estados alternativos (nas execuções alternativas) que foi atingido
2. Rejeita w
   1. Após processar o último símbolo lido os estados alternativos não são finais
   2. Em algum momento do processamento de w o conjunto de estados alternativos atingido é vazio. O AFN para por indefinição.

Exemplo 2[4]

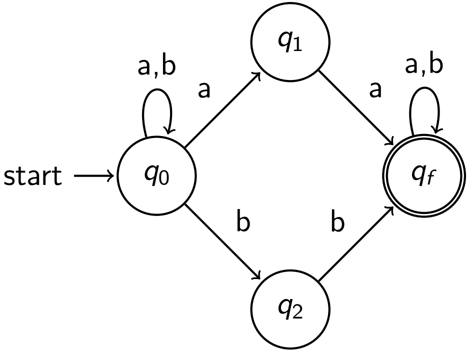
Projetar um AFN N_2 sobre $\Sigma = \{a, b\}$ que considere

$$L_2 = \{w \mid w \text{ possui } aa \text{ ou } bb \text{ como subpalavra}\}$$

tal que $N_2 = (\{q_0, q_1, q_2, q_f\}, \Sigma, \delta_f, q_0, \{q_f\})$.

[4] Menezes, PB

Exemplo 2: uma solução



Detalhes de N_2

Notem que em N_2 temos como função de transição:

δ_2	a	b
q_0	$\{q_0, q_1\}$	$\{q_0, q_2\}$
q_1	q_f	$\varnothing$
q_2	$\varnothing$	q_f
q_f	q_f	q_f

Exemplo 3[5]

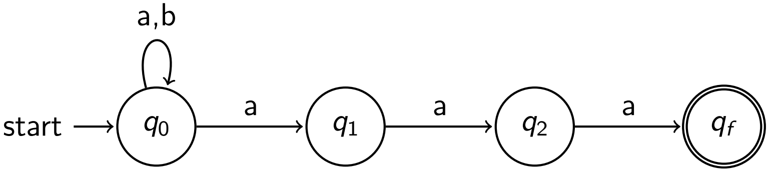
Projetar um AFN N_3 sobre $\Sigma = \{a, b\}$ que considere

$$L_2 = \{w \mid w \text{ possui } aaa \text{ como sufixo}\}$$

tal que $N_2 = (\{q_0, q_1, q_2, q_f\}, \Sigma, \delta_f, q_0, \{q_f\})$.

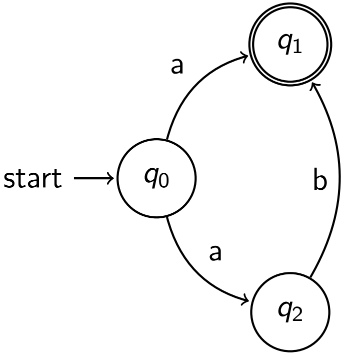
[5] Menezes, PB

Exemplo 3: uma solução



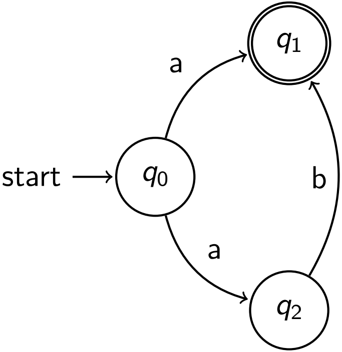
Exercício em sala (1): 5 minutos

Determine a linguagem do AFN N_4 abaixo:



Exercício em sala (1): 5 minutos

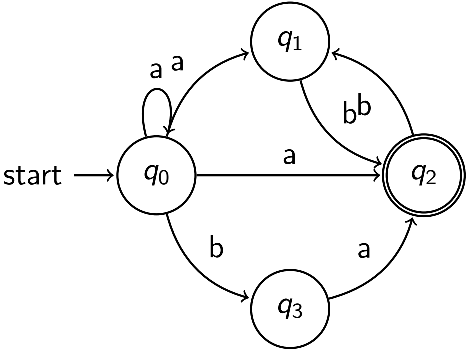
Determine a linguagem do AFN N_4 abaixo:



Resposta: $L(N_4) = \{a \cup ab\}$

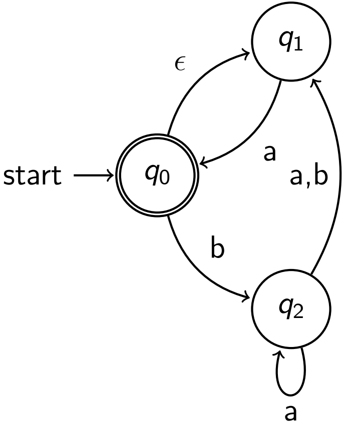
Exercício em sala (2): 5 minutos

Determine a linguagem do AFN abaixo:



Exercício em sala (3): 5 minutos

Determine a linguagem do AFN abaixo:



Observação

Parece mas...

A inclusão do não-determinismo é, aparentemente, um acréscimo à capacidade de resolução aos AFD mas na realidade o não-determinismo não aumenta seu poder computacional.

AFD equivalente

Assim, para cada AFN é possível construir um AFD equivalente que realiza as mesmas computações.
O contrário também é verdadeiro.