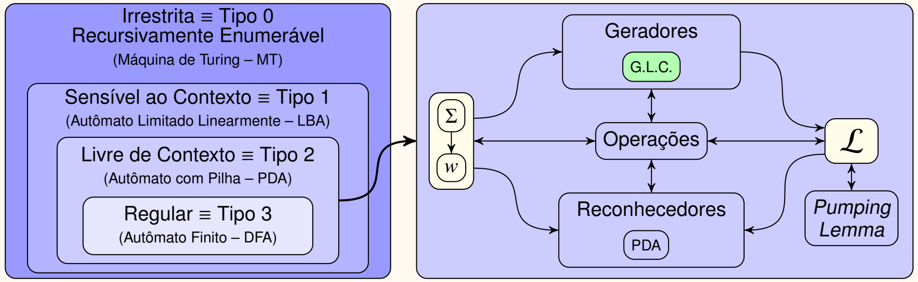
Linguagens Formais e Autômatos

Humberto Longo

Instituto de Informática
Universidade Federal de Goiás

Bacharelado em Ciência da Computação, 2021/1

Roteiro



Gramática Livre de Contexto

Definição 1.1 (GLC)

- Quádrupla $G = (V, \Sigma, P, S)$, onde:
  - V : Conjunto finito de variáveis.
    - Letras maiúsculas.
  - Σ : Conjunto finito de símbolos (terminais).
    - $a, b, c, \cdots \in \Sigma$.
    - $\ldots, U, V, W, X, Y, Z \in (V \cup \Sigma)^*$.
    - $\Sigma \cap V = \emptyset$.
  - P : Conjunto finito de regras de derivação.
    - $A \to w \equiv (A, w) \in V \times (V \cup \Sigma)^*$.
    - $A \to \varepsilon$ (derivação vazia)
  - $S \in V$: Variável inicial.

Gramática Livre de Contexto

Definição 1.2 (GLC)

- Significado da expressão “livre de contexto”???
- Para tais linguagens, cujas regras de derivações são da forma $A \to w$, a variável A deriva w sem depender (livre) de qualquer análise dos símbolos que antecedem ou seguem A (contexto).

Gramática Livre de Contexto

Revisão

- A aplicação de $A \to w$ à variável A em uAv gera a cadeia uwv.
- Se $(A \to w) \in P$, então $uAv \Rightarrow uwv$.
- $u \overset{*}{\Longrightarrow} v$ se $u = v$ ou $\exists\, u_1, u_2, \ldots, u_k,\ k \geqslant 0$ tal que $u \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \cdots \Rightarrow u_k \Rightarrow v$.

Gramática Livre de Contexto

Revisão

$\to$: Definição de regra de derivação.
- Regra de derivação pertence ao conjunto $V \times (V \cup \Sigma)^*$.

$\Rightarrow$: Aplicação de regra de derivação.
- Aplicação transforma uma cadeia em outra e pertence ao conjunto $(V \cup \Sigma)^+ \times (V \cup \Sigma)^*$.

$\overset{*}{\Longrightarrow}$: Derivação usando zero ou mais regras de derivação.

$\overset{+}{\Longrightarrow}$: Derivação usando uma ou mais regras de derivação.

$\overset{n}{\Longrightarrow}$: Derivação de comprimento n, $n \geqslant 0$:
- $v \overset{n}{\Longrightarrow} w$: w é derivado a partir de v usando n regras de derivação.

Gramática Livre de Contexto

Exemplo 1.3

- $G_1 = (V = \{A, B\}, \Sigma = \{0, 1, \#\}, P, A)$, onde: $P = \left\{\begin{matrix} A \to 0A1 \\ A \to B \\ B \to \# \end{matrix}\right\}$
- Derivação da cadeia $000\#111$:

$$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111$$

Gramática Livre de Contexto

Definição 1.4

- Seja $G = (V, \Sigma, P, S)$ uma GLC e $v \in (V \cup \Sigma)^*$. O conjunto de cadeias deriváveis a partir de v é definido como:

  Base: v é derivável a partir de v.

  Recursão: Se $u = xAy$ é derivável a partir de v ($v \overset{*}{\Longrightarrow} xAy$) e $A \to w \in P$, então a cadeia xwy é derivável a partir de v ($v \overset{*}{\Longrightarrow} xwy$).

  Fecho: As cadeias deriváveis a partir de v são exatamente as cadeias construídas a partir de v pela aplicação da recursão um número finito de vezes.

Gramática Livre de Contexto

Definição 1.5

- Seja $G = (V, \Sigma, P, S)$ uma GLC:

Forma sentencial: $w \in (V \cup \Sigma)^*$ tal que existe uma derivação $S \overset{*}{\Longrightarrow} w$ em G.
- Sequências de símbolos deriváveis a partir do símbolo inicial.

Sentença (Cadeia): $w \in \Sigma^*$ tal que existe uma derivação $S \overset{*}{\Longrightarrow} w$ em G.
- Formas sentenciais que só contém símbolos terminais.

Linguagem: $\mathcal{L}(G) = \{w \in \Sigma^* \mid S \overset{*}{\Longrightarrow} w\}$.
- Conjunto de sentenças geras pela gramática.

Gramática Livre de Contexto

Definição 1.6

- Recursão nas regras de derivação de uma GLC $G = (V, \Sigma, P, S)$:

Recursão direta: $A \rightarrow uAv$.
- Permite gerar qualquer número de cópias da subcadeia u, seguido de A e igual número de v's.
- É necessária uma regra não recursiva para parar o processo de derivação.

Recursão indireta: $A \Rightarrow w \overset{+}{\Longrightarrow} uAv$, onde A não ocorre em w.
- Variável recursiva: $A \overset{+}{\Longrightarrow} uAv$.

Gramática Livre de Contexto

Exemplo 1.7

- Gramática G que gera a linguagem composta de cadeias com número positivo par de a's:

$$G = (V, \Sigma, P, S)$$

$$V = \{S, A\}$$

$$\Sigma = \{a, b\}$$

$$P = \left\{\begin{matrix} S \rightarrow AA\,, \\ A \rightarrow AAA \mid bA \mid Ab \mid a \end{matrix}\right\}$$

Gramática Livre de Contexto

Exemplo 1.7

$S \Rightarrow AA$	$S \Rightarrow AA$	$S \Rightarrow AA$	$S \Rightarrow AA$
$\Rightarrow \underline{a}A$	$\Rightarrow \underline{AAA}A$	$\Rightarrow A\underline{a}$	$\Rightarrow \underline{a}A$
$\Rightarrow a\underline{AAA}$	$\Rightarrow \underline{a}AAA$	$\Rightarrow \underline{AAA}a$	$\Rightarrow a\underline{AAA}$
$\Rightarrow a\underline{bA}AA$	$\Rightarrow a\underline{bA}AA$	$\Rightarrow AA\underline{bA}a$	$\Rightarrow aAA\underline{a}$
$\Rightarrow ab\underline{a}AA$	$\Rightarrow ab\underline{a}AA$	$\Rightarrow AAb\underline{a}a$	$\Rightarrow a\underline{bA}Aa$
$\Rightarrow aba\underline{bA}A$	$\Rightarrow aba\underline{bA}A$	$\Rightarrow A\underline{bA}baa$	$\Rightarrow ab\underline{Ab}Aa$
$\Rightarrow abab\underline{a}A$	$\Rightarrow abab\underline{a}A$	$\Rightarrow Ab\underline{a}baa$	$\Rightarrow abAb\underline{a}a$
$\Rightarrow ababa\underline{a}$	$\Rightarrow ababa\underline{a}$	$\Rightarrow \underline{a}babaa$	$\Rightarrow ab\underline{a}baa$
(esq.)	(esq.)	(dir.)	

$$P = \left\{\begin{matrix} S \rightarrow AA\,, \\ A \rightarrow AAA \mid bA \mid Ab \mid a \end{matrix}\right\}$$

Gramática Livre de Contexto

Exercícios

- Construir gramática G tal que:
  1. $\mathcal{L} = \{a^n b^m \mid n \geqslant 0 \text{ e } m \geqslant 1\}$.
  2. $\mathcal{L} = \{a^i b^j c^i \mid i \geqslant 0 \text{ e } j \geqslant 1\}$.
  3. $\mathcal{L} = \{w \mid w \in \{0,1\}^+$ e não tem 1's consecutivos$\}$.
  4. $\mathcal{L} = \{w \mid w \in \{0,1,2\}^+$ e todos os 0's sejam consecutivos$\}$.

Gramática Livre de Contexto

Lema 1.8

- *Seja G uma GLC e $v \overset{n}{\Longrightarrow} w$ uma derivação em G, onde*

$$v = w_1 A_1 w_2 A_2 w_3 \ldots w_k A_k w_{k+1},$$

com $w_i \in \Sigma^*$. Então existem cadeias $p_i \in (\Sigma \cup V)^*$ tais que:

1. $A_i \overset{t_i}{\Longrightarrow} p_i$;
2. $w = w_1 p_1 w_2 p_2 w_3 \ldots w_k p_k w_{k+1}$; e
3. $\sum_{i=1}^{k} t_i = n$.

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Base: Consiste de derivações da forma $v \overset{0}{\Longrightarrow} w$. Neste caso, $w = v$ e cada A_i é igual ao correspondente p_i. As derivações têm a forma $A_i \overset{0}{\Longrightarrow} p_i$.

□

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Base: Consiste de derivações da forma $v \overset{0}{\Longrightarrow} w$. Neste caso, $w = v$ e cada A_i é igual ao correspondente p_i. As derivações têm a forma $A_i \overset{0}{\Longrightarrow} p_i$.

  Hipótese: Suponha que todas as derivações da forma $v \overset{n}{\Longrightarrow} w$ podem ser decompostas em derivações a partir de A_i, as variáveis de v, as quais juntas formam uma derivação de w a partir de v de comprimento n.

□

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Passo: Seja $v \overset{n+1}{\Longrightarrow} w$ uma derivação em G com

  $$v = w_1 A_1 w_2 A_2 \ldots w_k A_k w_{k+1},$$

  com $w_i \in \Sigma^*$.

  Essa derivação pode ser escrita como $v \overset{1}{\Longrightarrow} u \overset{n}{\Longrightarrow} w$. Isto reduz a derivação original a uma derivação de comprimento n, a qual está na forma correta (pela aplicação de uma derivação simples e por hipótese de indução).

□

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Passo: A derivação $v \overset{1}{\Longrightarrow} u$ transforma uma das variáveis de v (A_j) com a regra

  $$A_j \rightarrow u_1 B_1 u_2 B_2 u_3 \ldots u_m B_m u_{m+1},$$

  onde cada $u_i \in \Sigma^*$.

  A cadeia u é obtida a partir de v pela substituição de A_j pelo lado direito de sua regra de derivação. Neste caso u é escrito como:

  $$u = w_1 A_1 \ldots A_{j-1} w_j u_1 B_1 u_2 B_2 \ldots u_m B_m u_{m+1} w_{j+1} A_{j+1} \ldots w_k A_k w_{k+1}.$$

□

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Passo: Como w é derivável a partir de u usando n aplicações de regras de derivação. A hipótese indutiva garante que existem cadeias $p_1, \ldots, p_{j-1}, q_1, \ldots, q_m$ e $p_{j+1}, \ldots, p_k$ que satisfazem:

  1. $A_i \overset{t_i}{\Longrightarrow} p_i$ para $i = 1, \ldots, j-1, j+1, \ldots, k$;
     $B_i \overset{s_i}{\Longrightarrow} q_i$ para $i = 1, \ldots, m$;
  2. $w = w_1 p_1 w_2 \ldots p_{j-1} w_j u_1 q_1 u_2 \ldots u_m q_m u_{m+1} w_{j+1} p_{j+1} \ldots w_k p_k w_{k+1}$;
  3. $\sum_{i=1}^{j-1} t_i + \sum_{i=j+1}^{k} t_i + \sum_{i=1}^{m} s_i = n$.

□

Gramática Livre de Contexto

Demonstração.

- Indução no comprimento da derivação de w a partir de v:

  Passo: Combinando-se a regra $A_j \rightarrow u_1 B_1 u_2 B_2 \ldots u_m B_m u_{m+1}$ com as derivações $B_i \overset{*}{\Longrightarrow} q_i$, obtem-se a derivação

  $$A_j \overset{*}{\Longrightarrow} u_1 q_1 u_2 q_2 \ldots u_m q_m u_{m+1} = p_j$$

  cujo comprimento é a soma dos comprimentos das derivações a partir dos B_i's mais 1.

  As derivações $A_i \overset{*}{\Longrightarrow} p_i$, $i = 1, \ldots, k$, fornecem a decomposição desejada da derivação de w a partir de v.

□

Gramática Livre de Contexto

- Interpretação do Lema 1.8:
  - Flexibilidade e modularidade das derivações em GLC's.
  - Qualquer derivação complexa pode ser quebrada em subderivações das variáveis que a constituem.
- Modularidade explorada no projeto de linguagens complexas:
  - Uso de variáveis para definir subconjuntos menores da linguagem.
  - Combinação das sublinguagens através de regras adicionais na gramática.

Gramática Livre de Contexto

Exemplo 1.9

- $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, B\} \\ \Sigma = \{a, b\} \\ P = \begin{Bmatrix} S \to aSa \mid aBa \\ B \to bB \mid b \end{Bmatrix} \end{cases}$
- $S \to aSa$: gera igual número de a's em cada lado da cadeia.
- $S \to aBa$: elimina recursão e garante presença de um par de a's na cadeia.
- $B \to bB$: gera qualquer número de b's.
- $B \to b$: remove a variável B.
- $\mathcal{L}(G) = \{a^n b^m a^n \mid n > 0,\ m > 0\}$.

Gramática Livre de Contexto

Exemplo 1.10

- $\mathcal{L}(G) = \{a^n b^m c^m d^{2n} \mid n \geqslant 0,\ m > 0\}$.
- $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, A\} \\ \Sigma = \{a, b, c, d\} \\ P = \begin{Bmatrix} S \to aSdd \mid A, \\ A \to bAc \mid bc \end{Bmatrix} \end{cases}$
- Derivações da variável S gera a's e d's.
- Derivações da variável A gera b's e c's.
- $A \to bc$: elimina recursão e garante presença de bc na cadeia.

Gramática Livre de Contexto

Exemplo 1.11

- $\mathcal{L}(G) = \{w \in \{a, b\}^* \mid w = w^R\}$.
- $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, A\} \\ \Sigma = \{a, b\} \\ P = \begin{Bmatrix} S \to a \mid b \mid \varepsilon, \\ S \to aSa \mid bSb \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.12

- $\mathcal{L}(G) = \{a^n b^m \mid 0 \leqslant n \leqslant m \leqslant 2n\}$.
- $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S\} \\ \Sigma = \{a, b\} \\ P = \{S \to aSb \mid aSbb \mid \varepsilon\} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.13

- Linguagem gerada pela expressão regular a^+b^*:
  - $G_1 = (V_1, \Sigma, P_1, S)$, onde: $\begin{cases} V_1 = \{S, A, B\} \\ \Sigma = \{a, b\} \\ P_1 = \begin{Bmatrix} S \to AB, \\ A \to aA \mid a, \\ B \to bB \mid \varepsilon \end{Bmatrix} \end{cases}$
  - $G_2 = (V_2, \Sigma, P_2, S)$, onde: $\begin{cases} V_2 = \{S, B\} \\ \Sigma = \{a, b\} \\ P_2 = \begin{Bmatrix} S \to aS \mid aB, \\ B \to bB \mid \varepsilon \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.14

- Linguagem gerada pela expressão regular $a^*ba^*ba^*$ (cadeias que contêm dois b's):
  - $G_1 = (V_1, \Sigma, P_1, S)$, onde: $\begin{cases} V_1 = \{S, A, B\} \\ \Sigma = \{a, b\} \\ P_1 = \begin{Bmatrix} S \to aS \mid bA, \\ A \to aA \mid bB, \\ B \to aB \mid \varepsilon \end{Bmatrix} \end{cases}$
  - $G_2 = (V_2, \Sigma, P_2, S)$, onde: $\begin{cases} V_2 = \{S, A\} \\ \Sigma = \{a, b\} \\ P_2 = \begin{Bmatrix} S \to AbAbA, \\ A \to aA \mid \varepsilon \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.15

- Linguagem cujas cadeias contêm pelo menos dois b's:
  - $a^*ba^*b(a \cup b)^*$.
  - $G_1 = (V_1, \Sigma, P_1, S)$, onde: $\begin{cases} V_1 = \{S, A, B\} \\ \Sigma = \{a, b\} \\ P_1 = \begin{Bmatrix} S \to aS \mid bA \\ A \to aA \mid bB \\ B \to aB \mid bB \mid \varepsilon \end{Bmatrix} \end{cases}$
  - $G_2 = (V_2, \Sigma, P_2, S)$, onde: $\begin{cases} V_2 = \{S, A\} \\ \Sigma = \{a, b\} \\ P_2 = \begin{Bmatrix} S \to AbAbA \\ A \to aA \mid bA \mid \varepsilon \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.16

- Linguagem gerada pela expressão regular $a^*b^+a^*b^+a^*$:
  - $G_1 = (V_1, \Sigma, P_1, S)$, onde: $\begin{cases} V_1 = \{S, A, B, C, D\} \\ \Sigma = \{a, b\} \\ P_1 = \begin{Bmatrix} S \to aS \mid bA\,, \\ A \to bA \mid aB \mid B\,, \\ B \to aB \mid bC\,, \\ C \to bC \mid D\,, \\ D \to aD \mid \varepsilon \end{Bmatrix} \end{cases}$
  - $G_2 = (V_2, \Sigma, P_2, S)$, onde: $\begin{cases} V_2 = \{S, A, B\} \\ \Sigma = \{a, b\} \\ P_2 = \begin{Bmatrix} S \to ABABA\,, \\ A \to aA \mid \varepsilon\,, \\ B \to bB \mid b \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.17

- Linguagem cujas cadeias são de comprimento par (alfabeto $\{a, b\}$):
  - $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, O\} \\ \Sigma = \{a, b\} \\ P = \begin{Bmatrix} S \to aO \mid bO \mid \varepsilon\,, \\ O \to aS \mid bS \end{Bmatrix} \end{cases}$
  - S : formas sentenciais com número par de terminais.
  - O : formas sentenciais com número ímpar de terminais.
  - **Exercício:** Linguagens, sobre o alfabeto $\{a, b\}$, cujas cadeias são de comprimento múltiplo de $3, 4, 5, \ldots$.

Gramática Livre de Contexto

Exemplo 1.18

- Linguagem, sobre $\Sigma = \{a, b\}$, cujas cadeias contém número par de b's:
  - $G_1 = (V_1, \Sigma, P_1, S)$, onde: $\begin{cases} V_1 = \{S, A, B\} \\ \Sigma = \{a, b\} \\ P_1 = \begin{Bmatrix} S \to aS \mid bA \mid \varepsilon\,, \\ A \to aA \mid bS \mid bB\,, \\ B \to aB \mid \varepsilon \end{Bmatrix} \end{cases}$
  - $G_2 = (V_2, \Sigma, P_2, S)$, onde: $\begin{cases} V_2 = \{S, A\} \\ \Sigma = \{a, b\} \\ P_2 = \begin{Bmatrix} S \to aS \mid bA \mid \varepsilon\,, \\ A \to aA \mid bS \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.19

- Linguagem, sobre $\Sigma = \{a, b\}$, cujas cadeias contêm número par de a's e de b's:

Variável	Significado
S	Nr. par de a's e nr. par de b's
A	Nr. par de a's e nr. ímpar de b's
B	Nr. ímpar de a's e nr. par de b's
C	Nr. ímpar de a's e nr. ímpar de b's

  - $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, A, B, C\} \\ \Sigma = \{a, b\} \\ P = \begin{Bmatrix} S \to aB \mid bA \mid \varepsilon\,, \\ A \to aC \mid bS\,, \\ B \to aS \mid bC\,, \\ C \to aA \mid bB \end{Bmatrix} \end{cases}$

Gramática Livre de Contexto

Exemplo 1.20

- Linguagem, sobre $\{a, b, c\}$, cujas cadeias não contêm abc:
  - $G = (V, \Sigma, P, S)$, onde: $\begin{cases} V = \{S, A, B\} \\ \Sigma = \{a, b, c\} \\ P = \begin{Bmatrix} S \to aA \mid bS \mid cS \mid \varepsilon, \\ A \to aA \mid bB \mid cS \mid \varepsilon, \\ B \to aA \mid bS \mid \varepsilon \end{Bmatrix} \end{cases}$
- Máximo de uma variável em cada forma sentencial:
  - A: ocorre quando o terminal prévio é um a;
  - B: ocorre somente precedida por ab (não pode gerar um c).

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.