Discrete Event Systems

Exercise Session 5

Roland Schmid

nsg.ee.ethz.ch

ETH Zürich (D-ITET)

19 October 2023

1 Revisiting Context-Free Grammars

Consider the context-free languages from last week (cf. Exercise 4.1) on the alphabet $\Sigma = \{0, 1\}$:

a) $L_1 = \{w \mid \text{the length of } w \text{ is odd}\}$

b) $L_2 = \{w \mid \text{contains more 1s than 0s}\}$

For each of them, give a context-free grammar in Chomsky Normal Form (CNF) and try finding a grammar with the minimum number of non-terminal symbols. If possible, give a right-linear and a left-linear grammar for the language.

Chomsky Normal Form

- Definition: A CFG is said to be in Chomsky Normal Form if every rule in the grammar has one of the following forms:
  - $S \rightarrow \varepsilon$ (ε for epsilon's sake only)
  - $A \rightarrow BC$ (dyadic variable productions)
  - $A \rightarrow a$ (unit terminal productions)

  where *S* is the start variable, *A,B,C* are variables and *a* is a terminal.

- Thus epsilons may only appear on the right hand side of the start symbol and other rights are either 2 variables or a single terminal.

CFG → CNF

- Converting a general grammar into Chomsky Normal Form works in four steps:

1. Ensure that the start variable doesn't appear on the right hand side of any rule.
2. Remove all epsilon productions, except from start variable.
3. Remove unit variable productions of the form $A \rightarrow B$ where A and B are variables.
4. Add variables and dyadic variable rules to replace any longer non-dyadic or non-variable productions

1 Revisiting Context-Free Grammars

Consider the context-free languages from last week (cf. Exercise 4.1) on the alphabet $\Sigma = \{0, 1\}$:

a) $L_1 = \{w \mid \text{the length of } w \text{ is odd}\}$

b) $L_2 = \{w \mid \text{contains more 1s than 0s}\}$

For each of them, give a context-free grammar in Chomsky Normal Form (CNF) and try finding a grammar with the minimum number of non-terminal symbols. If possible, give a right-linear and a left-linear grammar for the language.

2 Regular, Context-Free or Not?

For the following languages, determine whether they are context free or not. Prove your claims!

a) $L = \{1^k \mid k \text{ prime}\}$

b) $L = \{w\#x\#y\#z \mid w, x, y, z \in \{a, b\}^* \text{ and } |w| = |z|, |x| = |y|\}$

c) $L = \{w\#x\#y\#z \mid w, x, y, z \in \{a, b\}^* \text{ and } |w| = |y|, |x| = |z|\}$

d) $L = \{x \mid x \in \{0, 1\}^*, \text{ and } x \text{ contains an even number of '0's and an even number of '1's}\}$

3 Tandem-Pumping Lemma [Exam HS21]

Given the alphabet $\Sigma = \{0, 1, \#\}$, consider the language:

$$L = \{\, a\#b\#c \mid a, b, c \in \{0,1\}^*,\ c = 2a,\ \#_0(b) = \#_0(c)\,\}$$

for unsigned binary numbers a, b, and c. For example, $0\#10\#0 \in L$ and $1\#00\#010 \in L$.

Recall: $\#_0(w)$ denotes the number of occurrences of the symbol $0 \in \Sigma$ in a word $w \in \Sigma^*$.

a) Show that $w = 1^p\#0\#1^p0$ is tandem-pumpable in L.

Hint: Split up $w = uvxyz$ such that $x = \#0\#$.

b) Use the tandem-pumping lemma to show that L is not context-free.

Hint: Choose a string $w = a\#b\#c$ where $1 \notin b$, i.e. $b \in 0^*$.

c) Can we use any string $w = a\#b\#c$ where $b = b_1 1\, b_2$ to apply the tandem-pumping lemma?

4 Java is not regular! [Bonus question]

Prove that the programming language `java` is not regular! More precisely, show that a single statement in `java` cannot be recognized by a regular language.
Hint: Assume that strings in your program do not contain the symbols "{" or "}".