1 Proof of correctness of union construction

Last time we constructed a machine that we claimed recognized the union of two regular languages. That is, given $M_1 = (Q_1, \Sigma, \delta_1, F_1, q_{01})$ and $M_2 = (Q_2, \Sigma, \delta_2, F_2, q_{02})$, we constructed a machine $M$ and claimed that $L(M) = L(M_1) \cup L(M_2)$. In this lecture, we proved that the construction is correct.

Here is the construction: $M = (Q_1 \times Q_2, \Sigma, \delta, F, (q_{01}, q_{02}))$ where

$$
\delta(((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))
$$

$$
F = \{(q_1, q_2) \mid q_1 \in F_1 \text{ or } q_2 \in F_2\}
$$

To prove that this construction is correct, we first proved that the extended transition function behaves as expected.

**Claim:** for all $x \in \Sigma^*$, $q_1 \in Q_1$ and $q_2 \in Q_2$, $\tilde{\delta}((q_1, q_2), x) = (\tilde{\delta}_1(q_1, x), \tilde{\delta}_2(q_2, x))$.

**Proof:** by induction on the structure of $x$. If $x = \epsilon$, then the left-hand side is just $(q_1, q_2)$ (by definition of $\tilde{\delta}$). Similarly, plugging in the definition of $\tilde{\delta}$ into the right hand side also yields $(q_1, q_2)$. So in the $x = \epsilon$ case the claim holds.

If $x = ya$, we can simplify the left-hand side:

$$
\tilde{\delta}((q_1, q_2), xa) = \delta(\tilde{\delta}((q_1, q_2), x), a)
$$

$$
= \delta((\delta_1(q_1, x), \delta_2(q_2, x)), a)
$$

$$
= (\delta_1(\delta_1(q_1, x), a), \delta_2(\delta_2(q_2, x), a))
$$

by definition of $\tilde{\delta}$

by Inductive Hypothesis

by definition of $\delta$

by definition of $\delta_1$ and $\delta_2$

Which is equal to the right hand side. This concludes the inductive proof of the claim.
Using this fact and the definition of $L(M)$ we can prove that $L(M) = L(M_1) \cup L(M_2)$. We have

$$L(M) = \{ x \in \Sigma^* \mid \hat{\delta}((q_{01}, q_{02}), x) \in F \}$$

by definition

$$= \{ x \in \Sigma^* \mid \hat{\delta}_1(q_{01}, x) \in F_1 \text{ or } \hat{\delta}_2(q_{02}, x) \in F_2 \}$$

by definition of $F$ and the claim

$$= \{ x \in \Sigma^* \mid \hat{\delta}_1(q_{01}, x) \in F_1 \} \cup \{ x \in \Sigma^* \mid \hat{\delta}_2(q_{02}, x) \in F_2 \}$$

$$= L(M_1) \cup L(M_2)$$

as claimed.

2 Regular expressions

We discussed another way of describing sets of strings, regular expressions. Regular expressions are a compact way of representing certain sets of strings.

The set of regular expressions is defined inductively:

- $\emptyset$ is a regular expression.
- $\epsilon$ is a regular expression.
- for all $a \in \Sigma$, $a$ is a regular expression.
- if $r_1$ and $r_2$ are regular expressions, then $r_1r_2$ is a regular expression (called the concatenation of $r_1$ and $r_2$).
- if $r_1$ and $r_2$ are regular expressions, then $r_1|r_2$ is a regular expression (called the alternation of $r_1$ and $r_2$).
- if $r$ is a regular expression, then so is $r^*$, called the Kleene closure of $r$.

Each regular expression has an associated language of strings that it matches, written as $L(r)$. The function $L$ is defined inductively:

- $L(\emptyset) = \emptyset$
- $L(\epsilon) = \{ \epsilon \}$
- $L(a) = \{ a \}$
- $L(r_1r_2) = \{ xy \mid x \in L(r_1) \text{ and } y \in L(r_2) \}$
- $L(r_1|r_2) = L(r_1) \cup L(r_2)$
- $L(r^*) = \{ x_1x_2x_3 \cdots x_n \mid x_i \in L(r) \}$