Linguagens formais, automâtos e computabilidade

Autômatos finitos

Marco Aurélio Graciotto Silva

Março de 2015

Agenda

Definição

Um autômato finito não determinístico AFND é uma quíntupla $A =< \Sigma, S, S_0, \delta, F >$ em que:

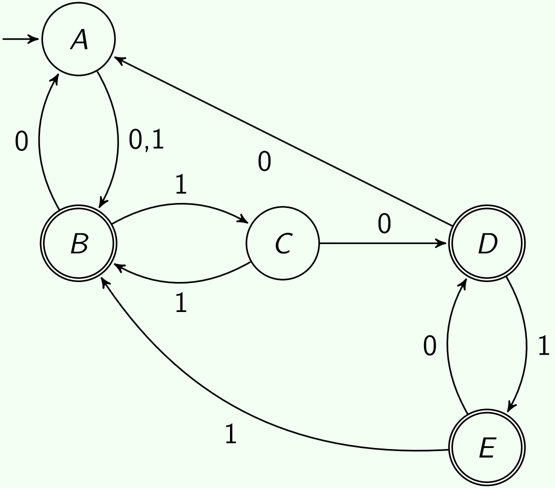
- Σ é o alfabeto de entrada,
- S é o conjunto finito de estados,
- S_0 é o **conjunto de estados iniciais**, tal que $S_0 \subseteq S$,
- δ é a função de transição de estados, tal que $\delta : S \times \Sigma_\epsilon \to 2^S$,
- F é o conjunto de estados finais, tal que $F \subseteq S$.

Diferenças com relação a AFD

- Agora é um conjunto de estados iniciais.
- Função de transição de estados, para um mesmo símbolo, pode resultar em vários estados.
- Função de transição de estados pode ser para ϵ.

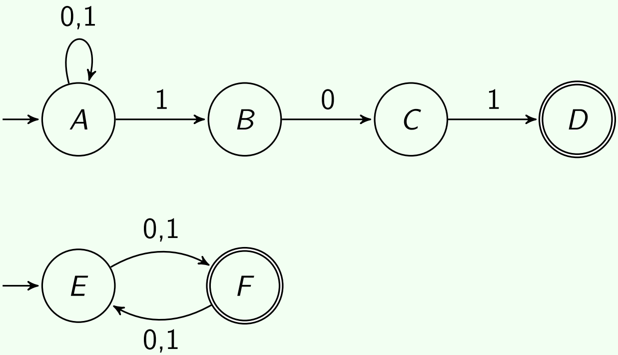
AFD - Exemplo

$L = \{xa \in \{0,1\}^* \mid a = 101 \vee |xa| \bmod 2 = 1\}$



AFND - Exemplo

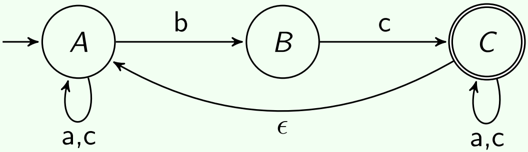
$L = \{xa \in \{0,1\}^* \mid a = 101 \vee |xa| \bmod 2 = 1\}$



Transições ϵ

Transição que ocorre sem que nenhum símbolo de entrada seja processado (ou, melhor, é processado o símbolo ϵ).

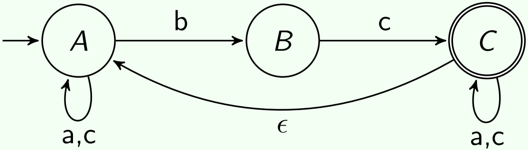
Exemplo: $L = \{(x^n y^m z^o)^p \in \{a,b,c\}^* \mid x, z \in \{a,c\} \wedge y = bc\} \wedge p > 0$



Closure (fecho)

O fecho (*closure*) de um estado $s \in S$ é o conjunto de estados que podem ser alcançados a partir de transições ϵ.

Exemplo: $L = \{a^n x^m b^o \in \{a,b,c\}^* \mid a, b \in \{a,c\} \wedge x = bc\}$



- $CL(A) = \{A\}$
- $CL(B) = \{B\}$
- $CL(C) = \{C, A\}$

Função de transição estendida

Dado o AFND $A =< \Sigma, S, S_0, \delta, F >$, a Função de Transição Estendida $\overline{\delta}$ é definida como $\overline{\delta} : S \times \Sigma^* \rightarrow S$, tal que:

- $\forall s \in S, \overline{\delta}(s, \epsilon) = CL(s)$
- $(\forall s \in S) \wedge (\forall a \in \Sigma), \overline{\delta}(s, a) = CL(\delta(s, a))$
- $(\forall s \in S) \wedge (\forall a \in \Sigma) \wedge (\forall x \in \Sigma^*), \overline{\delta}(s, ax) = \overline{\delta}(\delta(s, a), x)$

LFAC

UTFPR-CM

Autômatos finitos não determinísticos

Equivalência entre AFD e AFND

Propriedades de autômatos finitos

Exercícios

Construa um AFND para cada uma das seguintes linguagens:

- $\{xa \in \{0,1\}^* \mid a = 00\}$ (usar apenas três estados)
  - Tradução: Palavras no alfabeto $\{0,1\}^*$ que terminam com 00.
- $0^*1^*0^+$ (usar apenas três estados)
- $\{x \in \{0,1\}^* \mid$ 1011 ou 111 são subpalavras de x$\}$

Substituição das transições ϵ

- Gerar nova função de transição δ_N a partir da δ_E:
  - Calcule o fecho de cada estado q do AFND: $CL(q)$.
  - Para cada estado q do AFND, calcule a nova função de transição δ_N para incluir as transições para cada símbolo do alfabeto $a \in \Sigma$ considerando os estados $p \in CL(q)$.
  - Assim temos que $\delta_N(q, a) = \bigcup_{p \in CL(q)} \delta_E(p, a)$

AFND-ϵ

	0	1	ϵ
$\{A\}$	$\{E\}$	$\{B\}$	$\emptyset$
$\{B\}$	$\emptyset$	$\{C\}$	$\{D\}$
$\{C\}$	$\emptyset$	$\{D\}$	$\emptyset$
* $\{D\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\{E\}$	$\{F\}$	$\emptyset$	$\{B, C\}$
$\{F\}$	$\{D\}$	$\emptyset$	$\emptyset$

Cálculo da nova função de transição

Considerar o fecho $CL(s)$ e definir função de transição em função da união dos estados resultantes da aplicação da função antiga para cada estado contido no fecho.

Conversão de um AFND com transições ϵ em um AFND

AFND-ϵ

	0	1	ϵ
$\{A\}$	$\{E\}$	$\{B\}$	$\emptyset$
$\{B\}$	$\emptyset$	$\{C\}$	$\{D\}$
$\{C\}$	$\emptyset$	$\{D\}$	$\emptyset$
* $\{D\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\{E\}$	$\{F\}$	$\emptyset$	$\{B, C\}$
$\{F\}$	$\{D\}$	$\emptyset$	$\emptyset$

Closures

- $CL(A) = \{A\}$
- $CL(B) = \{B, D\}$
- $CL(C) = \{C\}$
- $CL(D) = \{D\}$
- $CL(E) = \{E, B, C\} = \{E, B, D, C\}$
- $CL(F) = \{F\}$

Cálculo da nova função de transição

Considerar o fecho $CL(s)$ e definir função de transição em função da união dos estados resultantes da aplicação da função antiga para cada estado contido no fecho.

Conversão de um AFND com transições ϵ em um AFND

AFND-ϵ

	0	1	ϵ
$\{A\}$	$\{E\}$	$\{B\}$	$\emptyset$
$\{B\}$	$\emptyset$	$\{C\}$	$\{D\}$
$\{C\}$	$\emptyset$	$\{D\}$	$\emptyset$
* $\{D\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\{E\}$	$\{F\}$	$\emptyset$	$\{B, C\}$
$\{F\}$	$\{D\}$	$\emptyset$	$\emptyset$

AFND

	0	1
$\{A\}$	E	B
$\{B\}$	$\emptyset$	C
$\{C\}$	$\emptyset$	D
$\{D\}$	$\emptyset$	$\emptyset$
$\{E\}$	F	C, D
$\{F\}$	D	$\emptyset$

Closures

- $CL(B) = \{B, D\}$
- $CL(E) = \{E, B, C\} = \{E, B, D, C\}$

Conversão de um AFND com transições ϵ em um AFND

AFND-ϵ

	0	1	ϵ
$\{A\}$	$\{E\}$	$\{B\}$	$\emptyset$
$\{B\}$	$\emptyset$	$\{C\}$	$\{D\}$
$\{C\}$	$\emptyset$	$\{D\}$	$\emptyset$
* $\{D\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\{E\}$	$\{F\}$	$\emptyset$	$\{B, C\}$
$\{F\}$	$\{D\}$	$\emptyset$	$\emptyset$

AFND

	0	1
$\{A\}$	E	B
* $\{B\}$	$\emptyset$	C
$\{C\}$	$\emptyset$	D
* $\{D\}$	$\emptyset$	$\emptyset$
* $\{E\}$	F	C, D
$\{F\}$	D	$\emptyset$

Ajuste dos estados finais

Se o fecho do estado inclui um estado que era final, então o estado também é final.

Closures

- $CL(B) = \{B, D\}$
- $CL(E) = \{E, B, C\} = \{E, B, D, C\}$

Tabuleiro

Considere um tabuleiro de 3x3. Cada quadrado do tabuleiro é um estado e a cor de cada posição (vermelho/***red*** ou preto/***b**lack*) é o alfabeto de entrada. A posição inicial é a do canto superior esquerdo do tabuleiro. O estado final é a posição inferior direita do tabuleiro.

Movimentos possíveis:

- Sair do estado atual para um estado adjacente vermelho (r)
- Sair do estado atual para um estado adjacente preto (b)

Tabuleiro

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
1	2, 4	5

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2,4}		
{5}		

Para cada classe de equivalência que consta na tabela, vamos definir a função em relação aos símbolos de entrada.

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}
{2, 4, 6, 8}		
{1, 3, 5, 7}		

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}
{2, 4, 6, 8}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 5, 7}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}
{2, 4, 6, 8}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 5, 7}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 7, 9}		
{1, 3, 5, 7, 9}		

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}
{2, 4, 6, 8}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 5, 7}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 7, 9}	{2, 4, 6, 8}	{5}
{1, 3, 5, 7, 9}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}

Conversão de um AFND em um AFD

	r	b
1	2, 4	5
2	4, 6	1, 3, 5
3	2, 6	5
4	2, 8	1, 5, 7
5	2, 4, 6, 8	1, 3, 7, 9
6	2, 8	3, 5, 9
7	4, 8	5
8	4, 6	5, 7, 9
* 9	6, 8	5

Tabuleiro AFD

	r	b
{1}	{2, 4}	{5}
{2, 4}	{2, 4, 6, 8}	{1, 3, 5, 7}
{5}	{2, 4, 6, 8}	{1, 3, 7, 9}
{2, 4, 6, 8}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
{1, 3, 5, 7}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}
* {1, 3, 7, 9}	{2, 4, 6, 8}	{5}
* {1, 3, 5, 7, 9}	{2, 4, 6, 8}	{1, 3, 5, 7, 9}

Estados finais

Toda classe de equivalência que possui um estado que pertence ao conjunto original de estados finais também é um estado final.

Conversão de um AFND em um AFD

$\delta(a, s) = \emptyset$

E quando a transição aponta para o estado vazio, o que fazemos?

AF com transição para o $\emptyset$

	0	1
A	$\{E\}$	$\{B\}$
* B	$\emptyset$	$\{C\}$
C	$\emptyset$	$\{D\}$
* D	$\emptyset$	$\emptyset$
* E	$\{F\}$	$\{C\}$
F	$\{D\}$	$\emptyset$

AF com transição para o $\emptyset$

	0	1
$\{A\}$	$\{E\}$	$\{B\}$
$* \{B\}$	$\emptyset$	$\{C\}$
$\{C\}$	$\emptyset$	$\{D\}$
$* \{D\}$	$\emptyset$	$\emptyset$
$* \{E\}$	$\{F\}$	$\{C\}$
$\{F\}$	$\{D\}$	$\emptyset$

AFD

	0	1
$\{A\}$	$\{E\}$	$\{B\}$
$* \{B\}$	$\{T\}$	$\{C\}$
$\{C\}$	$\{T\}$	$\{D\}$
$* \{D\}$	$\{T\}$	$\{T\}$
$* \{E\}$	$\{F\}$	$\{C\}$
$\{F\}$	$\{D\}$	$\{T\}$
$\{T\}$	$\{T\}$	$\{T\}$

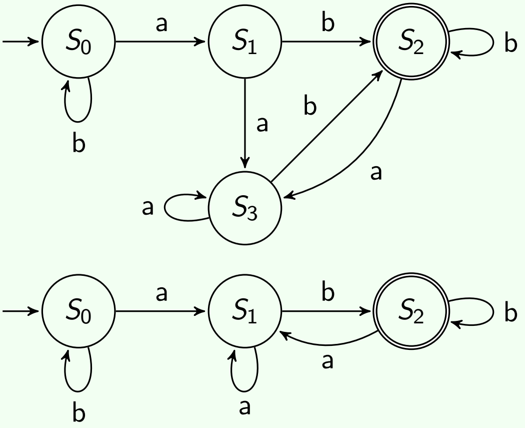
$\delta(a, s) = \emptyset$

Adicionar *trap state* (estado armadilha).

- Todas as transições originalmente para o $\emptyset$ vão para ele.
- O *trap state* define que, para qualquer símbolo do alfabeto, a transição é feita para ele mesmo.
- Obviamente, o *trap state* não é um estado final.

Dados dois autômatos finitos determinísticos (AFD) A e B, dizemos que A é equivalente a B se e somente se $L(A) = L(B)$.

Exemplo



Condições para equivalência

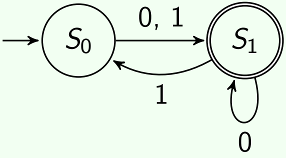
Condições

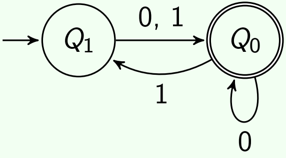
- Para que dois autômatos sejam equivalentes, é necessário que cada estado s do autômato A seja equivalente a um conjunto de estados S' do autômato B.
- Além disso, o comportamento dos autômatos devem ser idênticos (considerando esta relação).

Definição

Dados dois AFDs $A =< \Sigma, S_A, S_{0A}, \delta_A, F_A >$ e $B =< \Sigma, S_B, S_{0B}, \delta_B, F_B >$, deve existir uma função $\mu : S_A \to S_B$ tal que:

- $\mu(S_{0A}) = S_{0B}$
- $(\forall s \in S_A), s \in F_A \Leftrightarrow \mu(s) \in F_B$
- $(\forall s \in S_A)(\forall a \in \Sigma), \mu(\delta_A(s, a)) = \delta_B(\mu(s), a)$

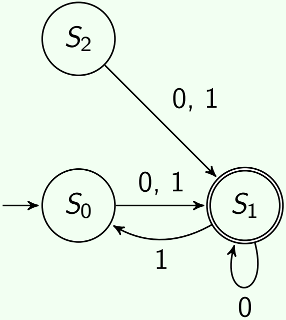


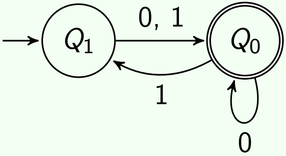


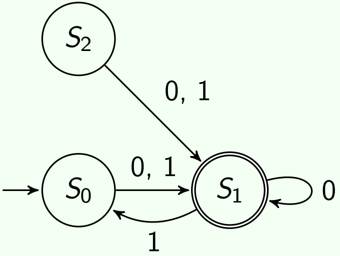
Explicação

A função μ apenas renomeia os estados S_0 para Q_1 e S_1 para Q_0.

Equivalência
Exemplo trivial 2 (estado inalcançável)







Definição

Dado o AFD $A = < \Sigma, S, S_0, \delta, F >$, diz-se que um estado $S_r \in S$ é alcancável se e somente se $\exists x \in \Sigma^* \mid \overline{\delta}(S_0, x) = S_r$.
Caso contrário, dizemos que o estado S_r é inalcançável.

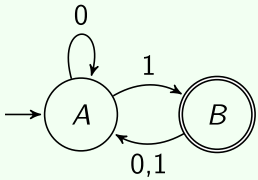
Um AFD no qual todos os estados são alcançáveis é chamado de conexo.

Algoritmo para determinar equivalência

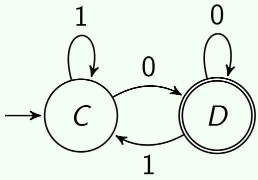
1. Faça o produto dos autômatos finitos determinísticos. Para cada par de estados $\{[q, r] \mid q \in S_A \wedge r \in S_B\}$, temos, para cada símbolo $a \in \Sigma$:
   - $\delta([q, r], a) = [\delta_A(q, a), \delta_B(r, a)]$
2. Estado inicial será o par de estados tal que seus elementos sejam estados iniciais no AFD original.
3. Estados finais serão os pares de estados em que apenas um de seus elementos seja um estado final no AFD original.
4. $L = M$ sse $L \times M = \emptyset$.

Equivalência
Exemplo

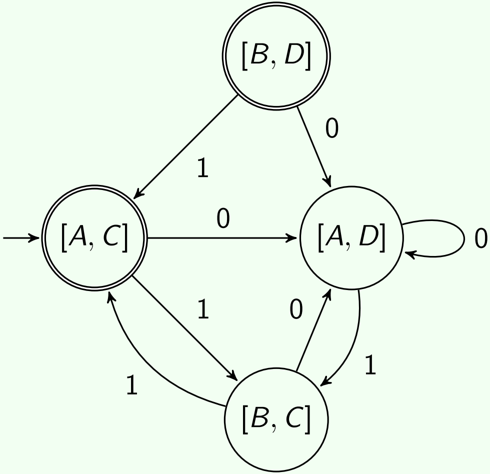
AFD A



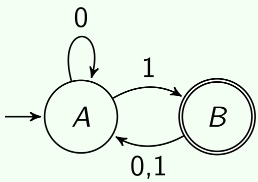
AFD B



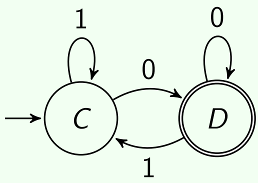
AFD $A \times B$



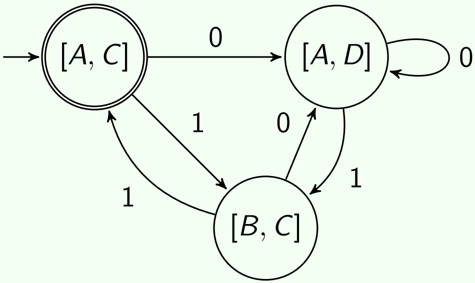
AFD A



AFD B



AFD $A \times B$



O estado $[B, D]$ é inalcançável e pode ser retirado.

$L(A \times B) \neq \emptyset$, logo A e B não são equivalentes.

Algoritmo ingênuo para determinar equivalência

1. Minimize os autômatos.
2. Se após a minimização eles não tiverem a mesma quantidade de estados, eles não são equivalentes.
3. Caso tenham a mesma quantidade de estados, deve existir uma função μ que defina o homomorfismo entre os autômatos.

AFD minimal

Definição

Um AFD $A =< \Sigma, S, S_0, \delta, F >$ é dito minimal se para qualquer AFD $B =< \Sigma, S_B, S_{0B}, \delta_B, F_B >$ tal que $A \equiv B$, temos que $|S| \leq |S_B|$.

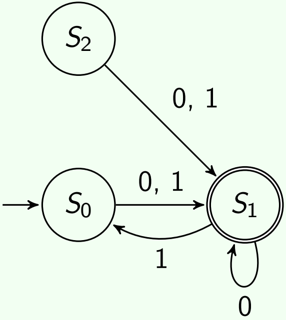
Em outras palavras

- Autômatos precisam ter o mesmo comportamento (reconhecer a mesma linguagem).
- A quantidade de estados do autômato minimal é menor ou igual à quantidade de estados de qualquer outro autômato equivalente.

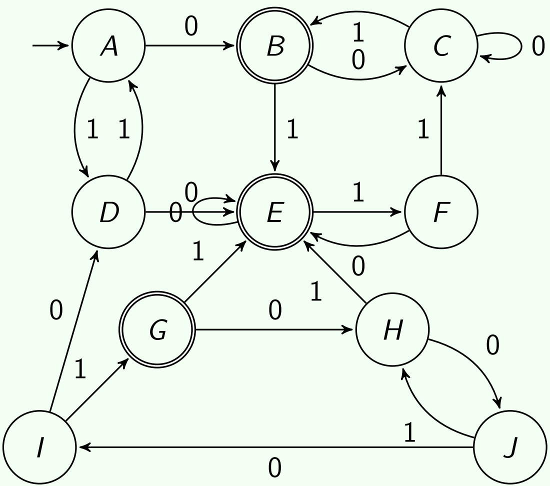
1. Transformar o AF em um AFD.
2. Retirar estados inalcançáveis.
3. Identificar e unir estados equivalentes.

1. Defina o conjunto S_c de estados alcançáveis.
2. Por definição, o estado inicial S_0 sempre é alcançável. Assim, inclua-o em S_c.
3. Aplique a função de transição δ do AFD para os estados alcançáveis. O estado obtido deve ser incluído em S_c.
4. Repita o passo anterior até que nenhum estado seja adicionado em S_c.

Exemplo 1

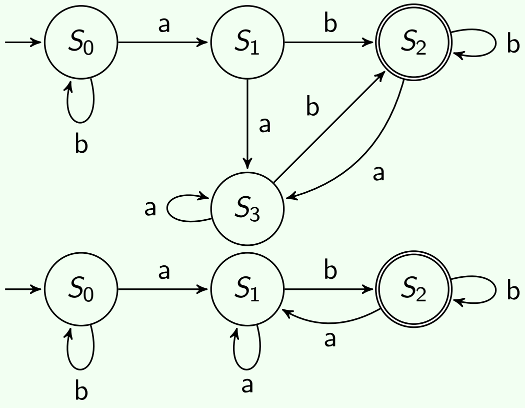


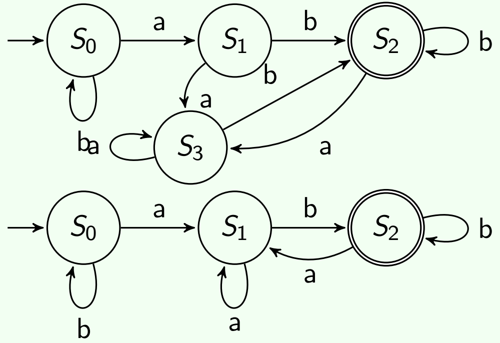
Exemplo 2



Estados equivalentes

Um estado é equivalente a outro estado se ambos desempenham o mesmo comportamento para o reconhecimento de uma string.





Estados S_1 e S_3 são equilvantes: não tem como diferenciá-los externamente: tudo o que for reconhecido a partir de S_1 também será reconhecido a partir de S_3 e vice-versa.

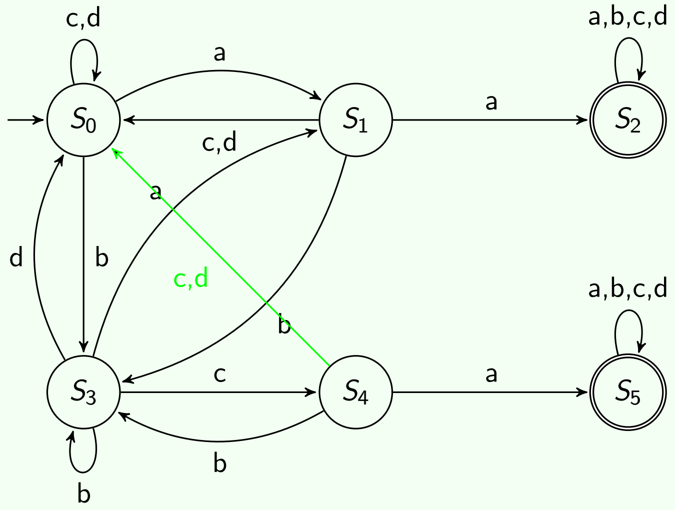
Definição

Dado o AFD $A = < \Sigma, S, S_0, \delta, F >$, diz-se que dois estados $s, t \in S$ são equivalentes se e somente se $\forall x \in \Sigma^*, \overline{\delta}(s, x) \in F \Leftrightarrow \overline{\delta}(t, x) \in F$.

Classe de equivalência

Estados que são equivalentes formam uma classe de equivalência.

Minimização de AFD
Identificação de estados equivalentes



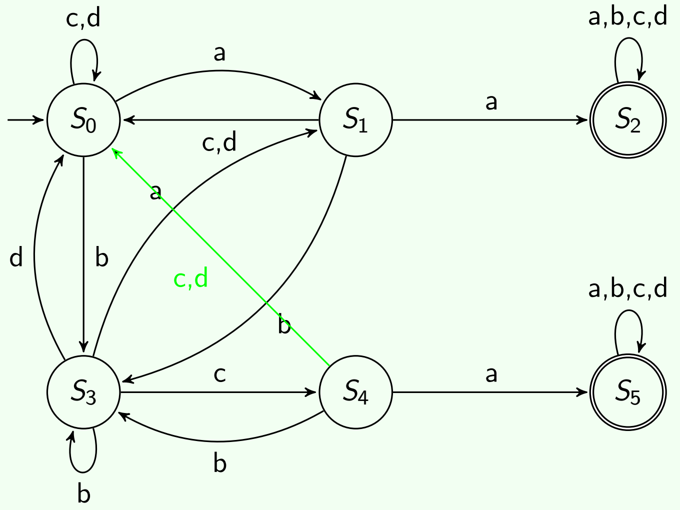
LFAC
UTFPR-CM
Autômatos finitos não determinísticos
Equivalência entre AFD e AFND
Propriedades de autômatos finitos
Equivalência
Minimização
Exercícios

AFD reduzido

Um AFD é dito reduzido quando a cardinalidade de suas classes de equivalência é 1.

Transformar um AFD em AFD reduzido

1. Identifique as classes de equivalência.
2. Transforme cada classe de equivalência em um estado.
   - Classe de equivalência que possui o estado inicial é o estado inicial do AFD.
   - Classes de equivalência que possuem quaisquer dos estados finais é estado final do AFD.
3. Defina δ de modo que $\delta([S_{eq}], a) = \delta(s, a)$, para $s \in [S_{eq}]$.


c,d
a
a,b,c,d
S_0
S_1
a
S_2
c,d
a
d
b
c,d
a,b,c,d
b
c
S_3
S_4
a
S_5
b
b

Minimização de AFD

Exemplo: Identificação de estados equivalentes

	S_5	S_4	S_3	S_2	S_1
S_0					
S_1					–
S_2				–	–
S_3			–	–	–
S_4		–	–	–	–

	S_5	S_4	S_3	S_2	S_1
S_0	0			0	
S_1	0			0	–
S_2		0	0	–	–
S_3	0		–	–	–
S_4	0	–	–	–	–

Distinguir com ϵ

- O primeiro passo é distinguir os pares de estados em relação ao ϵ.
- Na tabela estão marcados com 0 aqueles que foram distinguidos por nesta primeira tentativa:
  - Pares em que **um** (e apenas um!) dos estados é um estado de aceitação.

	S_5	S_4	S_3	S_2	S_1
S_0	0	1_a	1_c	0	1_a
S_1	0		1_a	0	–
S_2		0	0	–	–
S_3	0	1_a	–	–	–
S_4	0	–	–	–	–

Distinguir com palavra de tamanho 1

- O segundo passo é distinguir os pares de estados em relação a palavras de tamanho 1.
- Na tabela estão marcados com 1 aqueles que foram distinguidos por palavras de tamanho 1, seguido do símbolo utilizado para distinguir.

	S_5	S_4	S_3	S_2	S_1
S_0	0	1_a	1_c	0	1_a
S_1	0		1_a	0	–
S_2		0	0	–	–
S_3	0	1_a	–	–	–
S_4	0	–	–	–	–

Estados indistinguíveis

Os estados S_2 e S_5 são indistinguíveis. Observe as transições:

- $\delta([S_2, S_5], a) = [S_2, S_5]$
- $\delta([S_2, S_5], b) = [S_2, S_5]$
- $\delta([S_2, S_5], c) = [S_2, S_5]$
- $\delta([S_2, S_5], d) = [S_2, S_5]$

Em todos os casos, eles vão para o mesmo estado, não sendo possível obter uma decisão distinta para quaisquer palavras.

	S_5	S_4	S_3	S_2	S_1
S_0	0	1_a	1_c	0	1_a
S_1	0		1_a	0	–
S_2		0	0	–	–
S_3	0	1_a	–	–	–
S_4	0	–	–	–	–

Estados indistinguíveis

Os estados S_1 e S_4 são indistinguíveis. Observe as transições:

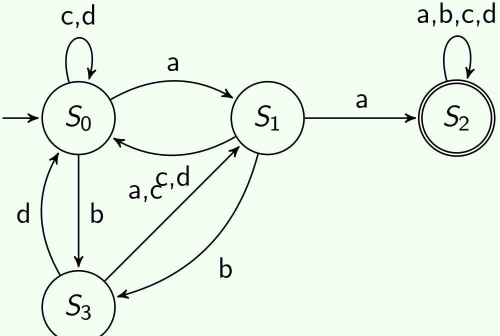
- $\delta([S_1, S_4], a) = [S_2, S_5]$
- $\delta([S_1, S_4], b) = [S_3, S_3]$
- $\delta([S_1, S_4], c) = [S_0, S_0]$
- $\delta([S_1, S_4], d) = [S_0, S_0]$

Em todos os casos, ou eles vão para o mesmo estado ou vão para estados que não conseguimos distinguir (no caso, $[S_2, S_5]$).

Exemplo

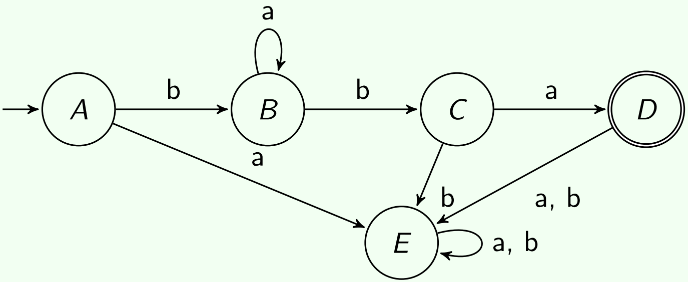
Estados equivantes:

- $[S_0] = \{S_0\}$
- $[S_1] = [S_4] = \{S_1, S_4\}$
- $[S_2] = [S_5] = \{S_2, S_5\}$
- $[S_3] = \{S_3\}$



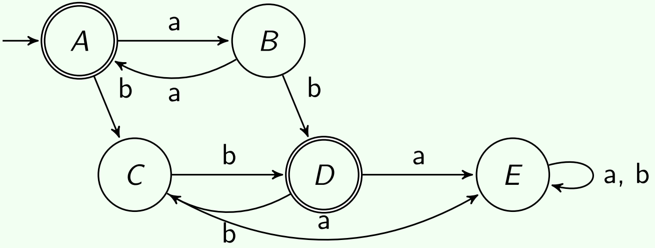
AFD $A = \{ba^n ba | n \geq 0\}$

AFD $A = \{ba^nba | n \geq 0\}$



AFD $A = \{a^m b^n | m, n \geq 0 \wedge (m + n) \text{ mod } 2 == 0\}$

AFD $A = \{a^m b^n | m, n \geq 0 \wedge (m+n) \bmod 2 == 0\}$



Desafio

Considere o jogo a seguir. Uma bola de gude é solta na abertura A ou B. As alavancas x, y e z direcionam a bola para a esquerda ou para a direita e, sempre após uma bola passar pela alavanca, a alavanca mude de estado. Em outras palavras, se a bola foi desviada para a esquerda no momento t_i, no momento t_{i+1} a bola será desviada para a direita.

Desafio

Modele este jogo com um autômato finito determinístico. Considere que uma bola de gude que passa por A como o valor 0 na entrada e uma bola de gude que passa por B como o valor 1. A palavra (sequência de bolas de gude) será aceita se a última bola de gude sair pela abertura D.

Exercícios

Desafio

```
     | A |        | B |
     |   |        |   |
     | x |        | y |
    /  /  \      /  /  \
   /       \    /       \
  /   /\    \  /    /\   \
 /   /  \    \/    /  \   \
/   /    \  z /   /    \   \
\   \    /   /    \    /   /
 \   \  /    /\    \  /   /
  \   \/    /  \    \/   /
   \       /    \       /
    \     /      \     /
     |   |        |   |
     |   |        |   |
     | C |        | D |
```