CS340: Theory of Computation — Sem I 2019-20

Lecture Notes 5: Properties of regular languages

Raghunath Tewari — IIT Kanpur

1 Closure Properties

We have already seen that regular languages are closed under union, concatenation and star operations. We will discuss some more closure properties of regular languages.

1.1 Complement, Intersection and Set Difference

It is easy to see that regular languages are closed under complement. If $D = (Q, \Sigma, \delta, q_0, F)$ is a DFA for a regular language L then a DFA for $\overline{L}$ is $D' = (Q, \Sigma, \delta, s, Q \setminus F)$. That is the DFA whose accept states are the non-accept states of the DFA D and vice versa. Then if $w \in L(D)$ then $w \notin L(D')$ and $w \notin L(D)$ then $w \in L(D')$.
Using De Morgan's Law,

$$A \cap B = \overline{\overline{A} \cup \overline{B}}.$$

Since regular languages are closed under union and complement, hence they are also closed under intersection.
$A \setminus B = A \cap \overline{B}$. Hence regular languages are closed under set difference.

1.2 Reversal

Let $w = a_1 a_2 \dots a_n$ be a string. Then $\text{rev}(w) = a_n a_{n-1} \dots a_1$. Extending the definition, we say that for a language $L \subseteq \Sigma^*$, $\text{rev}(L) = \{\text{rev}(w) \mid w \in L\}$.

Theorem 1. *If L is regular then* $\text{rev}(L)$ *is also regular.*

Consider a DFA $D = (Q, \Sigma, \delta, q_0, F)$ such that $L = L(D)$. Now any string that is in the language L, will start at the start state q_0 and end up at one of the accept states in F . To design an automaton for $\text{rev}(L)$ we will invert the transitions of D. Since we do not know a priori in which state a string would be accepted, we would use nondeterminism to "guess" a starting position in the reversed automaton. Here is the formal construction. Let $D' = (Q', \Sigma, \delta', q_0', F')$ be an NFA for $\text{rev}(L)$ defined as follows.

- $Q' = Q \cup \{q_0'\}$.
- $\delta'(q_0', \epsilon) = F$
  $\delta'(q, a) = \{r \mid \delta(r, a) = q\}$
- $F' = \{q_0\}$

1.3 First-Halves

For a language $L \subseteq \Sigma^*$, define

$$\text{FirstHalves}(L) = \{x \mid \exists y \text{ such that } |x| = |y|, xy \in L\}.$$

For example, let $L = \{0, 10, 110, 1011, 100110\}$ then $\text{FirstHalves}(L) = \{1, 10, 100\}$.

Theorem 2. *If L is regular then* FirstHalves(L) *is also regular.*

Let $D = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $L = L(D)$. We will design a pebble game on the DFA D, corresponding to the language FirstHalves(L) and then use the game to construct an automaton for FirstHalves(L).

1.3.1 Idea of the Construction

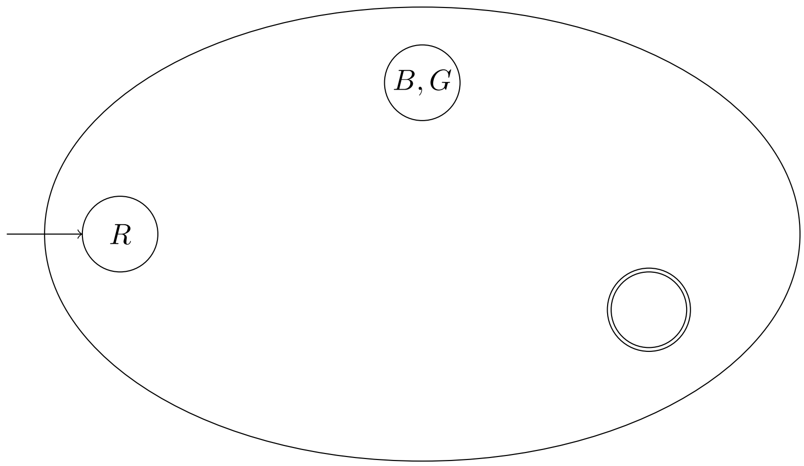

Figure 1: The DFA D: Initial configuration of the game

1. Starting configuration of the game

   - The red pebble R is placed at the start state of the DFA D.
   - The blue pebble B and the green pebble G are together placed at a nondeterministically chosen state of D (see the Figure 1.3.1).

   Let $w \in \Sigma^*$. Then R will correspond to tracing the first half of the string w, G will correspond to tracing the second half of the string, and B will remember the initial position of G.

2. Moves of the game.

   - R moves according to the transition function of D.
   - B remains static.
   - For every step of R, G takes one step nondeterministically.

3. Winning configuration of the game

   - R and B are in the same state.
   - G is in some accept state of D (see Figure 2).

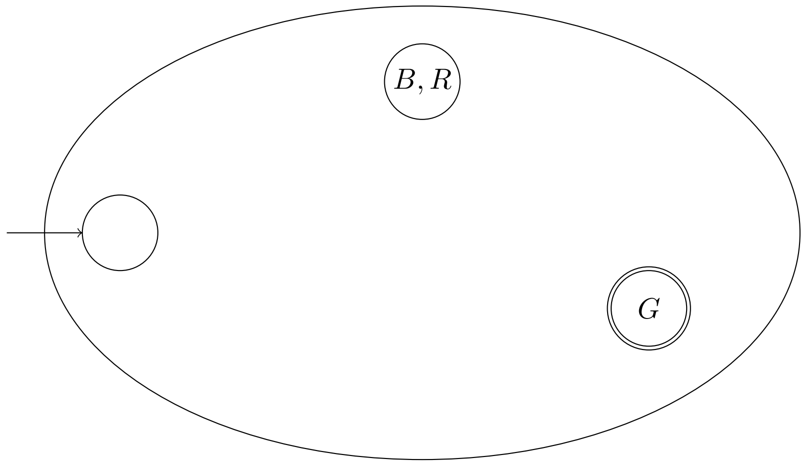

Figure 2: The DFA D: Winning configuration of the game

1.3.2 Formal Construction of the NFA

We will now design an NFA for FirstHalves(L) based on the above game. Let $N = (Q', \Sigma, \delta', q_0', F')$ where

- $Q' = Q^3 \cup \{q_s\}$, where q_s is an additional state.
- $$\begin{aligned} \delta'(q_s, \epsilon) &= \{(q_0, q, q) \mid q \in Q\} \\ \delta'((p, q, r), a) &= \{(\delta(p, a), q, (\delta(r, b)) \mid b \in \Sigma\} \end{aligned}$$
- $q_0' = q_s$.
- $F' = \{(q, q, f) \mid q \in Q, f \in F\}$

1.3.3 Proof of Correctness

We will show that FirstHalves(L) $= L(N)$. For a string $x \in \Sigma^*$ we will use the notation $\delta(q, x)$ to denote the state (resp. set of states) reachable from q on the string x when δ is the transition function of a DFA (resp. NFA).

Let $x \in$ FirstHalves(L). Then by definition of FirstHalves(L), there exists $y \in \Sigma^*$ such that $xy \in L$ and $|x| = |y|$. Let $\delta(q_0, x) = r$ and $\delta(q_0, xy) = f$. Since $xy \in L$ therefore $f \in F$. Also this implies that $\delta(r, y) = f$. Now, $\delta'(q_s, \epsilon) \ni (q_0, r, r)$ and

$$\begin{aligned} \delta'((q_0, r, r), x) &\ni (\delta(q_0, x), r, \delta(r, y)) \\ &= (r, r, f). \end{aligned}$$

By definition of F', $(r, r, f) \in F'$. Hence $x \in L(N)$.

Now for the other direction let $x \in L(N)$. There exists a state $r \in Q$ and a state $f \in F$ such that $\delta'((q_0, r, r), x) \ni (r, r, f)$. This gives us that $\delta(q_0, x) = r$. Also according to the definition of δ', for every step of the first coordinate of a tuple in Q^3, the third coordinate also takes exactly one step. Hence there exists a $y \in \Sigma^*$ such that $|x| = |y|$ and $\delta(r, y) = f$. This implies that $\delta(q_0, xy) = f$ and therefore $x \in$ FirstHalves(L).

Remark. The previous example illustrates the use of the *product automaton* construction. A similar construction can be used to show closure of regular languages under intersection.

Exercise 1. 1. For a language $L \subseteq \Sigma^*$, define

$$\text{SecondHalves}(L) = \{y \mid \exists x \text{ such that } |x| = |y|, xy \in L\}.$$

2. For a language L, let

$$\text{MiddleThirds}(L) = \{y \mid \exists x, z \text{ and } |x| = |y| = |z| \text{ and } xyz \in L\}$$

For example, $\text{MiddleThirds}(\{\epsilon, a, ab, bab, bbab, aabbab\}) = \{\epsilon, a, bb\}$.
Prove that if L is regular, $\text{MiddleThirds}(L)$ is also regular.

3. Given $L \subseteq \{0, 1\}^*$, define

$$L' = \{xy \mid x1y \in L\}.$$

Show that if L is regular then L' is also regular.

4. For a language A, let

$$A'' = \{xz \mid \exists y \text{ and } |x| = |y| = |z| \text{ and } xyz \in A\}$$

Show that even if A is regular, A'' is not necessarily regular.