REMOVENDO AMBIGUIDADE

- REMOVER AMBIGUIDADE PODE NÃO SER FÁCIL

- NÃO PODE EXISTIR ALGORITMO PARA DECIDIR SE UMA DADA LINGUAGEM É OU NÃO AMBÍGUA

- ALÉM DISSO HÁ LINGUAGENS PARA AS QUAIS TODA GRAMÁTICA É AMBÍGUA.

  → CHAMAMOS TAIS LINGUAGENS DE INERENTEMENTE AMBÍGUA

EX: $L_{IN} = \{a^i b^j c^k : i, j, k \geq 0 \text{ E } i = j \text{ OU } j = k\}$

LINGUAGENS REGULARES

EX:
$$S \to 0A \mid 0$$
$$A \to 0 \mid \varepsilon$$

0: $S \Rightarrow 0$

OU $S \Rightarrow 0A \Rightarrow 0$

- SEMPRE É POSSÍVEL OBTER UMA GRAMÁTICA NÃO AMBÍGUA PARA UMA LINGUAGEM REGULAR

Alg: DADO GRAMÁTICA G PARA LINGUAGEM REG., DEVOLVE GRAMÁTICA G' NÃO AMBÍGUA.

1) DETERMINO AFND M T.Q. $L(M) = L(G)$
2) DETERMINA AFD M' T.Q. $L(M') = L(M)$
3) DETERMINA GRAMÁTICA G', OBTIDO DE M'.

Bombeamento LCC

- Como provar que uma linguagem não é LLC?

Ex: $L_{abc} = \{a^n b^n c^n : n \geq 0\}$ não é LLC

- Não é suficiente não ser capaz de construir uma GCC que gere L_{abc}

- A estratégia é a mesma usada para provar que uma linguagem não é regular.

  → Vamos provar que toda LLC possui uma propriedade que podemos testar.

<u>Def</u>: Dizemos que uma árvore é m-ária se cada vértice tem no máximo m filhos.

- A altura de uma árvore é o comprimento do caminho mais longo entre a raiz e uma folha

<u>Def</u>: $f(h)$ = # folhas de uma árvore m-ária completa com altura h

É fácil ver que temos $f(h) = m f(h-1)$

Logo $f(h) = m^h$

<u>Def</u>: A amplitude $\alpha(G)$ de uma GLC é o comprimento máximo das palavras que aparecem à direita de uma regra

Ex: $X \to 0X1 \mid 0 \mid 1 \mid \varepsilon$ tem amplitude 3

OBS: SE UMA GLC TEM AMPLITUDE α, TODAS AS SUAS ÁRVORES SÃO M-ÁRIAS.

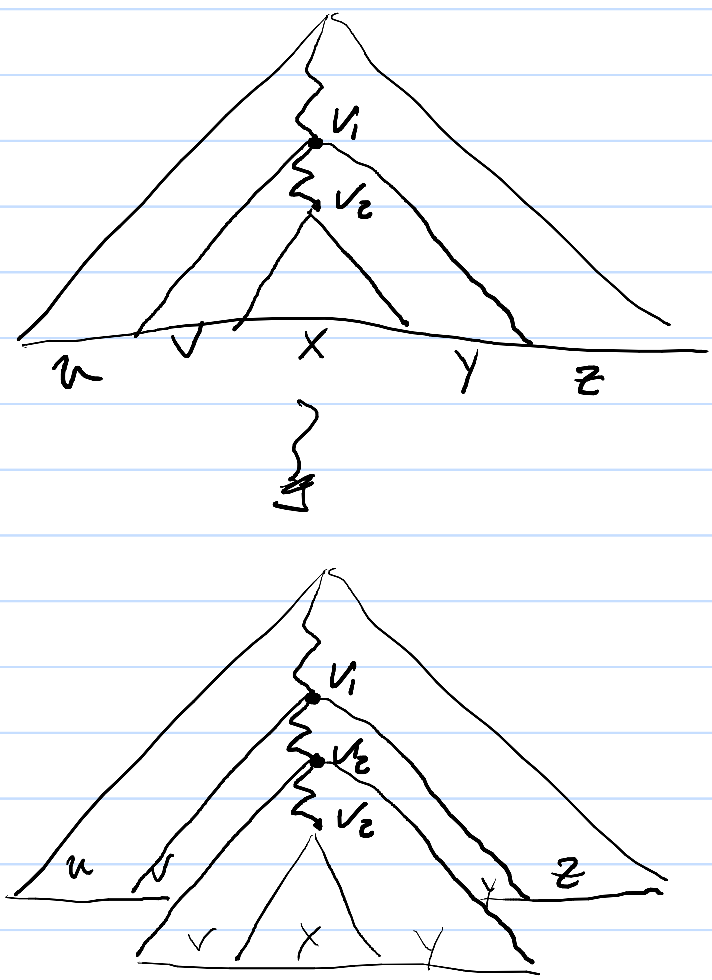
LEMA: SEJA G UMA GLC. SE X É UMA VARIÁVEL DE G E W É COLHEITA DE UMA X-ÁRVORE DE ALTURA h, ENTÃO

$$|w| \leq \alpha(G)^h$$

$$|w| > \alpha(G)^{|V|}$$

PELO LEMA, SE A COLHEITA DE UMA ÁRVORE É MUITO GRANDE, ENTÃO A ALTURA DA ÁRVORE TAMBÉM SERÁ.
E, SE ESSA ALTURA FOR MAIOR QUE O NÚMERO DE VARIÁVEIS DA GRAMÁTICA, UM CAMINHO MÁXIMO CONTERÁ DOIS VÉRTICES v_1 E v_2 COM MESMA VARIÁVEL A

AS REGRAS ASSOCIADAS A v_1 E v_2 TÊM A À ESQ.



LEMA DO BOMBEAMENTO: SEJA G UMA GLC. EXISTE INTEIRO p QUE DEPENDE DE G, T.Q., SE $w \in L(G)$ E $|w| \geq p$, ENTÃO EXISTE DECOMP. DE w COMO $w = uvxyz$, ONDE

1) $vy \neq \varepsilon$
2) $|vxy| \leq p$
3) $uv^n xy^n z \in L(G)$, PARA TODO $n \geq 0$

PROVA: INDUÇÃO EM $\alpha(G)$.

- TOME $p = \alpha(G)^{K+1}$, ONDE K É O NÚMERO DE VARIÁVEIS DE G.
- SEJA w T.Q. $|w| \geq p$, E SEJA T UMA ÁRVORE DE DERIVAÇÃO P/ w.
- T TEM ALTURA PELO MENOS K+1
- HÁ CAMINHO DA RAIZ ATÉ UMA FOLHA QUE REPETE VARIÁVEL, DIGAMOS A
- SEJAM V_1 E V_2 OS DOIS ÚLTIMOS VÉRTICES QUE POSSUEM A
  ⇝ SEJA τ_i COM RAIZ EM V_i
  ⇝ SEJA X_i A COLHEITA DE τ_i
  X_2 É SUBPALAVRA DE X_1
  ⇝ $X_1 = vX_2y$
  ⇝ COMO τ_2 TEM ALTURA MENOR QUE K, TEMOS $|vX_2y| \leq p$
  ⇝ SE $vy = \varepsilon$, τ_1 SERIA UM CAMINHO.

- COMO USAR:
  1) SUPÕE QUE A LINGUAGEM É LC
  2) ESCOLHE PALAVRA MAIOR QUE p
  3) MOSTRAR QUE NÃO É POSSÍVEL BOMBEAR

EXEMPLO: $L_{abc} = \{a^m b^m c^m : m \geq 0\}$

TOME $w = a^m b^m c^m$ COM $m \geq p$

SUPONHA $w = uvxyz$ COM $|vxy| \leq p$

SE $vxy \subseteq a^m$, b^m, OU c^m, ENTÃO $uv^2xy^2z \notin L_{abc}$

ENTÃO PODEMOS SUPOR $vxy \subseteq a^i b^j$ COM $i + j = p$

MAS NESSE CASO

$$uv^2xy^2z = a^{m+i} b^{m+j} c^m \notin L_{abc}$$

EXEMPLO: $L_{PRIMOS} = \{0^p : p$ É PRIMO POSITIVO$\}$

EXISTE p TAL QUE O BOMBEAMENTO TEM QUE FUNCIONAR

SEJA m UM PRIMO MAIOR QUE p

$w = 0^m \in L_{PRIMOS}$

ESCREVA $w = uvxyz$, COM $|vxy| \leq p$, $vy \neq \varepsilon$

CLARAMENTE $v = 0^i$, $y = 0^j$

PELO LEMA $uv^n xy^n z \in L_{PRIMOS}$. $\forall n$

TOME $n = m + 1$

$$uv^{m+1}xy^{m+1}z = 0^{m + mi + mj} = 0^{m(1+i+j)}$$

COMO $m(1+i+j)$ NÃO É PRIMO L_{PRIMOS}

NÃO É LLC