Linguagens Formais e Autômatos

Gramáticas Regulares

Felipe Cunha

Material extraído do livro e slides do Prof. Newton Vieira (http://dcc.ufmg.br/~nvieira) e do Prof. Andrei Rimsa Álvares

Gramáticas Regulares

- Até agora foram vistas três formas de se especificar uma linguagem regular
  - Usando notação de conjuntos
    - Aplicação: desenvolvimento da teoria
  - Desenhando um diagrama de estados em forma de grafo
    - Aplicação: processo de concepção de um reconhecedor
  - Usando expressões regulares
    - Aplicação: para manipulações formais e para referência compacta a conjunto de palavras
- Uma nova forma é através de gramáticas regulares, mediante a um conjunto de regras que a gera
  - Aplicação: mérito teórico de prover um lugar para as linguagens regulares na Hierarquia de Chomsky

Gramáticas Regulares

- Como linguagens regulares podem ser especificadas?
  - Automâtos finitos via um reconhecedor para ela
  - Expressões regulares via uma expressão que a denota
  - Gramáticas regulares via um gerador para ela

- Uma gramática regular permite mostrar como gerar todas, e apenas, as palavras de uma linguagem

Gramáticas Regulares

- Uma gramática regular (GR) é uma gramática (V, Σ, R, P), em que cada regra tem uma das formas:
  - $X \rightarrow a$
  - $X \rightarrow aY$
  - $X \rightarrow \lambda$

  - , onde X, Y $\in$ V e a $\in$ Σ

- Formato das formas sentenciais wA, onde w $\in$ Σ+ e A $\in$ V

Exemplo

- Seja $L = \{w \in \{a,b,c\}^* \mid w$ não contém $abc\}$, uma GR que gera L seria $(\{A,B,C\}, \{a,b,c\}, R, A)$, onde R contém as regras
  - $A \to aB \mid bA \mid cA \mid \lambda$
  - $B \to aB \mid bC \mid cA \mid \lambda$
  - $C \to aB \mid bA \mid \lambda$

Linguagem regular

- Toda **gramática regular** gera uma **linguagem regular**

Seja uma GR $G = (V, \Sigma, R, P)$.

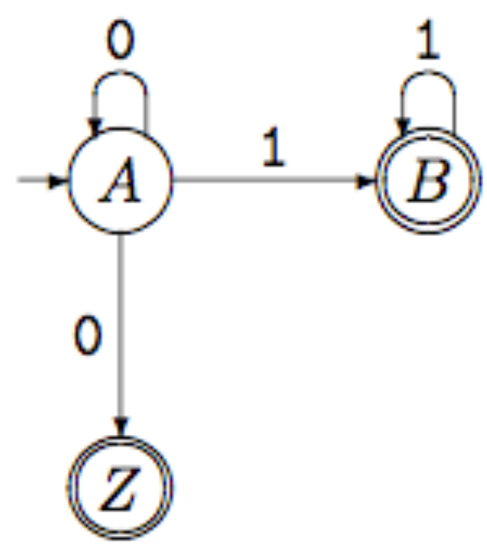
Constrói-se um AFN $M = (E, \Sigma, \delta, \{P\}, F)$ tal que $L(M) = L(G)$. Seja algum $Z \notin V$.

- $E = \begin{cases} V \cup \{Z\} & \text{se } R \text{ contém regra da forma } X \rightarrow a \\ V & \text{caso contrário.} \end{cases}$
- Para toda regra da forma:
  - $X \rightarrow aY$ faça $Y \in \delta(X, a)$,
  - $X \rightarrow a$ faça $Z \in \delta(X, a)$.
- $F = \begin{cases} \{X | X \rightarrow \lambda \in R\} \cup \{Z\} & \text{se } Z \in E \\ \{X | X \rightarrow \lambda \in R\} & \text{caso contrário.} \end{cases}$

Exemplo

- Seja $L(G) = 0^*(0 + 1^+)$. A gramática regular que a reconhece é GR $G = (\{A,B\}, \{0,1\}, R, A)$, onde R é dado por
  - $A \rightarrow 0A \mid 1B \mid 0$
  - $B \rightarrow 1B \mid \lambda$

Regras	Transições	Observações
$A \rightarrow 0A$	$A \in \delta(A, 0)$	
$A \rightarrow 1B$	$B \in \delta(A, 1)$	
$A \rightarrow 0$	$Z \in \delta(A, 0)$	Z é estado final
$B \rightarrow 1B$	$B \in \delta(B, 1)$	
$B \rightarrow \lambda$		B é estado final



Gramática Regular

- Toda **linguagem regular** é gerada por **gramática regular**

- Seja um AFN M = (E, Σ, δ, {i}, F). Uma GR que gera L(M) seria G = (E, Σ, R, i), onde

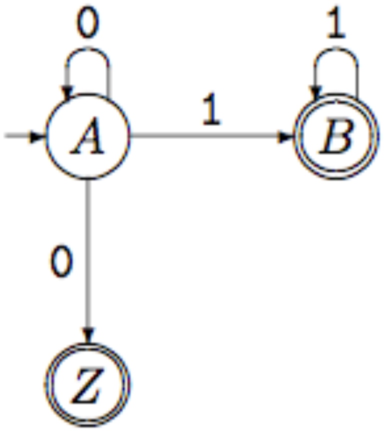
$$R = \{e \rightarrow ae' \mid e' \in \delta(e,a)\} \cup \{e \rightarrow \lambda \mid e \in F\}$$

- Deve-se construir gramática regular para L(M) = L(G) de forma que:

$$i \Rightarrow w^{*} \text{ se, e somente se, } \hat{\delta}(\{i\},w\} \cap F \neq \varnothing$$

Exemplo

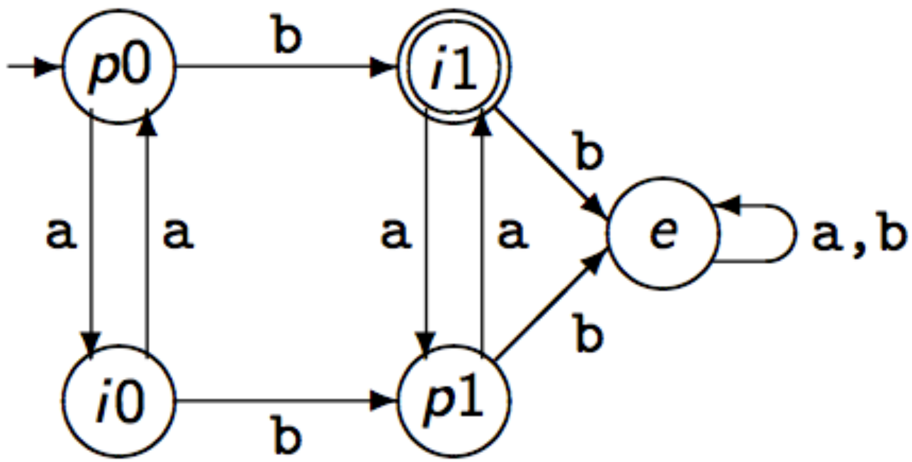
- Seja o AFN dado por:



  - A GR gerada para essa automâto é dada por $G = (\{A,B,Z\}, \{0,1\}, R, A)$, onde R é dado por:
    - $A \rightarrow 0A \mid 0Z \mid 1B$
    - $B \rightarrow 1B \mid \lambda$
    - $Z \rightarrow \lambda$

Exercícios

- Obtenha a GR correspondente ao AF

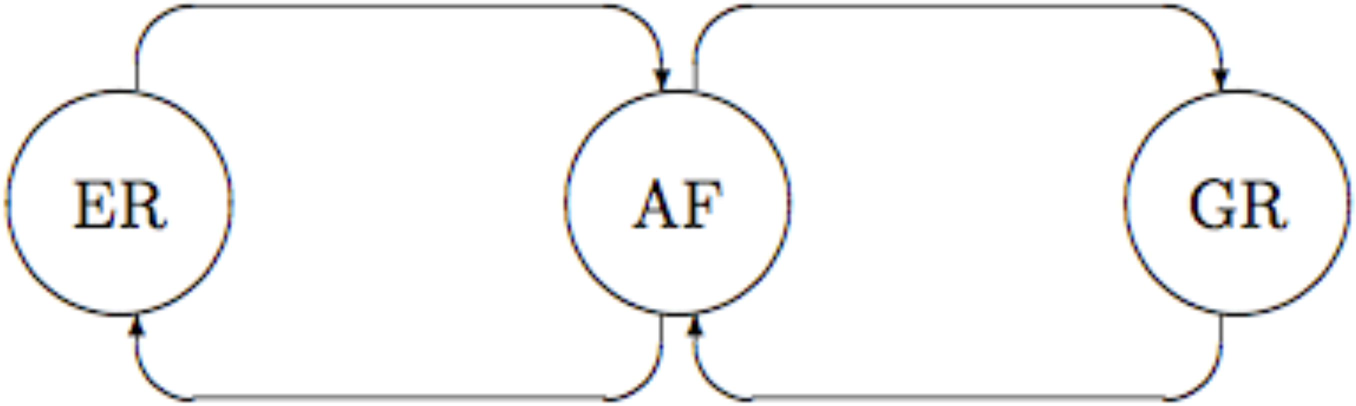


- Obtenha o AFN para a seguinte GR

$$P \to \text{aP} \mid \text{bP} \mid \text{aA}$$
$$A \to \text{a} \mid \text{bB}$$
$$B \to \text{bA}$$

Síntese

- AF's, ER's e GR's são formalismos alternativos para linguagens regulares



Transformações entre formalismos