Relations between Regular and Context-Free Languages

Teodor Rus

`rus@cs.uiowa.edu`

The University of Iowa, Department of Computer Science

Source

Most of the material discussing relations between regular languages and context-free languages is taken from Fleck, Chapters 2 and 4.

Linear grammar rules

Let $G = (V, \Sigma, R, S)$ be a CFG and $r \in R$.

- r is linear if $rhs(r) \in \Sigma^* \circ V \circ \Sigma^*$;
- r is *right-linear* if $rhs(r) \in \Sigma^* \circ V$;
- r is *left-linear* if $rhs(t) \in V \circ \Sigma^*$;
- r is *terminating* if $rhs(r) \in \Sigma^*$.

Linear CFG

- A CFG G is called linear if all its productions are either linear or terminating;
- A CFG G is called right-linear if all its productions are either right-linear or terminating;
- A CFG G is called left-linear if all its productions are either left-linear or terminating.

Example linear grammar

$G = (\{A, B\}, \{0, 1\}, \{A \to 0A|B, B \to 1B|\epsilon\}, A)$
is a right-linear grammar.

Derivation:
$A \Rightarrow 0A \Rightarrow 00A \Rightarrow 00B \Rightarrow 001B \Rightarrow 0011B \Rightarrow 00111B \Rightarrow 00111$
is a derivation in G.

$$L(G) = 0^*1^*$$

Theorem 2.35

If $L \subseteq \Sigma^*$ is a regular language then there is a right linear grammar G with $L(G) = L$.

Proof: by construction. Since L is regular let DFA $D = (Q, \Sigma, \delta, q_0, F)$ with $L(D) = L$ and $\Sigma \cap Q = \emptyset$. Construct $G = (Q, \Sigma, R, q_0)$ where:
$R = \{q \rightarrow tq' | t \in \Sigma \wedge \delta(q,t) = q'\} \cup \{q \rightarrow \epsilon | q \in F\}$

1. G is right-linear and directly simulates D;
2. if $x = x_1 x_2 \ldots x_n \in L(D)$ with $\delta(q_0, x_1) = q_1$, $\delta(q_1, x_2) = q_2$, $\ldots$, $\delta(q_{n-1}, x_n) = q_n$ then $q_0 \Rightarrow x_1 q_1 \Rightarrow x_1 x_2 q_2 \Rightarrow \ldots \Rightarrow x_1 x_2 \ldots x_n q_n$
3. $q_n \in F$ and $q_n \rightarrow \epsilon \in R$, i.e., $x_1 x_2 \ldots x_n \in L(G) \wedge L(D) \subseteq L(G)$.
4. Similarly, $L(G) \subseteq L(D)$. Therefore $L(D) = L(G)$.

Corollary

No regular language is inherently ambiguous.

Proof: Let L be regular and DFA D accepting it.

- Each $x \in L$ has a unique accepting run in D;
- D runs are in one-to-one correspondence with the leftmost derivations in the associated grammar G_D;
- Hence, each $x \in L$ has a unique leftmost derivation in G_D.

Example

DFA D in Figure 1 accepts the language 0^*1^*

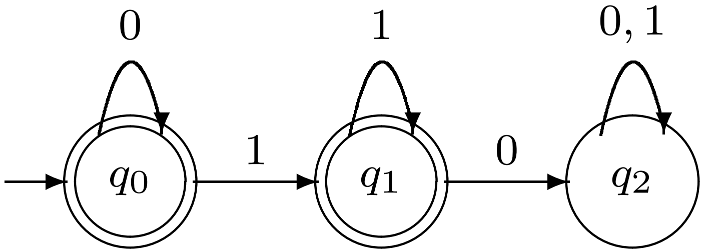

Figure 1: DFA accepting the language 0^*1^*

Grammar associated with D

- Using the construction from the proof we obtain $G_D = (\{q_0, q_1, q_2\}, \{0, 1\}, \{q_0 \rightarrow 0q_0|1q_1|\epsilon, q_1 \rightarrow 1q_1|0q_2|\epsilon, q_2 \rightarrow 0q_2|1q_2\}, q_0)$
- Since q_2 is dead we can eliminate its rules obtaining:

  $G_D = (\{q_0, q_1\}, \{0, 1\}, \{q_0 \rightarrow 0q_0|1q_1|\epsilon, q_1 \rightarrow 1q_1|\epsilon\}, q_0)$

Lemma

For each right-linear grammar $G = (N, \Sigma, P, S)$ there is a right-linear grammar $G' = (V', \Sigma, P', S)$ where each non-terminating rule $A \to \alpha \in P'$ has $|\alpha| \leq 2$ and each terminating rule $A \to x \in P'$ has $|x| \leq 1$.

Proof idea: Note that G does not forbid erasing ϵ or unit rules (ϵ may belong to a regular language). For $A \to \alpha \in P$ with $|\alpha| > 2$, to transform it equivalently we can use construction from Chomsky normal form.

Example

Consider $G = (\{A, B\}, \{a, b\}, \{A \to abA|abB, B \to aaaB|b\}, A)$. G' stated in the above Lemma is

$G' = (\{A, X_1, Y_1, B, Z_1, Z_2\}, \{a, b\}, R', A)$ where R' is:

$A \to aX_1|aY_1, X_1 \to bA, Y_1 \to bB,$

$B \to aZ_1|b, Z_1 \to aZ_2, Z_2 \to aB$

Theorem 2.36

For each right-linear grammar G, $L(G)$ is regular.

Proof: by construction

1. For $G = (V, \Sigma, R, S)$, using Lemma 2.20 we may assume that R has only productions of the form $X \rightarrow xY$ and $X \rightarrow z$, $X, Y \in V$, $z \in \Sigma \cup \{\epsilon\}$.
2. An NFA M that accepts $L(G)$ is $M = (V \cup \{\theta\}, \Sigma, \delta, S, \{\theta\})$ where for each $X \in V$ and $z \in \Sigma \cup \{\epsilon\}$:
   - if $X \rightarrow z \in R$ then $\delta(X, z) = \{\theta\} \cup \{Y | X \rightarrow zY \in R\}$
   - if $X \rightarrow z \notin R$ then $\delta(X, z) = \{Y | X \rightarrow zY \in R\}$
3. It can be directly checked that $L(M) = L(G)$.

Example

The NFA M accepting the language specified by $G' = (\{A, X_1, Y_1, B, Z_1, Z_2\}, \{a, b\}, R', A)$ where R' is: $A \to aX_1|aY_1, X_1 \to bA, Y_1 \to bB, B \to aZ_1|b, Z_1 \to aZ_2, Z_2 \to aB$ is in Figure 2

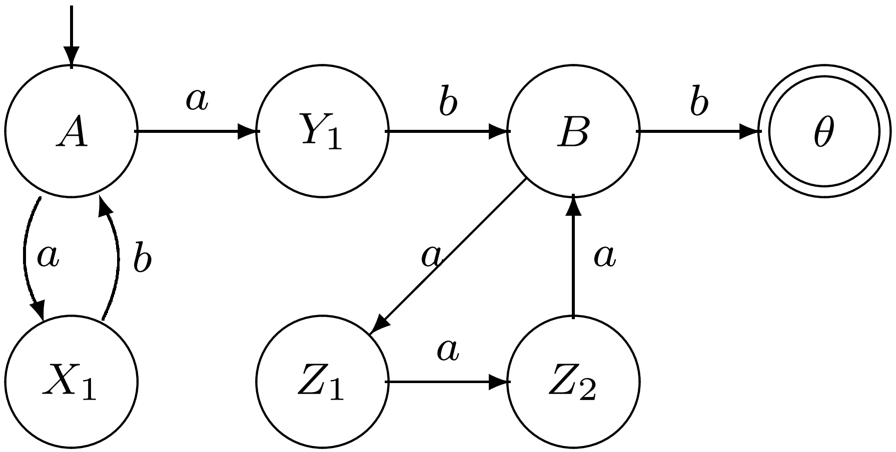

Figure 2: NFA equivalent to G'

Fact

- There is a perfect duality between recognition and generation of regular languages: when an acceptor makes an atomic step it consumes one input symbol; when a grammar rule is used in an atomic step it produces one symbol;
- The results established for right-linear grammars hold also for left-linear grammars;
- The term regular grammar is further used to refer to a CFG which is right or left linear.

Relationship

- The results produced here for regular grammars show that any regular language, RL, is a context-free language, CFL;
- However, as we know, the language $L = \{0^n1^n | n \geq 0\}$ is not regular;
- A CFG that generates $L = \{0^n1^n | n \geq 0\}$ is: $G = (\{S\}, \{0,1\}, \{S \rightarrow 0S1|\epsilon\}$ which shows that not all context free languages are regular.

Consequence: $RL \subset CFL$.