Autômato Finito Não-determinístico

Esdras Lins Bispo Jr.
bispojr@ufg.br

Linguagens Formais e Autômatos
Bacharelado em Ciência da Computação

20 de novembro de 2017

Plano de Aula

1. Revisão
    - Fecho em Linguagens Regulares (cont.)
2. Não-determinismo
3. Não-determinismo (cont.)

Sumário

Operações regulares

Teorema 1.25

A classe de linguagens regulares é fechada sob a operação de união.

Prova

Sejam A e B duas linguagens regulares. Se A e B são regulares, então existem dois AFDs $M_A = (Q_A, \Sigma_A, \delta_A, q_A, F_A)$ e $M_B = (Q_B, \Sigma_B, \delta_B, q_B, F_B)$ que as reconhecem, respectivamente. Como passo auxiliar, iremos construir o AFDs estendidos $M'_A = (Q'_A, \Sigma'_A, \delta'_A, q'_A, F'_A)$ e $M'_B = (Q'_B, \Sigma'_B, \delta'_B, q'_B, F'_B)$ dos AFDs M_A e M_B, respectivamente. Um AFD estendido O é um AFD equivalente a um dado AFD P de forma que $\Sigma_P \subset \Sigma O$. Desta forma, temos:

Operações regulares

Prova (cont.)

Elementos de M'_A:

- $Q'_A = Q_A \cup \{q_{fugaA}\}$;
- $\Sigma'_A = \Sigma_A \cup \Sigma_B$;
- $\delta'_A(q, a) = \begin{cases} \delta_A(q, a), & \text{se } a \in \Sigma_A \\ q_{fugaA}, & \text{caso contrário} \end{cases}$

  em que $q \in Q'_B$ e $a \in \Sigma'_B$;
- $q'_A = q_A$;
- $F'_A = F_A$.

Operações regulares

Prova (cont.)

Elementos de M'_B:

- $Q'_B = Q_B \cup \{q_{fugaB}\}$;
- $\Sigma'_B = \Sigma_A \cup \Sigma_B$;
- $\delta'_B(q, a) = \begin{cases} \delta_B(q, a), & \text{se } a \in \Sigma_B \\ q_{fugaB}, & \text{caso contrário} \end{cases}$
  em que $q \in Q'_B$ e $a \in \Sigma'_B$;
- $q'_B = q_B$;
- $F'_B = F_B$.

Operações regulares

Prova (cont.)

De posse de M'_A e M'_B, será construído o AFD $M_{A \cup B} = (Q, \Sigma, \delta, q_0, F)$ que reconhece $A \cup B$:
Elementos de $M_{A \cup B}$:

- $Q = Q'_A \times Q'_B$;
- $\Sigma = \Sigma'_A$;
- $\delta((x, y), a) = (\delta'_A(x, a), \delta'_B(y, a))$ em que $(x, y) \in Q$ e $a \in \Sigma$;
- $q_0 = (q'_A, q'_B)$;
- $F = \{(x, y) \in Q \mid x \in F'_A \text{ ou } y \in F'_B\}$.

Assim, como foi possível construir $M_{A \cup B}$, podemos dizer que a classe de linguagens regulares é fechada sob a operação de união ■

Sumário

Não-determinismo

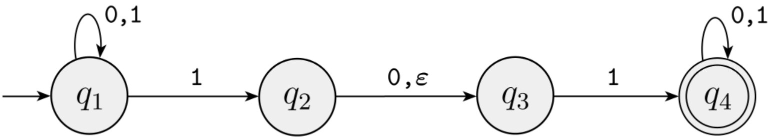


FIGURA 1.27
O autômato finito não-determinístico N_1

Não-determinismo

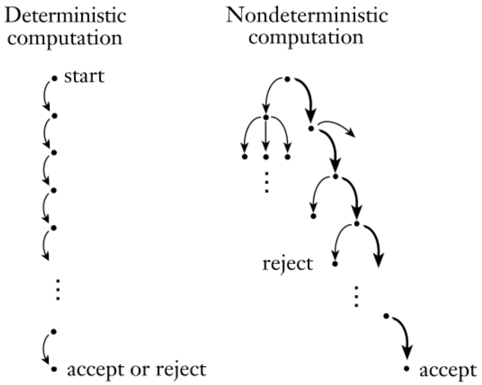


FIGURA 1.28
Computações determinísticas e não-determinísticas com um ramo de aceitação

Não-determinismo

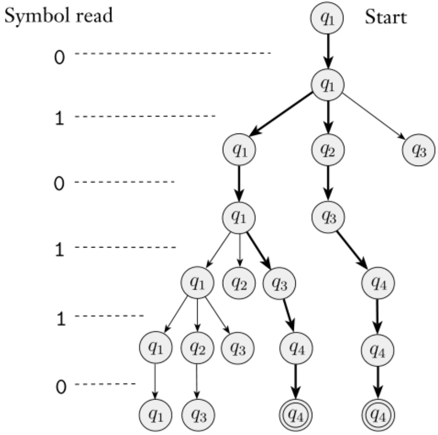


FIGURA 1.29
A computação de N_1 sobre a entrada 010110

Não-determinismo

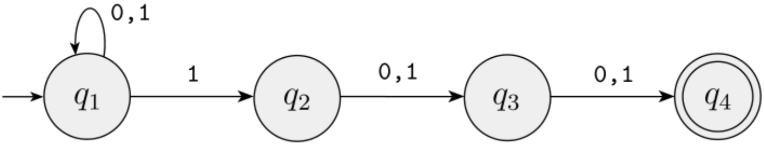


FIGURA 1.31
O AFN N_2 que reconhece A

Não-determinismo

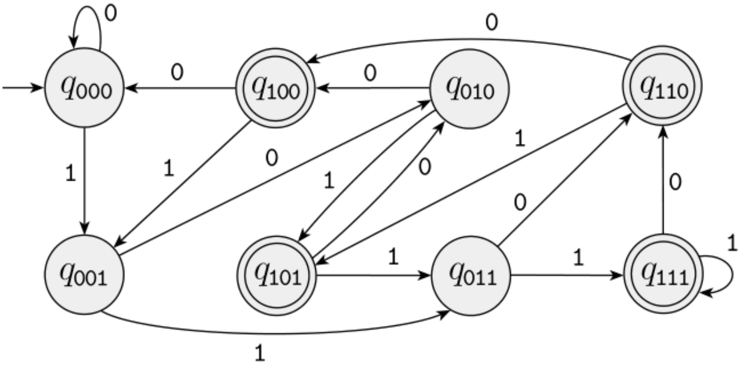


FIGURA 1.32
Um AFD que reconhece A

Não-determinismo

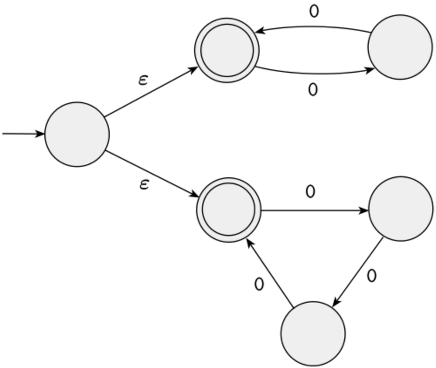


FIGURA 1.34
O AFN N_3

Sumário

Não-determinismo

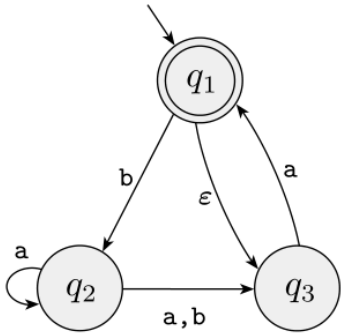


FIGURA 1.36
O AFN N_4

Não-determinismo

DEFINIÇÃO 1.37

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

1. Q é um conjunto finito de estados,
2. Σ é um alfabeto finito,
3. $\delta\colon Q \times \Sigma_\varepsilon \longrightarrow \mathcal{P}(Q)$ é a função de transição,
4. $q_0 \in Q$ é o estado inicial, e
5. $F \subseteq Q$ é o conjunto de estados de aceitação.

UFG
Regional Jataí

Descrição Formal

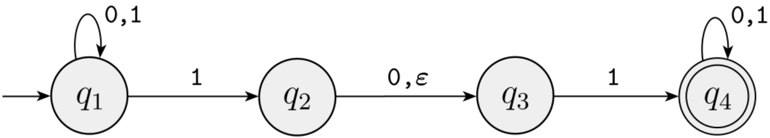


FIGURA 1.27
O autômato finito não-determinístico N_1

Descrição Formal

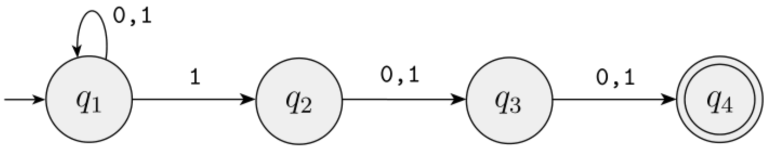


FIGURA 1.31
O AFN N_2 que reconhece A

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e suponha que $\omega = \omega_1\omega_2 \ldots, \omega_n$ seja uma cadeia

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e
suponha que $\omega = \omega_1\omega_2 \ldots, \omega_n$ seja uma cadeia
em que cada ω_i é um membro do alfabeto Σ_ϵ $(1 \leq i \leq n)$.

UFG
Regional Jataí

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e
suponha que $\omega = \omega_1\omega_2 \ldots, \omega_n$ seja uma cadeia
em que cada ω_i é um membro do alfabeto Σ_ϵ $(1 \leq i \leq n)$.
Então N **aceita** ω se uma sequência de estados $r_0, r_1, \ldots, r_n$ em Q
existe satisfazendo três condições:

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e
suponha que $\omega = \omega_1\omega_2 \ldots, \omega_n$ seja uma cadeia
em que cada ω_i é um membro do alfabeto Σ_ϵ $(1 \leq i \leq n)$.
Então N **aceita** ω se uma sequência de estados $r_0, r_1, \ldots, r_n$ em Q
existe satisfazendo três condições:

1. $r_0 = q_0$;
2. $r_{i+1} \in \delta(r_i, \omega_{i+1})$ (para $i = 0, \ldots, n-1$); e
3. $r_n \in F$.

Computação em um AFN

Definição

Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN e suponha que $\omega = \omega_1\omega_2 \ldots, \omega_n$ seja uma cadeia em que cada ω_i é um membro do alfabeto Σ_ϵ $(1 \leq i \leq n)$.
Então N **aceita** ω se uma sequência de estados $r_0, r_1, \ldots, r_n$ em Q existe satisfazendo três condições:

1. $r_0 = q_0$;
2. $r_{i+1} \in \delta(r_i, \omega_{i+1})$ (para $i = 0, \ldots, n-1$); e
3. $r_n \in F$.

Corolário

N reconhece a linguagem A, se $A = \{\omega \mid N \text{ aceita } \omega\}$.

Equivalência de AFNs e AFDs

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Equivalência de AFNs e AFDs

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Ideia da prova

- Se uma linguagem é reconhecida por um AFN, então temos de mostrar a existência de um AFD que também a reconhece;

Equivalência de AFNs e AFDs

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Ideia da prova

- Se uma linguagem é reconhecida por um AFN, então temos de mostrar a existência de um AFD que também a reconhece;
- Converter um AFN num AFD equivalente que o simule;

Equivalência de AFNs e AFDs

Teorema 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Ideia da prova

- Se uma linguagem é reconhecida por um AFN, então temos de mostrar a existência de um AFD que também a reconhece;
- Converter um AFN num AFD equivalente que o simule;
- Se k é o número de estados do AFN, então ele tem 2^k subconjuntos de estados;

Equivalência de AFNs e AFDs

Teorema 1.39

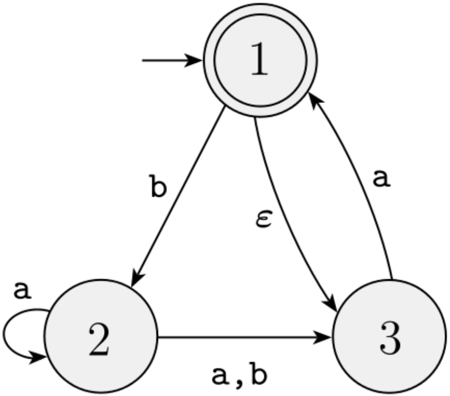
Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

Ideia da prova

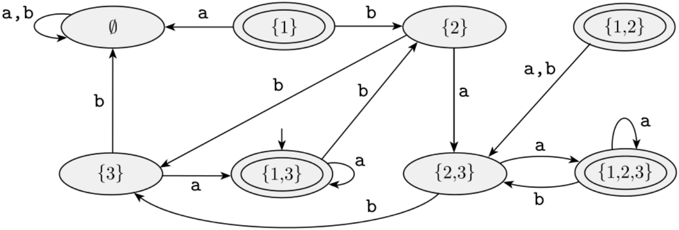
- Se uma linguagem é reconhecida por um AFN, então temos de mostrar a existência de um AFD que também a reconhece;
- Converter um AFN num AFD equivalente que o simule;
- Se k é o número de estados do AFN, então ele tem 2^k subconjuntos de estados;
- Portanto o AFD equivalente terá 2^k estados.

UFG Regional Jataí

Descrição Formal



Descrição Formal



Autômato Finito Não-determinístico

Esdras Lins Bispo Jr.
bispojr@ufg.br

Linguagens Formais e Autômatos
Bacharelado em Ciência da Computação

20 de novembro de 2017