

## Lema do bombeamento para gramáticas

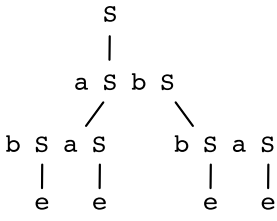
Na prática, os livros desenvolvem esse raciocínio com as gramáticas l.c., e é isso o que nós vamos ver em seguida.

Considere a seguinte gramática para a linguagem **BAL**

$$S \rightarrow aSbS \mid bSa \mid e$$

Como foi observado na aula passada, o fato de algumas regras possuírem duas variáveis no lado direito faz com que seja mais conveniente raciocinar sobre derivações na forma de árvore.

Por exemplo, a árvore de derivação abaixo



produz a palavra **ababba** — (*you* *consegue* *ver*?)

Agora, as coisas ficam mais interessantes quando nós observamos que uma árvore de derivação possui *subárvores* no seu interior

— Figura —

Quer dizer, qualquer variável no interior de uma árvore de derivação possui uma subárvore embaixo dela (que corresponde às regras que são aplicadas a partir daquela variável).

E nós sempre podemos trocar uma subárvore por outra, desde que as suas raízes sejam as mesmas.

Por exemplo, na passagem abaixo

— Figura —

nós trocamos a subárvore da esquerda por uma cópia da árvore original.

(*You* *consegue* *ver*?)

E daí, nós podemos trocar a subárvore mais à esquerda outra vez por uma cópia da primeira árvore

— Figura —

*You* *notou* *que* *já* *tem* *uma* *forma* *de* *bombeamento* *acontecendo* *aqui*?

Agora, só nos resta colocar as coisas no formato geral de um lema do bombeamento.

## Lema do bombeamento para linguagens LC

Suponha que  $L$  é uma linguagem LC infinita.

E suponha que  $G$  é uma gramática LC que produz a linguagem  $L$ .

Ora,  $L$  é infinita mas  $G$  não é.

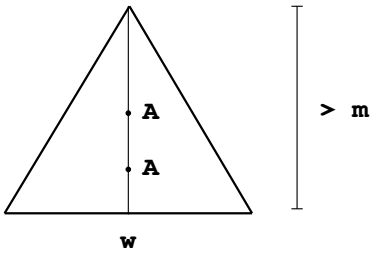
Quer dizer,  $G$  possui um número finito de variáveis (e de regras).

Isso significa que para produzir uma palavra muito grande de  $L$ , as variáveis de  $G$  vão ter que aparecer mais de uma vez durante a derivação — i.e., ao menos uma delas.

Mas nós podemos raciocinar em termos de *árvores de derivação*.

E a observação que nós podemos fazer é a seguinte

- em uma árvore de derivação muito grande, existe um caminho onde alguma variável  $A$  aparece mais de uma vez



A intuição é que isso corresponde a um laço na derivação



e que nós podemos utilizar esse laço para fazer o bombeamento na derivação.

Certo.

Mas, vamos recapitular o que nós temos até aqui.

A ideia é que

- se  $p$  é uma palavra muito grande de  $L$ , então  $p$  deve ser produzida por uma árvore de derivação com profundidade muito grande
  - mas, uma árvore de derivação com profundidade muito grande tem um caminho que forma um laço que pode ser bombeado
- Figura —

Agora, nós precisamos entender qual é o efeito do bombeamento sobre a palavra  $p$ .

O primeiro passo é voltar para a representação da árvore com as duas ocorrências da variável  $A$  no caminho

— Figura —

Daí, nós decompomos a árvore de derivação primeiro assim

— Figura —

Quer dizer,  $u$  é a porção da palavra  $p$  que é produzida por tudo que fica à esquerda do primeiro  $A$  na árvore.

E  $z$  é a porção da palavra  $p$  que é produzida por tudo que fica à direita do primeiro  $A$  na árvore.

A parte  $p'$  corresponde à porção da palavra que é produzida por essa variável  $A$ .

A seguir, nós fazemos outra decomposição

— Figura —

E agora nós estamos prontos para realizar o bombeamento.

Quer dizer, a ideia é trocar a subárvore do segundo  $A$  por uma cópia da subárvore do primeiro  $A$

— Figura —

E agora, nós podemos ver qual é o resultado do bombeamento:

- os fragmentos  $u$  e  $y$  foram duplicados

A operação inversa corresponde a trocar a subárvore do primeiro  $A$  por uma cópia da subárvore do segundo  $A$

— Figura —

E o resultado dessa operação é

- os fragmentos  $u$  e  $y$  foram removidos

Pronto!

Agora nós estamos prontos para enunciar o lema do bombeamento para linguagens LC

⇒ Se  $p$  é uma palavra muito grande de uma linguagem LC  $L$ , então  $p$  pode ser decomposta em 5 partes

$$p = uvxyz$$

de modo que as palavras

$$p' = uxz \quad \text{e} \quad p' = uvvxyyz$$

também pertencem à linguagem  $L$

De fato, como nós podemos aplicar o bombeamento mais de uma vez, qualquer palavra da forma

$$uv^kxy^kz$$

pertence à linguagem  $L$ .