Teoria da Computação
Simplificação de Gramáticas Livres de Contexto

Prof. Jefferson Magalhães de Morais

Simplificação de GLC

- Tem por objetivo tornar a gramática mais simples ou de prepará-las para posteriores aplicações.
- É importante notar que, qualquer que seja a transformação efetuada, a linguagem gerada deverá ser sempre a mesma.

Simplificação de GLC

- Não reduzem o poder de expressão das GLC

- São importantes para:
  - Construção e otimização de algoritmos
  - Demonstrações de teoremas

Simplificando GLC

- São simplificações:
  - exclusão de símbolos inúteis
    - variáveis ou terminais não-usados
    - para gerar palavras de terminais
  - exclusão de produções vazias da forma $A \rightarrow \varepsilon$
    - se ε pertence à linguagem,
    - é incluída uma produção vazia específica
  - exclusão de produções da forma $A \rightarrow B$
    - substituem uma variável por outra
    - não adicionam qualquer informação de geração de palavra

Eliminação de Símbolos Inúteis

- Símbolos Inúteis : um símbolo (terminal ou não-terminal) é inútil se ele não aparece na derivação de nenhuma sentença.
- Podendo ser:
  - **Estéril**: se não gera nenhuma sequência de terminais pertencente a uma sentença
  - **Inalcançável**:se não aparece em nenhuma forma sentencial da gramática.

Determinação do conjunto de símbolos férteis

- Qualquer variável gera palavra de terminais
  - gera um novo conjunto de variáveis
  - inicialmente, considera todas as variáveis que geram terminais diretamente (ex: $A \rightarrow a$)
  - a seguir, são adicionadas as variáveis que geram palavras de terminais indiretamente (ex: $B \rightarrow Ab$)

Determinação do conjunto de símbolos férteis

- Pode ser efetuada através do seguinte algoritmo:
  - Construir o conjunto $N0 = \emptyset$ e fazer $i = 1$
  - Repetir

    $Ni = Ni\text{-}1 \cup \{ A \mid A \rightarrow \alpha \in P \text{ e } \alpha \in (Ni\text{-}1 \cup T)^* \}$

    $i = i + 1$
  - até que $Ni = Ni\text{-}1$
  - Ni é o conjunto de símbolos férteis.
- Se o símbolo inicial não fizer parte do conjunto de símbolos férteis, a linguagem gerada pela gramática é vazia.

Exemplo

- Retirar os símbolos estéreis da gramática: G = ({S,A,B,C,D}, {a,b,c,d}, P, S)
  - P: S → a A

    A → a | b B

    B → b | d D

    C → cC | c

    D → d D

Exemplo

- Solução
  - $N0 = \emptyset$
  - $N1 = \{A,B,C\}$
  - $N2 = \{S,A,B,C\}$
  - $N3 = \{S,A,B,C\} = N2$
- Conjunto de símbolos férteis: {S,A,B,C}
- Gramática simplificada:
  - G’ = ({S,A,B,C}, {a,b,c}, P’, S)
  - P’:
    - $S \rightarrow a\ A$
    - $A \rightarrow a \mid b\ B$
    - $B \rightarrow b$
    - $C \rightarrow cC \mid c$

Determinação do conjunto de símbolos alcançáveis

- Qualquer símbolo é atingível a partir do símbolo inicial
  - analisa as produções da gramática a partir do símbolo inicial
  - inicialmente, considera exclusivamente o símbolo inicial
  - após, as produções da gramática são aplicadas e os símbolos referenciados adicionados aos novos conjuntos

Determinação do conjunto de símbolo alcançáveis

- Pode ser efetuada através do seguinte algoritmo:
  - Construir o conjunto $V0 = \{S\}$ (S = símbolo inicial) e fazer $i = 1$
  - Repetir
    - $Vi = \{ X \mid \text{existe algum } A \rightarrow \alpha X \beta \text{ e } A \in Vi\text{-}1 \text{ e } \alpha, \beta \in (N \cup T)^* \} \cup V\, i\text{-}1$
    - $i = i + 1$
  - até que $Vi = Vi\text{-}1$
  - Vi é o conjunto de símbolos alcançáveis.

Determinação do conjunto de símbolo alcançáveis - Exemplo

- Simplificar a gramática G’ do exemplo anterior, retirando os símbolos inalcançáveis.
- Solução:
    - V0 = {S}
    - V1 = {S, a, A}
    - V2 = {S, a, A, b, B}
    - V3 = {S, a, A, b, B} = V2
  - Conjunto de símbolos alcançáveis: {S, a, A, b, B}
  - Gramática simplificada:
  - G’ = ({S,A,B}, {a,b}, P’’, S)
  - P’’: $S \rightarrow a\ A$

    $A \rightarrow a \mid b\ B$

    $B \rightarrow b$

12

Transformação de uma GLC qualquer para uma GLC **ε**-Livre

- Variáveis que constituem produções vazias
  - $A \rightarrow \varepsilon$. variáveis que geram ε diretamente
  - $B \rightarrow A$. variáveis que geram ε indiretamente

Transformação de uma GLC qualquer para uma GLC ε-Livre

- Esta transformação sempre é possível e pode ser efetuada pelo seguinte algoritmo:
  - Reunir em um conjunto os não-terminais que derivam direta ou indiretamente a sentença vazia: $Ne = \{A \mid A \in N \text{ e } A \overset{+}{\rightarrow} \varepsilon\}$
  - Construir o conjunto de regras P' como segue:
    - incluir em P' todas as regras de P, com exceção daquelas da forma $A \rightarrow \varepsilon$
    - para cada ocorrência de um símbolo Ne do lado direito de alguma regra de P, incluir em P' mais uma regra, substituindo este símbolo por ε. Isto é, para regra de P do tipo $A \rightarrow \alpha B \beta$, $B \in Ne$ e $\alpha, \beta \in V^*$ incluir em P' a regra $A \rightarrow \alpha\ \beta$

Transformação de uma GLC qualquer para uma GLC ε-Livre

- Se $S \in Ne$, adicionar a P' as regras $S' \rightarrow S$ e $S' \rightarrow \varepsilon$, sendo que N' ficará igual a $N \cup S'$. Caso contrário trocar os nomes de S por S' e N por N'.

- A nova gramática será definida por: $G' = (N', T, P', S')$

Exemplo 1

- Transformar as GLC abaixo, definidas pelo respectivo
- conjunto de regras de produção P, para GLC **ε**-Livres.
  - G = ({S, D, C}, {b,c,d,e}, P, S)
  - P: S → b D C e

    D → d D | **ε**

    C → c C | **ε**
  - Solução:
    - Ne = {D, C}
    - P ‘: S → b D C e | b C e | b D e | b e

      D → d D | d

      C → c C | c

Exemplo 2

- Transformar as GLC abaixo, definidas pelo respectivo conjunto de regras de produção P, para GLC **ε**-Livres.
  - G = ({S}, {a}, P, S)
    - P: S → a S | **ε**
    - Solução:

      Ne = {S}

      P ':S' → S | **ε**

      S → a S | a

Remoção de Produções Simples

- Produções simples são produções da forma $A \rightarrow B$ onde A e $B \in N$. Onde:
  - A pode ser substituída por B
  - não adiciona informação alguma em termos de geração de palavras

Remoção de Produções Simples

- Podem ser removidas de uma GLC através do seguinte algoritmo:
  - Transformar a GLC em uma GLC **ε**-livre, se necessário
  - Para todo não-terminal de N, construir um conjunto com os não-terminais que ele pode derivar, em um ou mais passos. Isto é, para todo A ∈ N, construir NA = { B | A *→ B }
  - Construir P' como segue:
    - se B **→α** ∈ P e não é uma produção simples, adicione a P' as produções: A → **α** para todo A | B ∈ NA
  - A GLC equivalente, sem produções simples, será definida por: G' = (N, T, P', S)

Exemplo 1

- Transformar as GLC abaixo em gramáticas equivalentes que não apresentem produções simples.
  - G = ({S, A}, {a,b}, P, S)
    - P: S → b S | A

      A → a A | a
    - Solução:
      - Ns = {A}
      - NA = { }
      - P ‘: S → b S | a A | a

        A → a A | a

Exemplo 2

- Transformar as GLC abaixo em gramáticas equivalentes que não apresentem produções simples.
  - G = ({S, A, B}, {a,b,c}, P, S)
    - P: $S \rightarrow a\,S\,b \mid A$

      $A \rightarrow a\,A \mid B$

      $B \rightarrow b\,B\,c \mid b\,c$
    - Solução:

      Ns = {A, B}

      NA = {B}

      NB = {}

      P ‘: $S \rightarrow a\,S\,b \mid a\,A \mid b\,B\,c \mid b\,c$

      $A \rightarrow a\,A \mid b\,B\,c \mid b\,c$

      $B \rightarrow b\,B\,c \mid b\,c$

Simplificações combinadas

- Considerando as simplificações de gramáticas LC, nem todas as combinações de simplificação atingem o resultado desejado. Recomenda-se a seguinte sequência de simplificação:
  1. Exclusão de produções vazias
  2. Exclusão de produções simples
  3. Exclusão de símbolos inúteis

22