AUTÓMATOS E LINGUAGENS FORMAIS

(LCC/LMAT)

2. Autómatos Finitos

Departamento de Matemática

Universidade do Minho

2022/2023

Os autómatos finitos são máquinas de estados rudimentares que, simultaneamente, constituem um modelo de computação básico e uma ferramenta para o estudo das linguagens regulares.

Definição

Um autómato (finito) é um quíntuplo $\mathcal{A} = (Q, A, \delta, i, F)$ onde

1. Q é um conjunto finito não vazio, chamado o conjunto de estados de $\mathcal{A}$;
2. A é um alfabeto finito, chamado o alfabeto (de entrada) de $\mathcal{A}$;
3. $\delta : Q \times A \rightarrow \mathcal{P}(Q)$ é uma função, designada a função transição de $\mathcal{A}$. Cada triplo (p, a, q), em que $p, q \in Q$ e $a \in A$ são tais que $q \in \delta(p, a)$, diz-se uma transição de $\mathcal{A}$;
4. $i \in Q$ é dito o estado inicial de $\mathcal{A}$;
5. $F \subseteq Q$ é dito o conjunto de estados finais de $\mathcal{A}$.

Habitualmente, notaremos autómatos finitos por: $\mathcal{A}, \mathcal{B}, \mathcal{A}', \mathcal{A}_1, \ldots$.

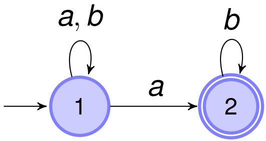
Observação

Um autómato é habitualmente representado por um grafo orientado com arestas etiquetadas, cujos vértices corresponderão aos estados do autómato e onde cada aresta corresponderá a uma transição do autómato, sendo a respetiva etiqueta pela letra envolvida na transição.

Por exemplo, o autómato $\mathcal{A} = (\{1,2\}, \{a,b\}, \delta, 1, \{2\})$, com δ dada por

δ	1	2
a	$\{1,2\}$	$\emptyset$
b	$\{1\}$	$\{2\}$

é representado pelo grafo seguinte (onde os estados/vértices inicial e final são identificados com uma seta e com dupla circunferência,resp.):



Definição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato. A extensão a palavras da função transição δ é a função

$$\delta^* : Q \times A^* \to \mathcal{P}(Q)$$

definida, por recursão em palavras:

$$\delta^*(q, \epsilon) = \{q\}, \quad \text{para cada } q \in Q;$$

$$\delta^*(q, au) = \bigcup_{p \in \delta(q,a)} \delta^*(p, u), \quad \text{para cada } q \in Q,\ u \in A^* \text{ e } a \in A.$$

Por exemplo, considerando o autómato $\mathcal{A}$ do exemplo anterior:

$$\begin{aligned}\delta^*(1, a) &= \bigcup_{p \in \delta(1,a)} \delta^*(p, \epsilon) = \bigcup_{p \in \{1,2\}} \delta^*(p, \epsilon) \\ &= \delta^*(1, \epsilon) \cup \delta^*(2, \epsilon) = \{1\} \cup \{2\} = \{1, 2\}.\end{aligned}$$

Verifique que: $\delta^*(2,a) = \emptyset$; $\delta^*(1,b) = \{1\}$; $\delta^*(2,b) = \{2\}$; $\delta^*(1,ab) = \{1,2\}$.

Proposição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato. Para todo $q \in Q$:

1. $\delta^*(q, a) = \delta(q, a)$, para qualquer $a \in A$;
2. $\delta^*(q, u \cdot v) = \bigcup_{p \in \delta^*(q,u)} \delta^*(p, v)$, para quaisquer $u, v \in A^*$;
3. $\delta^*(q, ua) = \bigcup_{p \in \delta^*(q,u)} \delta(p, a)$, para quaisquer $u \in A^*$, $a \in A$.

Demonstração:

A primeira parte segue facilmente das definições.

A segunda parte segue por indução na palavra u.

A terceira parte é consequência das duas partes anteriores.

Observação

Devido à primeira parte da proposição anterior, muitas vezes, para denotar a extensão a palavras de uma função transição δ, escreveremos simplesmente δ (em vez de δ^*).

Definição

Seja $\mathcal{A}$ um autómato. Um caminho em $\mathcal{A}$ é uma sequência finita de transições de $\mathcal{A}$ da forma:

$$(q_0, a_1, q_1), (q_1, a_2, q_2), \ldots, (q_{n-1}, a_n, q_n)$$

dizendo-se que:

- q_0 é a sua origem ou início e q_n é o seu destino ou fim;
- a palavra $a_1 a_2 ... a_n$ é a etiqueta do caminho.

Observação

Um caminho num autómato corresponde exatamente a um caminho no grafo correspondente, podendo representar-se, simplificadamente, por:

$$q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots q_{n-1} \xrightarrow{a_n} q_n$$

uma vez que num caminho é exigida a coincidência do destino de cada transição/aresta etiquetada com a origem da seguinte.

Por exemplo, no autómato $\mathcal{A}$ do slide 3 há caminhos

$$1 \xrightarrow{a} 2 \xrightarrow{b} 2 \xrightarrow{b} 2 \qquad \text{e} \qquad 1 \xrightarrow{a} 1 \xrightarrow{b} 1 \xrightarrow{b} 1,$$

ambos com etiqueta *abb*, o 1º com origem 1 e destino 2 e o 2º com a mesma origem, mas destino 1.

Definição

O comprimento de um caminho é o comprimento da respetiva sequência de transições/arestas etiquetadas, que coincide com o comprimento da sua etiqueta.

Por exemplo, ambos os caminhos acima têm comprimento 3 (tal como a etiqueta *abb*).

Proposição

Sejam $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato, $p, q \in Q$ e u uma palavra sobre A, não vazia. Então, existem caminhos com origem p e destino q etiquetados por u se e só se $q \in \delta(p, u)$.

Demonstração: Por indução no comprimento de u.

Notação

Dado um autómato $\mathcal{A} = (Q, A, \delta, i, F)$, dados estados $p, q \in Q$ e dada uma palavra $u \in A^*$, a notação

$$p \xrightarrow{u} q$$

significará: $q \in \delta(p, u)$.

Face à proposição anterior, esta notação pode também ser lida como: existem caminhos com origem p e destino q etiquetados por u.

Observação

Por vezes, dados estados p e q, é usada a terminologia q é acessível de p ou q é atingível de p quando, para alguma etiqueta u, se tem $p \xrightarrow{u} q$.

Definição

Um caminho com origem no estado/vértice inicial do autómato diz-se bem sucedido quando o seu destino é um estado/vértice final do autómato.

Por exemplo, atrás, identificámos dois caminhos etiquetados pela palavra *abb* (relativos ao autómato do slide 3), com origem no estado inicial do autómato, nomeadamente:

$$1 \xrightarrow{a} 2 \xrightarrow{b} 2 \xrightarrow{b} 2 \qquad \text{e} \qquad 1 \xrightarrow{a} 1 \xrightarrow{b} 1 \xrightarrow{b} 1 \,.$$

Portanto, o 1º caminho é bem sucedido, mas o 2º não.

Definição

Sejam $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato e $u \in A^*$. Diz-se que u é reconhecida ou aceite por $\mathcal{A}$ quando $\delta(i, u) \cap F \neq \emptyset$. Caso contrário, diremos que u não é reconhecida ou não é aceite ou é rejeitada por $\mathcal{A}$.

Por exemplo, o autómato do slide 3 tem estado inicial 1 e estado final 2 e observámos (slide 4) que $\delta(1, a) = \{1, 2\}$, $\delta(1, b) = \{1\}$ e $\delta(1, ab) = \{1, 2\}$. Assim, as palavras a e ab são reconhecidas pelo autómato, mas a palavra b é rejeitada.

Observação

1. ϵ é reconhecida por um autómato se e só se o estado inicial é também estado final.
2. Uma palavra u não vazia é reconhecida por um autómato se e só se existe pelo menos um caminho bem sucedido com etiqueta u.

Por exemplo, $1 \xrightarrow{a} 1 \xrightarrow{a} 1 \xrightarrow{a} 2$ é um caminho bem sucedido no autómato do slide 3. Logo, a palavra aaa é reconhecida.

Definição

A linguagem reconhecida ou aceite por um autómato $\mathcal{A}$ é o conjunto das palavras reconhecidas por $\mathcal{A}$, sendo notada por $L(\mathcal{A})$.

Consideremos, de novo, o autómato $\mathcal{A}$ do slide 3. Tem-se que $L(\mathcal{A})$ é a linguagem

$$L = \{uab^n : u \in \{a,b\}^* \wedge n \in \mathbb{N}_0\} .$$

Por um lado, pode provar-se

(i) $\forall u \in \{a,b\}^* .\, 1 \in \delta(1,u)$ (indução em u) e

(ii) $\forall n \in \mathbb{N}_0 .\, 2 \in \delta(2,b^n)$ (indução em n) ,

donde se pode deduzir

$$\forall u \in \{a,b\}^*, n \in \mathbb{N}_0 .\, 2 \in \delta(1,uab^n) ,$$

(porquê?). Consequentemente, que $L \subseteq L(\mathcal{A})$.

Por outro um lado, prova-se por indução em x que

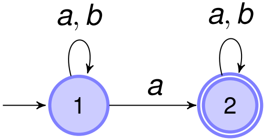
$$\forall x \in \{a,b\}^* .\, (\, 1 \xrightarrow{x} 2 \Rightarrow \exists u \in \{a,b\}^*, n \in \mathbb{N}_0 .\, x = uab^n \,) ,$$

donde, $L(\mathcal{A}) \subseteq L$.

Definição

Dois autómatos $\mathcal{A}$ e $\mathcal{B}$ dizem-se equivalentes quando reconhecem a mesma linguagem, ou seja, quando $L(\mathcal{A}) = L(\mathcal{B})$.

O autómato $\mathcal{A}$ do slide 3 é equivalente, por exemplo, ao autómato

$\mathcal{B} =$ 

Por um lado, pode provar-se que $L(\mathcal{B}) = A^* aA^*$ (com $A = \{a, b\}$).

Por outro lado, pode provar-se que, de facto,

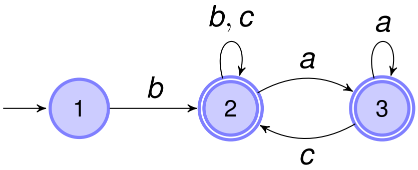
$$A^* aA^* = \{uab^n : u \in \{a, b\}^* \wedge n \in \mathbb{N}_0\},$$

que já provámos ser a linguagem reconhecida por $\mathcal{A}$.

Definição

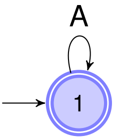
Uma linguagem L diz-se reconhecível (por autómatos finitos) quando algum autómato reconhece L, ou seja, quando existe $\mathcal{A}$ tal que $L = L(\mathcal{A})$.

Por exemplo, a linguagem $L = bA^* \setminus A^*abA^*$ das palavras sobre o alfabeto $A = \{a, b, c\}$ que começam por b e não têm o factor ab é reconhecível. De facto, L é reconhecida pelo autómato $\mathcal{A}$ seguinte:

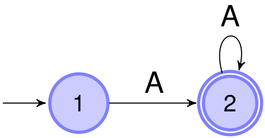


Note-se que $L = b(b+c)^*\left(aa^*c(b+c)^*\right)^*(\epsilon + aa^*)$ é uma linguagem regular.

Outros exemplos imediatos de linguagens reconhecíveis são A^* e A^+, onde A é um alfabeto qualquer. De facto estas linguagens são reconhecidas respectivamente pelos autómatos:



e



Na verdade, o resultado o fundador da teoria das linguagens regulares e dos autómatos finitos (a demonstrar posteriormente) estabelece que:

Teorema[Kleene'1954]

Uma linguagem é regular se e só se é reconhecível por autómatos finitos.

Definição

Dado um alfabeto A, $Rec(A)$ denotará o conjunto das linguagens reconhecíveis sobre A.

Dado um alfabeto A finito, já sabemos que $\mathcal{P}(A^*)$, o conjunto de todas as linguagens sobre A, é infinito não numerável. E quanto a $Rec(A)$?

Proposição

O conjunto $Rec(A)$ é infinito numerável.

Demonstração: Fixemos um conjunto numerável $\mathcal{Q} = \{q_0, q_1, q_2, \ldots\}$. Cada linguagem reconhecível é reconhecida por algum autómato finito $\mathcal{A} = (Q, A, \delta, i, F)$, de alfabeto A e conjunto de estados $Q \subseteq \mathcal{Q}$. Pode mostrar-se que estes autómatos formam um conjunto equipotente a $\mathbb{N}$ (com auxílio dos factos: (i) $\mathcal{P}_{\text{fin}}(X)$, o conjunto dos subconjuntos finitos de X, é numerável quando X é numerável; (ii) o produto cartesiano de conjuntos numeráveis é ainda numerável). Portanto, como há uma infinidade de linguagens reconhecíveis (porquê?), $Rec(A)$ também é um conjunto numerável.

Corolário

Existem linguagens que não são reconhecíveis por autómatos finitos.

O lema seguinte fornece uma condição necessária para que uma linguagem infinita seja reconhecível.

Lema da Iteração

Seja L uma linguagem reconhecível infinita, sobre um alfabeto A. Então, existe uma constante $c \in \mathbb{N}$ tal que, para toda a palavra $u \in L$ de comprimento superior ou igual a c, existem palavras $x, y, z \in A^*$ tais que:

i) $u = xyz$;

ii) $|xy| \leq c$ e $y \neq \epsilon$;

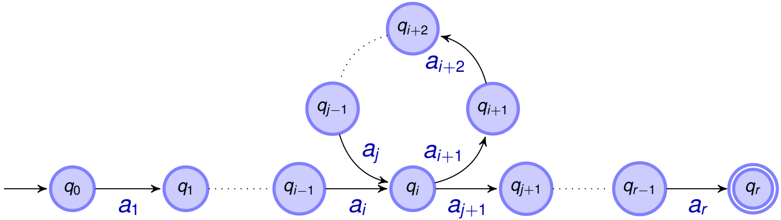
iii) $\forall k \in \mathbb{N}_0,\ xy^k z \in L$.

Demonstração: Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato que reconhece L e seja $c = |Q|$. Então, para cada $u = a_1 a_2 \cdots a_r \in L$, com $r \geq c$, existe um caminho bem sucedido de $\mathcal{A}$,

$$i = q_0 \xrightarrow{a_1} q_1 \xrightarrow{a_2} q_2 \cdots q_{r-1} \xrightarrow{a_r} q_r$$

de etiqueta u.

Dado que $r \geq c$, existem inteiros i e j, com $0 \leq i < j \leq c$, tais que $q_i = q_j$, e a palavra $a_{i+1} \cdots a_j$ é a etiqueta de um ciclo envolvendo q_i:



Sejam

$$x = \begin{cases} \epsilon & \text{se } i = 0 \\ a_1 \cdots a_i & \text{se } i \neq 0 \end{cases}, \quad y = a_{i+1} \cdots a_j, \quad z = \begin{cases} \epsilon & \text{se } j = r \\ a_{j+1} \cdots a_r & \text{se } j \neq r \end{cases}.$$

Então: i) $u = xyz$; ii) $|xy| = j \leq c$ e $y \neq \epsilon$; iii) para todo o $k \in \mathbb{N}_0$, $xy^kz \in L$, pois xy^kz é a etiqueta de um caminho bem sucedido de $\mathcal{A}$.

Por vezes, o Lema da Iteração pode ser usado para provar que uma dada linguagem não é reconhecível.

Exemplo

A linguagem $L = \{a^m b^m : m \in \mathbb{N}_0\}$ sobre o alfabeto $A = \{a, b\}$ não é reconhecível.

Demonstração: Tendo em vista uma contradição, suponhamos que L é reconhecível. Como L é infinita, pelo Lema da Iteração, existe $c \in \mathbb{N}$ tal que, para todo o $u \in L$, com $|u| \geq c$, existem palavras $x, y, z \in A^*$ tais que:

i) $u = xyz$;

ii) $|xy| \leq c$ e $y \neq \epsilon$;

iii) $\forall k \in \mathbb{N}_0,\ xy^k z \in L$.

Consideremos, em particular, a palavra $u = a^c b^c$. Como $u \in L$ e $|u| \geq c$, deduz-se de i)- iii) que $u = xyz$ para alguns $x, y, z \in A^*$ tais que $|xy| \leq c$, $y \neq \epsilon$ e

$$xy^2 z \in L. \tag{1}$$

Como $|xy| \leq c$, xy é um prefixo de a^c, donde

$$x = a^r, \quad y = a^s \quad \text{e} \quad z = a^t b^c$$

com $r, t \in \mathbb{N}_0$, $s \in \mathbb{N}$ e $r + s + t = c$, como se ilustra na seguinte figura.

a^c		b^c
x	y	z

Portanto

$$xy^2z = a^r(a^s)^2 a^t b^c = a^{r+2s+t} b^c.$$

Ora, como $s \neq 0$, tem-se

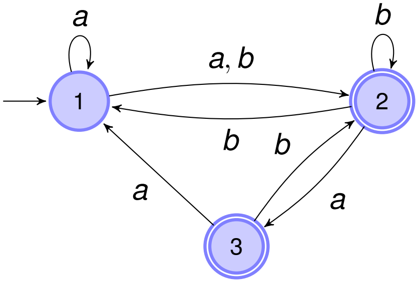
$$r + 2s + t \neq r + s + t = c$$

e, por isso, $xy^2z \notin L$. Isto contradiz (1) e, consequentemente, L não é reconhecível.

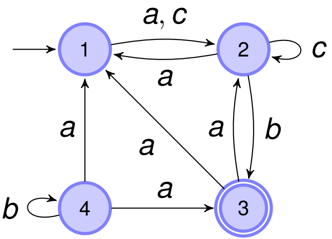
Definição

Um autómato $\mathcal{A} = (Q, A, \delta, i, F)$ diz-se completo quando para cada par $(q, a) \in Q \times A$, $\delta(q, a) \neq \emptyset$. Ou seja, $\mathcal{A}$ é completo se para cada estado $q \in Q$ e cada letra $a \in A$, existe *pelo menos* uma transição $q \xrightarrow{a} p$ de origem q e etiqueta a.

Por exemplo, o autómato seguinte, de alfabeto $A = \{a, b\}$, é completo:



O seguinte autómato $\mathcal{A}$, de alfabeto $A = \{a, b, c\}$, não é completo:

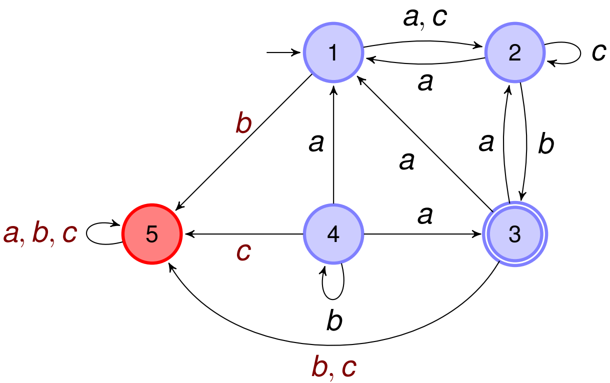


Teorema

Todo o autómato $\mathcal{A}$ é equivalente a um autómato $\mathcal{A}'$ completo.

Demonstração: Suponhamos que $\mathcal{A} = (Q, A, \delta, i, F)$ é não é completo. Seja $\mathcal{A}' = (Q \cup \{p\}, A, \delta', i, F)$, onde p é um novo estado e δ' é a extensão de δ a $Q \cup \{p\}$ tal que : se $\delta(q, a) = \emptyset$ então $\delta'(q, a) = \{p\}$; $\delta'(p, a) = \{p\}$. Então, $\mathcal{A}'$ é um autómato completo, que se prova facilmente ser equivalente a $\mathcal{A}$.

Por exemplo, a construção envolvida na demonstração anterior aplicada ao autómato $\mathcal{A}$ do exemplo anterior produz o seguinte autómato $\mathcal{A}'$, que é completo e equivalente a $\mathcal{A}$:

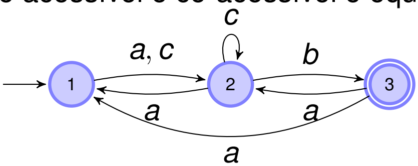


Definição

Um autómato $\mathcal{A} = (Q, A, \delta, i, F)$ diz-se:

- acessível quando, para qualquer estado $q \in Q$, existem caminhos do estado inicial i para q;
- co-acessível quando, para qualquer $q \in Q$, existem caminhos de q para algum estado final $f \in F$.

O autómato $\mathcal{A}'$ do exemplo anterior não é acessível (o estado 4 não é acessível), nem co-acessível (o estado 5 não é co-acessível).
O seguinte autómato acessível e co-acessível é equivalente a $\mathcal{A}'$:



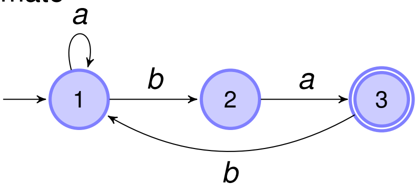
Teorema

Todo o autómato é equivalente a um autómato acessível e co-acessível.

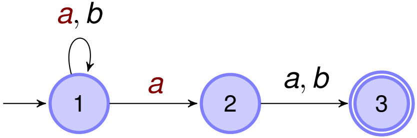
Definição

Um autómato $\mathcal{A} = (Q, A, \delta, i, F)$ diz-se determinista ou determinístico quando, para cada estado $q \in Q$ e cada letra $a \in A$, $\delta(q, a)$ tem no máximo um elemento, ou seja, existe no máximo um estado p tal que $q \xrightarrow{a} p$.

Por exemplo, o autómato



é determinista , enquanto que o seguinte não o é



Teorema

Todo o autómato $\mathcal{A}$ é equivalente a um autómato determinista $D(\mathcal{A})$.

Demonstração: Dado um autómato $\mathcal{A} = (Q, A, \delta, i, F)$, seja $D(\mathcal{A}) = (Q', A, \delta', i', F')$ o autómato tal que:

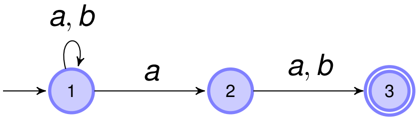
- $Q' = \mathcal{P}(Q)$;
- $\delta' : Q' \times A \to Q'$ é a função definida, para cada $X \in Q'$ e cada $a \in A$, por $\delta'(X, a) = \bigcup_{q \in X} \delta(q, a)$;
- $i' = \{i\}$;
- $F' = \{X \in Q' : X \cap F \neq \emptyset\}$.

O autómato $D(\mathcal{A})$ é determinista por construção e prova-se que $D(\mathcal{A})$ é equivalente a $\mathcal{A}$.

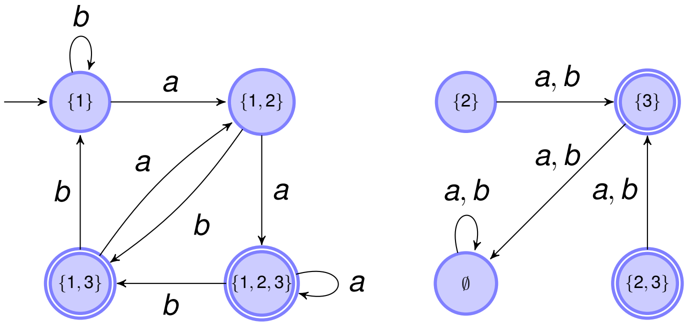
Corolário

Todas as linguagens reconhecíveis são reconhecidas por autómatos deterministas.

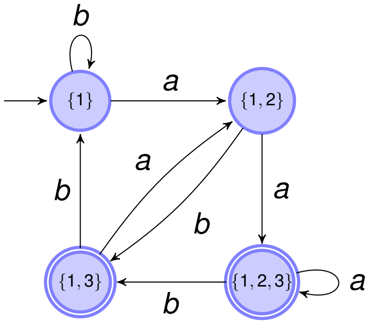
Por exemplo, consideremos o autómato não determinista $\mathcal{A}$



O autómato $D(\mathcal{A})$ é o seguinte



Se no autómato anterior $D(\mathcal{A})$ nos restringirmos aos estados acessíveis do estado inicial, obtemos o autómato $DA(\mathcal{A})$ que se segue, que é equivalente a $\mathcal{A}$ e a $D(\mathcal{A})$ e que é determinista, completo e acessível:



Teorema

Todo o autómato $\mathcal{A}$ é equivalente a um autómato determinista, completo e acessível $DA(\mathcal{A})$.

Definição

Um autómato com transições-ϵ é um quíntuplo $\mathcal{A} = (Q, A, \delta, i, F)$, que obdece às mesmas condições dos autómatos finitos, com exceção da função transição δ, que está definida em $Q \times (A \cup \{\epsilon\})$ (e não apenas em $Q \times A$).

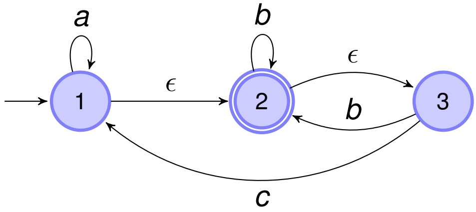
Observação

Nos autómatos com transições vazias (outra designação para as transições-ϵ), as transições e as arestas dos correspondentes grafos podem ser etiquetadas pela palavra vazia.

Observação

Os autómatos com transições vazias são também chamados assíncronos, por oposição aos autómatos sem transições vazias, que são também chamados síncronos.

O grafo seguinte representa um autómato com transições vazias:



Definição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato assíncrono. Para cada estado $q \in Q$, denotamos por $\text{fecho}_\epsilon(q)$ o conjunto formado pelo próprio q e pelos estados atingíveis a partir de q por um caminho de etiqueta ϵ.

No exemplo acima tem-se,

$$\text{fecho}_\epsilon(1) = \{1, 2, 3\}, \quad \text{fecho}_\epsilon(2) = \{2, 3\} \quad \text{e} \quad \text{fecho}_\epsilon(3) = \{3\}.$$

Teorema

Todo o autómato assíncrono $\mathcal{A}$ é equivalente a um autómato síncrono $\mathcal{A}'$.

Demonstração: Sendo $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato assíncrono qualquer, seja $\mathcal{A}' = (Q, A, \delta', i, F')$ o autómato tal que:

- $\delta' : Q \times A \to \mathcal{P}(Q)$ é a função definida, para cada $(q, a) \in Q \times A$, por

$$\delta'(q, a) = \bigcup_{p \,\in\, \mathrm{fecho}_\epsilon(q)} \delta(p, a).$$

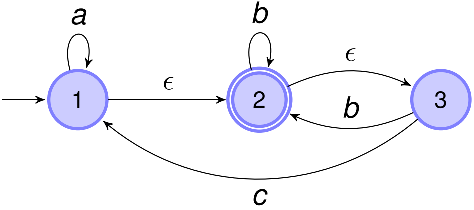
- $F' = \{q \in Q : \mathrm{fecho}_\epsilon(q) \cap F \neq \emptyset\}$.

Por construção, $\mathcal{A}'$ é um autómato síncrono e prova-se que é equivalente a $\mathcal{A}$.

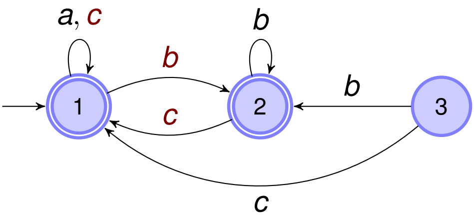
Corolário

Os autómatos com transições vazias reconhecem exatamente as mesmas linguagens que os autómatos sem transições vazias.

Relembremos o autómato com transições vazias $\mathcal{A}$ do exemplo anterior:



Vimos já que $\text{fecho}_\epsilon(1) = \{1,2,3\}$, $\text{fecho}_\epsilon(2) = \{2,3\}$ e $\text{fecho}_\epsilon(3) = \{3\}$.
O autómato síncrono $\mathcal{A}'$ descrito na demonstração do teorema anterior é:

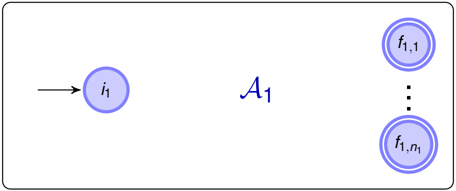


Os três lemas que se seguem serão usados para provar a parte do Teorema de Kleene que estabelece que qualquer linguagem regular é reconhecível por autómatos.

Lema

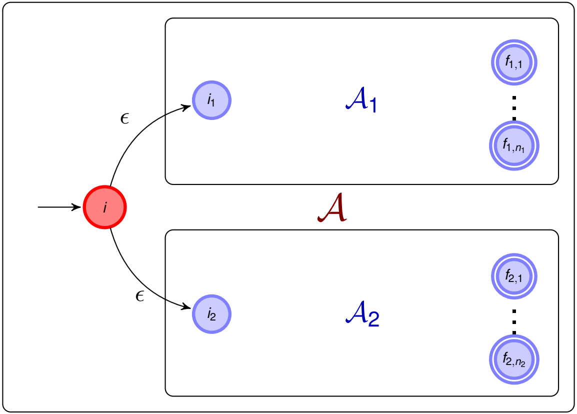
Se L_1 e L_2 são linguagens reconhecíveis, então $L_1 \cup L_2$ é uma linguagem reconhecível.

Demonstração: Seja $\mathcal{A}_1 = (Q_1, A, \delta_1, i_1, \{f_{1,1}, \ldots, f_{1,n_1}\})$ um autómato com transições vazias que reconhece L_1



e seja $\mathcal{A}_2 = (Q_2, A, \delta_2, i_2, \{f_{2,1}, \ldots, f_{2,n_2}\})$ um autómato que reconhece L_2. (Sem perda de generalidade, podemos assumir que $Q_1 \cap Q_2 = \emptyset$.)

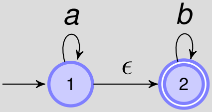
Então, o seguinte autómato $\mathcal{A} = (Q_1 \cup Q_2 \cup \{i\}, A, \delta, i, F_1 \cup F_2)$ (para $i \notin Q_1 \cup Q_2$)



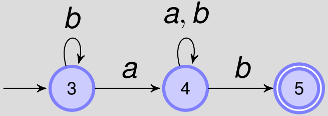
reconhece $L_1 \cup L_2$. Logo, $L_1 \cup L_2$ é uma linguagem reconhecível.

Exemplo

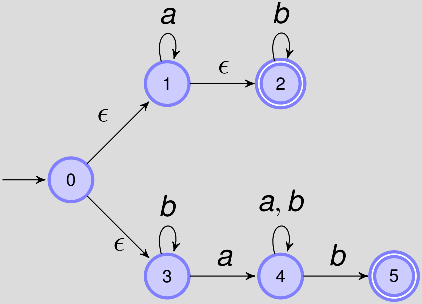
As linguagens a^*b^* e $b^*a(a+b)^*b$ são reconhecidas pelos autómatos



e



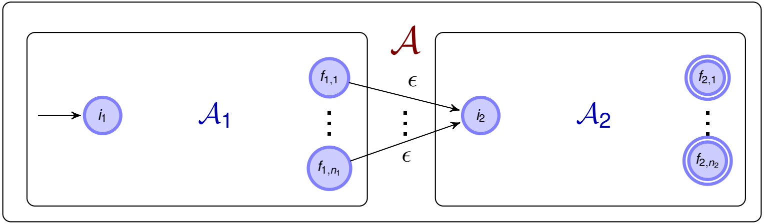
Aplicando a construção do lema anterior obtém-se o autómato que se segue, que reconhecerá a linguagem $a^*b^*+b^*a(a+b)^*b$.



Lema

Se L_1 e L_2 são linguagens reconhecíveis, então $L_1 \cdot L_2$ é uma linguagem reconhecível.

Demonstração: Sejam $\mathcal{A}_1 = (Q_1, A, \delta_1, i_1, F_1)$ e $\mathcal{A}_2 = (Q_2, A, \delta_2, i_2, F_2)$ autómatos com transições vazias que reconhecem L_1 e L_2 (com $Q_1 \cap Q_2 = \emptyset$).
Então, o seguinte autómato $\mathcal{A} = (Q_1 \cup Q_2, A, \delta, i_1, F_2)$

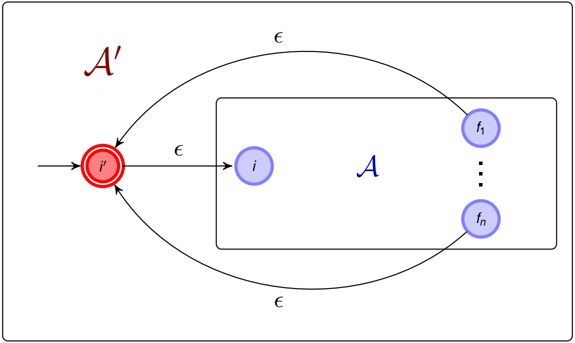


reconhece $L_1 \cdot L_2$. Portanto $L_1 \cdot L_2$ é uma linguagem reconhecível.

Lema

Se L é uma linguagem reconhecível, então L^* é uma linguagem reconhecível.

Demonstração: Seja $\mathcal{A} = (Q, A, \delta_1, i, \{f_1, \ldots, f_n\})$ um autómato que reconhece L. Então, o seguinte autómato $\mathcal{A}' = (Q \cup \{i'\}, A, \delta', i', \{i'\})$ (para $i' \notin Q$)



reconhece L^*, donde esta linguagem é reconhecível.

Combinando os lemas anteriores podemos provar uma parte do Teorema de Kleene:

Proposição

Se $L \subseteq A^*$ é uma linguagem regular, então L é reconhecível.

Demonstração: Por indução estrutural sobre $Reg(A)$.

1. Se $L = \emptyset$ ou $L = \{\epsilon\}$, então é claro que L é reconhecível.
2. Se $L = \{a\}$ em que $a \in A$, então é também claro que L é reconhecível.
3. Seja $L = L_1 \cup L_2$ (resp. $L = L_1 \cdot L_2$) e suponhamos, por hipótese de indução, que L_1 e L_2 são linguagens reconhecíveis. Então, pelo lema do slide 33 (resp. slide 35), L é reconhecível.
4. Seja $L = K^*$ e suponhamos, por H.I., que K é uma linguagem reconhecível. Então, pelo lema do slide anterior, L é reconhecível.

De (1)-(4) resulta, pelo Princípio de Indução Estrutural para $Reg(A)$, que L é uma linguagem reconhecível.

Para completar a demonstração do Teorema de Kleene, falta provar que toda a linguagem reconhecida por um autómato finito pode ser representada por uma expressão regular.

Definição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato finito. Sem perda de generalidade, consideremos que $Q = \{1, \ldots, n\}$. Associa-se a $\mathcal{A}$ um sistema de equações lineares à direita

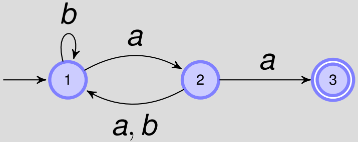
$$\begin{cases} X_1 = r_{11}X_1 + r_{12}X_2 + \cdots + r_{1n}X_n + s_1 \\ \quad\vdots \\ X_n = r_{n1}X_1 + r_{n2}X_2 + \cdots + r_{nn}X_n + s_n \end{cases}$$

onde, para cada $j, k \in Q$,

- r_{jk} é uma expressão regular que representa a linguagem
$$R_{jk} = \{a \in A : \text{existe uma transição } j \xrightarrow{a} k\};$$
- $s_j = \begin{cases} \epsilon & \text{se } j \in F \\ \emptyset & \text{se } j \notin F \end{cases}$

Exemplo

Consideremos o seguinte autómato $\mathcal{A}$:



O sistema associado a $\mathcal{A}$ é:

$$\begin{cases} X_1 = bX_1 + aX_2 + \emptyset X_3 + \emptyset \\ X_2 = (a+b)X_1 + \emptyset X_2 + aX_3 + \emptyset \\ X_3 = \emptyset X_1 + \emptyset X_2 + \emptyset X_3 + \epsilon \end{cases}$$

Lema

Seja $(t_1, t_2, \ldots, t_n) \in ER(A)^n$ a solução mínima do sistema associado ao autómato finito $\mathcal{A} = (Q, A, \delta, i, F)$.
Então, para cada estado $j \in Q$,

$$\begin{aligned}\mathcal{L}(t_j) = \; & \{u \in A^+ : \text{existe em } \mathcal{A} \text{ um caminho de etiqueta } u \\ & \qquad \text{de } j \text{ para algum estado final } \} \\ & \cup X\,,\end{aligned}$$

$$\text{onde } X = \{\epsilon\} \text{ se } j \in F \text{ e } X = \emptyset \text{ se } j \notin F.$$

Em particular,

$$L(\mathcal{A}) = \mathcal{L}(t_i),$$

ou seja, a linguagem reconhecida pelo autómato $\mathcal{A}$ é representada pela expressão regular t_i, onde i é o estado inicial de $\mathcal{A}$.

Como consequência imediata do lema anterior tem-se:

Proposição

Se L é uma linguagem reconhecível, então L é regular.

Este resultado juntamente com a proposição do slide 37, provam o resultado fundamental da teoria dos autómatos finitos:

Teorema [Kleene'1954]

Uma linguagem é regular se e só se é reconhecível.

Exemplo

Consideremos de novo o autómato $\mathcal{A}$ do exemplo anterior (slide 39).

Já vimos que este autómato tem associado o seguinte sistema de equações lineares:

$$\begin{cases} X_1 = bX_1 + aX_2 + \emptyset X_3 + \emptyset \\ X_2 = (a+b)X_1 + \emptyset X_2 + aX_3 + \emptyset \\ X_3 = \emptyset X_1 + \emptyset X_2 + \emptyset X_3 + \epsilon \end{cases}$$

que pode ser escrito simplesmente na forma

$$\begin{cases} X_1 = bX_1 + aX_2 \\ X_2 = (a+b)X_1 + aX_3 \\ X_3 = \epsilon \end{cases}$$

Atendendo ao lema anterior (slide 40), para encontrar uma expressão regular que represente a linguagem reconhecida pelo autómato $\mathcal{A}$, basta resolver este sistema e, em particular, identificar a solução para a incógnita X_1, associada ao estado inicial de $\mathcal{A}$.

Exemplo (continuação)

Pode-se deduzir sucessivamente

$$\begin{cases} X_1 = bX_1 + aX_2 \\ X_2 = (a+b)X_1 + aX_3 \\ X_3 = \epsilon \end{cases} \Leftrightarrow \begin{cases} X_1 = bX_1 + aX_2 \\ X_2 = (a+b)X_1 + a\epsilon \\ X_3 = \epsilon \end{cases}$$

$$\Leftrightarrow \begin{cases} X_1 = bX_1 + a((a+b)X_1 + a) \\ X_2 = (a+b)X_1 + a \\ X_3 = \epsilon \end{cases} \Leftrightarrow \begin{cases} X_1 = (b + a^2 + ab)X_1 + a^2 \\ X_2 = (a+b)X_1 + a \\ X_3 = \epsilon \end{cases}$$

$$\Leftrightarrow \begin{cases} X_1 = (b + a^2 + ab)^* a^2 \\ X_2 = (a+b)X_1 + a \\ X_3 = \epsilon \end{cases} \Leftrightarrow \begin{cases} X_1 = (b + a^2 + ab)^* a^2 \\ X_2 = (a+b)(b+a^2+ab)^* a^2 + a \\ X_3 = \epsilon \end{cases}$$

A solução mínima do sistema é portanto

$$((b + a^2 + ab)^* a^2, (a+b)(b+a^2+ab)^* a^2 + a, \epsilon).$$

Assim, conclui-se que $L(\mathcal{A})$ é representada pela expressão regular $(b + a^2 + ab)^* a^2$.

Recorde-se que dada uma linguagem L reconhecível, L é reconhecida por um autómato finito determinista completo acessível (DCA) $\mathcal{A} = (Q, A, \delta, i, F)$, cuja função transição pode ser vista como uma função total sobrejectiva $\delta : Q \times A \to Q$.

Os slides que se seguem descrevem conceitos que constituirão a base para o cálculo de um autómato DCA minimal para a linguagem L a partir do autómato $\mathcal{A}$.

Definição

Um autómato DCA $\mathcal{A}$ diz-se minimal quando não existem autómatos DCA equivalentes com menos estados do que $\mathcal{A}$.

Definição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA. Dois estados $q, q' \in Q$ dizem-se equivalentes, escrevendo-se $q \sim q'$, quando:

$$q \sim q' \quad \text{se e só se} \quad \forall u \in A^*,\ \delta(q, u) \in F \Leftrightarrow \delta(q', u) \in F.$$

Lema

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA. Então:

1. $\sim$ é uma relação de equivalência em Q;
2. Denotando por $\overline{q}$ a classe de equivalência de $q \in Q$ para $\sim$, e o conjunto das classes de equivalência por $\overline{Q} = Q/\!\sim\, = \{\overline{q} : q \in Q\}$, a correspondência $\overline{\delta} : \begin{array}{rcl} \overline{Q} \times A & \rightarrow & \overline{Q} \\ (\overline{q}, a) & \mapsto & \overline{\delta(q, a)} \end{array}$ é uma função.

Demonstração:

1. Exercício.
2. Para quaisquer $q_1, q_2 \in Q$ e $a \in A$,

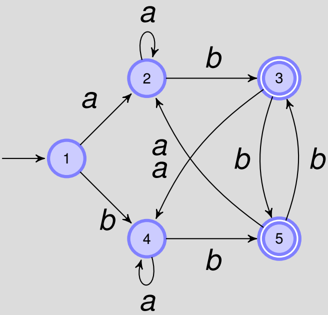
$$\begin{aligned} \overline{q_1} = \overline{q_2} & \Longrightarrow q_1 \sim q_2 \\ & \Longrightarrow \forall u \in A^*,\ \delta(q_1, au) \in F \Leftrightarrow \delta(q_2, au) \in F \\ & \Longrightarrow \forall u \in A^*,\ \delta(\delta(q_1, a), u) \in F \Leftrightarrow \delta(\delta(q_2, a), u) \in F \\ & \Longrightarrow \delta(q_1, a) \sim \delta(q_2, a) \\ & \Longrightarrow \overline{\delta(q_1, a)} = \overline{\delta(q_2, a)}. \end{aligned}$$

Definição

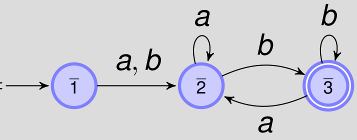
Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA. O autómato quociente de $\mathcal{A}$ por $\sim$ é o autómato $\mathcal{A}/\!\sim\, = (\overline{Q}, A, \overline{\delta}, \overline{i}, \overline{F})$ onde $\overline{F} = \{\overline{f} : f \in F\}$ é o conjunto das classes de equivalência dos elementos de F.

Exemplo

$\mathcal{A} =$



$\mathcal{A}/\!\sim\, =$



Adiante veremos que: $\overline{Q} = Q/\!\sim\, = \{\{1\}, \{2, 4\}, \{3, 5\}\} = \{\overline{1}, \overline{2}, \overline{3}\}$.

Proposição

Se $\mathcal{A} = (Q, A, \delta, i, F)$ é um autómato DCA, então o autómato quociente $\mathcal{A}/\sim$ é um autómato DCA equivalente a $\mathcal{A}$ que é minimal.

Para o cálculo de $\mathcal{A}/\sim$ é necessário resolver o seguinte problema:

Problema

Dado um autómato determinista completo acessível $\mathcal{A}$, como calcular a relação de equivalência $\sim$?

Definição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA e sejam $q, q' \in Q$. Para cada $k \in \mathbb{N}_0$, define-se uma relação binária $\sim_k$ em Q pondo, para quaisquer $q, q' \in Q$,

$$q \sim_k q' \quad \text{sse} \quad \forall u \in A^*,\ |u| \leq k \Rightarrow \Big(\delta(q, u) \in F \Leftrightarrow \delta(q', u) \in F\Big).$$

Proposição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA.

1. $\sim_k$ é uma relação de equivalência, para todo $k \in \mathbb{N}_0$;
2. $\sim_{k+1} \subseteq \sim_k$, para todo $k \in \mathbb{N}_0$;
3. $q \sim q'$ se e só se $q \sim_k q'$, para todo o $k \in \mathbb{N}_0$;
4. $q \sim_0 q'$ se e só se $q \in F \Leftrightarrow q' \in F$.

Proposição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA. As relações $\sim_k$ podem ser definidas recursivamente da seguinte forma:

1. $q \sim_0 q'$ sse $q, q' \in F$ ou $q, q' \in Q \setminus F$.
   Ou seja, $Q/\!\sim_0 = \{F, Q \setminus F\}$;
2. para cada $k \in \mathbb{N}_0$,

   $$q \sim_{k+1} q' \quad \text{sse} \quad q \sim_k q' \text{ e } \delta(q, a) \sim_k \delta(q', a) \text{ para todo o } a \in A.$$

Proposição

Seja $\mathcal{A} = (Q, A, \delta, i, F)$ um autómato DCA e seja n o número de estados de $\mathcal{A}$. Então,

1. $\sim_{r+1} = \sim_r$ para algum $r \in \{0, 1, \ldots, n-2\}$.
2. para todo $k \in \mathbb{N}_0$, $\sim_{k+1} = \sim_k$ implica $\sim_k = \sim$.

Exemplo

Consideremos o autómato $\mathcal{A}$ do slide 46, e determinemos as relações $\sim_k$ (e $\sim$) de $\mathcal{A}$, recorrendo aos dois lemas anteriores.
Dado que o conjunto de estados finais é $\{3, 5\}$ tem-se

$$Q/{\sim_0} = \{\{1, 2, 4\}, \{3, 5\}\}.$$

Notando que $\sim_1 \subseteq \sim_0$, para calcular $\sim_1$ basta verificar em cada classe-$\sim_0$ quais os elementos que são ainda equivalentes módulo $\sim_1$.
Assim,

$$\begin{aligned}
1 \not\sim_1 2 &\quad \text{pois } \delta(1, b) = 4 \not\sim_0 3 = \delta(2, b);\\
2 \sim_1 4 &\quad \text{pois } 2 \sim_0 4 \text{ e } \delta(2, a) = 2 \sim_0 4 = \delta(4, a)\\
&\quad\quad \text{e } \delta(2, b) = 3 \sim_0 5 = \delta(4, b);\\
3 \sim_1 5 &\quad \text{pois } 3 \sim_0 5 \text{ e } \delta(3, a) = 4 \sim_0 2 = \delta(5, a)\\
&\quad\quad \text{e } \delta(3, b) = 5 \sim_0 3 = \delta(5, b).
\end{aligned}$$

Tem-se portanto $\sim_1 \neq \sim_0$ e

$$Q/{\sim_1} = \{\{1\}, \{2, 4\}, \{3, 5\}\}.$$

Exemplo (continuação)

Agora, no cálculo de $\sim_2$ verifica-se que

$$\begin{array}{ll} 2\sim_2 4 & \text{pois } 2\sim_1 4 \text{ e } \delta(2,a) = 2\sim_1 4 = \delta(4,a) \\ & \quad \text{e } \delta(2,b) = 3\sim_1 5 = \delta(4,b); \\ 3\sim_2 5 & \text{pois } 3\sim_1 5 \text{ e } \delta(3,a) = 4\sim_1 2 = \delta(5,a) \\ & \quad \text{e } \delta(3,b) = 5\sim_1 3 = \delta(5,b). \end{array}$$

Tem-se então $\sim_2 = \sim_1$, donde $\sim = \sim_1$ e

$$\overline{Q} = Q/\!\sim\, = \big\{\{1\},\{2,4\},\{3,5\}\big\} = \big\{\overline{1},\overline{2},\overline{3}\big\}.$$

No slide 46, vimos já como determinar o autómato quociente $\mathcal{A}/\!\sim$ quando este conjunto quociente $\overline{Q}$ é fornecido.