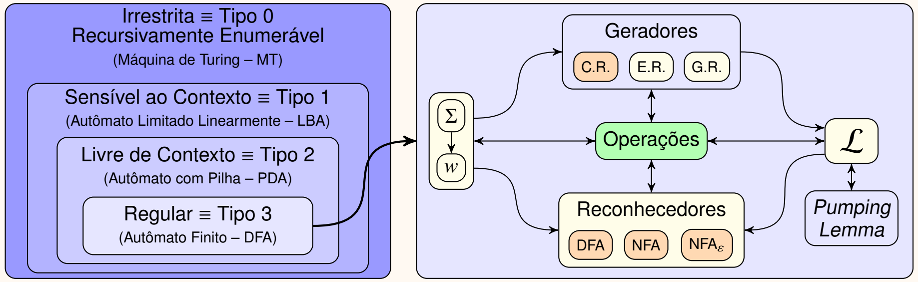
Linguagens Formais e Autômatos

Humberto Longo

Instituto de Informática
Universidade Federal de Goiás

Bacharelado em Ciência da Computação, 2021/1

Roteiro



Conjuntos regulares

- Um conjunto regular sobre um alfabeto Σ é definido como:
  - Base: $\varnothing$, $\{\varepsilon\}$ e $\{a\}$, para todo $a \in \Sigma$, são conjuntos regulares sobre Σ.
  - Recursão: se X e Y são conjuntos regulares sobre Σ, então $X \cup Y$, XY e X^* também são conjuntos regulares sobre Σ.
  - Fecho: X é um conjunto regular sobre Σ se pode ser obtido, a partir dos conjuntos regulares básicos, com a aplicação da recursão um número finito de vezes.

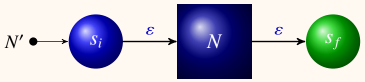
Linguagens e conjuntos regulares

- Todo autômato finito, com alfabeto Σ, aceita uma linguagem sobre Σ.
- A família de linguagens aceitas por autômatos consiste de conjuntos regulares sobre Σ:
  1. Todo conjunto regular é aceito por um NFA-ε.
  2. Toda linguagem aceita por um autômato é um conjunto regular.

NFA-ε's

Lema 1.25

- *Se $N = \langle \Sigma, S, s_0, \delta, F\rangle$ é um NFA-ε, então existe um NFA-ε equivalente $N' = \langle \Sigma, S \cup \{s_i, s_f\}, s_i, \delta', \{s_f\}\rangle$ tal que o respectivo diagrama de estados satisfaz:*
  1. *nenhum arco chega ao vértice s_i ($\delta'(s_i, \varepsilon) = s_0$),*
  2. *o único vértice final é o s_f ($\delta'(s_k, \varepsilon) = s_f$, $\forall\, s_k \in F$),*
  3. *nenhum arco sai do vértice s_f.*



Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

□

Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- Um conjunto regular sobre um alfabeto Σ é definido como:

Base: $\emptyset$, $\{\varepsilon\}$ e $\{a\}$, para todo $a \in \Sigma$, são conjuntos regulares sobre Σ.

Recursão: Se X e Y são conjuntos regulares sobre Σ, então $X \cup Y$, XY e X^* também são conjuntos regulares sobre Σ.

Fecho: X é um conjunto regular sobre Σ se pode ser obtido, a partir dos conjuntos regulares básicos, com a aplicação da recursão um número finito de vezes.

□

Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- Conjuntos regulares são construídos a partir dos conjuntos básicos usando operações de união, concatenação e fecho de Kleene.
- Transições ε podem ser usadas para construir máquinas mais complexas a partir de outras máquinas já existentes.

□

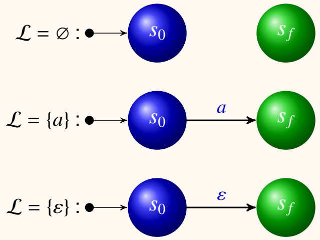
Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- Base da definição de conjunto regular:



□

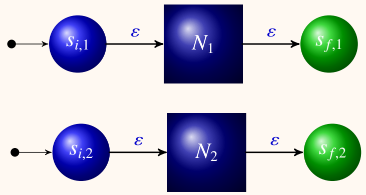
Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- Sejam $\mathcal{L}(N_1)$ e $\mathcal{L}(N_2)$ conjuntos regulares aceitos pelos NFA-ε's N_1 e N_2, respectivamente:



□

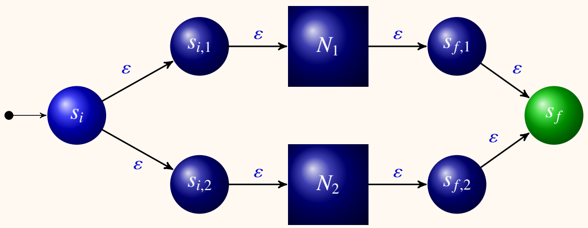
Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- $\mathcal{L}(N_1) \cup \mathcal{L}(N_2)$:



□

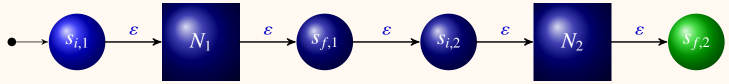
Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- $\mathcal{L}(N_1) \circ \mathcal{L}(N_2)$:



□

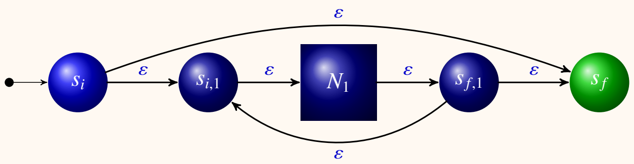
Linguagens e conjuntos regulares

Lema 1.26

- *Todo conjunto regular é aceito por um NFA-ε.*

Demonstração.

- $(\mathcal{L}(N_1))^*$:



□

Linguagens e conjuntos regulares

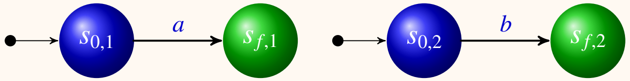
Exemplo 1.27

- A técnica usada na demonstração do Lema 1.26 pode ser usada para construir um NFA-ε que aceite $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$.

Linguagens e conjuntos regulares

Exemplo 1.27

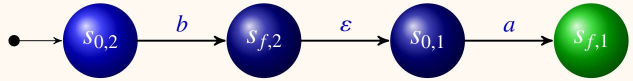
- NFA-ε que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$.
  1. Autômatos que aceitam $\{a\}$ e $\{b\}$:



Linguagens e conjuntos regulares

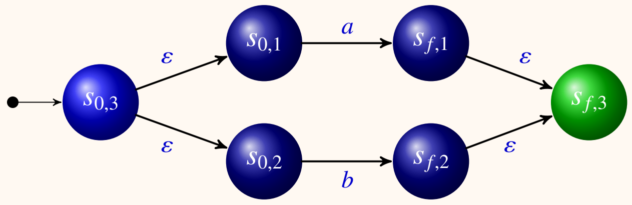
Exemplo 1.27

- NFA-ε que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$
  2. Autômato que aceita $\{b\} \circ \{a\}$:



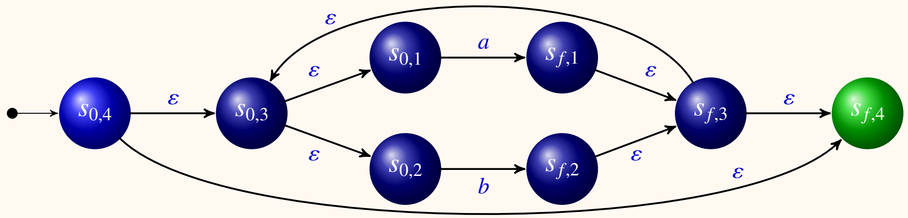
Exemplo 1.27

- NFA-ε que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$.
  3. Autômato que aceita $\{a\} \cup \{b\}$:



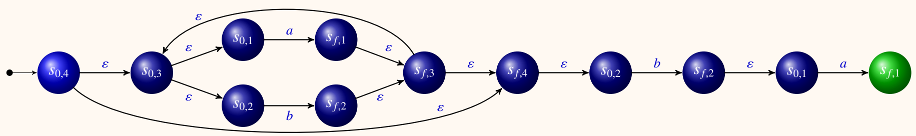
Exemplo 1.27

- NFA-ε que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$.
  4. Autômato que aceita $(\{a\} \cup \{b\})^*$:



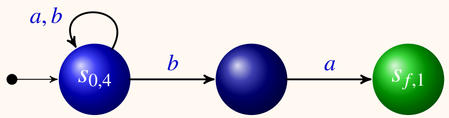
Exemplo 1.27

- NFA-ε que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$:



Exemplo 1.27

- NFA que aceita $(\{a\} \cup \{b\})^* \circ \{b\} \circ \{a\}$:



Fecho de linguagens regulares

Definição 1.28

- Uma linguagem $\mathcal{L}$, sobre um alfabeto Σ, é regular se $\mathcal{L}$ é:
  1. definida por um conjunto regular (expressão) sobre Σ;
  2. aceita por um DFA, NFA ou NFA-ε; ou
  3. gerada por uma gramática regular.

- Uma família de linguagens é fechada sobre uma operação se a aplicação da operação a algum membro da família produz um membro da própria família.
- Formulações equivalentes de linguagens regulares podem ser usadas para provar propriedades de fecho da família de linguagens regulares.

Fecho de linguagens regulares

Teorema 1.29

- *Se $\mathcal{L}_1$ e $\mathcal{L}_2$ são duas linguagens regulares, então $\mathcal{L}_1 \cup \mathcal{L}_2$, $\mathcal{L}_1 \circ \mathcal{L}_2$ e $\mathcal{L}_1^*$ são linguagens regulares.*

Demonstração.

- A definição recursiva de conjuntos regulares estabelece o fecho para essas operações. Além disso, essas linguagens são aceitas por NFA-ε (veja o teorema correspondente).

$\square$

Fecho de linguagens regulares

Teorema 1.30

- *Se $\mathcal{L}$ é uma linguagem regular, então a linguagem $\overline{\mathcal{L}}$ é regular.*

Demonstração.

- Se $\mathcal{L}$ é regular sobre o alfabeto Σ, então $\overline{\mathcal{L}} = \Sigma^* - \mathcal{L}$ também é regular. Um DFA que aceita a linguagem $\overline{\mathcal{L}}$ pode ser construído a partir de outro que aceita a linguagem $\mathcal{L}$.

$\square$

Fecho de linguagens regulares

Teorema 1.31

- *Se $\mathcal{L}_1$ e $\mathcal{L}_2$ são duas linguagens regulares, então a linguagem $\mathcal{L}_1 \cap \mathcal{L}_2$ é regular.*

Demonstração.

- Pela lei de DeMorgan $\mathcal{L}_1 \cap \mathcal{L}_2 = \overline{\overline{\mathcal{L}_1} \cup \overline{\mathcal{L}_2}}$.
- O lado direito da igualdade é regular já que é construído a partir de $\mathcal{L}_1$ e de $\mathcal{L}_2$ usando as operações de união e complementação.

$\square$

Fecho de linguagens regulares

Exemplo 1.32

- $\mathcal{L}_1 = (a \cup b)^* aa (a \cup b)^*$: cadeias que contém aa.
- $\mathcal{L}_2 = (a \cup b)^* bb (a \cup b)^*$: cadeias que contém bb.
- $\mathcal{L}$: cadeias sobre $\{a, b\}$ que contêm aa e não contêm bb.
- $\mathcal{L} = \mathcal{L}_1 \cap \overline{\mathcal{L}_2}$ é regular.

Caracterização de conjuntos regulares

Teorema 1.33 (Kleene)

- *Uma linguagem $\mathcal{L}$ é aceita por um DFA com alfabeto Σ se, e somente se, $\mathcal{L}$ é um conjunto regular sobre Σ.*

Livros texto

- R. P. Grimaldi
  Discrete and Combinatorial Mathematics – An Applied Introduction.
  Addison Wesley, 1994.
- D. J. Velleman
  How To Prove It – A Structured Approach.
  Cambridge University Press, 1996.
- J. E. Hopcroft; J. Ullman.
  Introdução Ã Teoria de Autômatos, Linguagens e Computação.
  Ed. Campus.
- T. A. Sudkamp.
  Languages and Machines – An Introduction to the Theory of Computer Science.
  Addison Wesley Longman, Inc. 1998.
- J. Carroll; D. Long.
  Theory of Finite Automata – With an Introduction to Formal Languages.
  Prentice-Hall, 1989.
- M. Sipser.
  Introduction to the Theory of Computation.
  PWS Publishing Company, 1997.
- H. R. Lewis; C. H. Papadimitriou
  Elementos de Teoria da Computação.
  Bookman, 2000.