Finite automata and formal languages (DIT323, TMV029)

Nils Anders Danielsson

2023-02-06

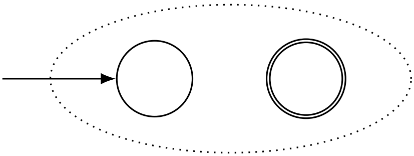
Today

- Translation from regular expressions to finite automata.
- More about regular expression algebra.
- The pumping lemma for regular languages.
- Some closure properties for regular languages.

Translating regular expressions to automata

Regular expressions to automata

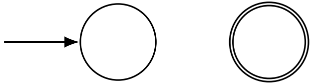
Given a regular expression in $RE(\Sigma)$ we construct an ε-NFA (with alphabet Σ) with exactly one accepting state, no transitions from the accepting state, and no transitions to the start state:



The translation is defined recursively.

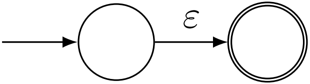
The empty language

$\varepsilon\text{-}NFA(\emptyset) =$



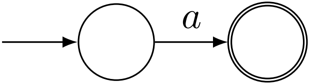
The empty string

$\varepsilon\text{-}NFA(\varepsilon) =$



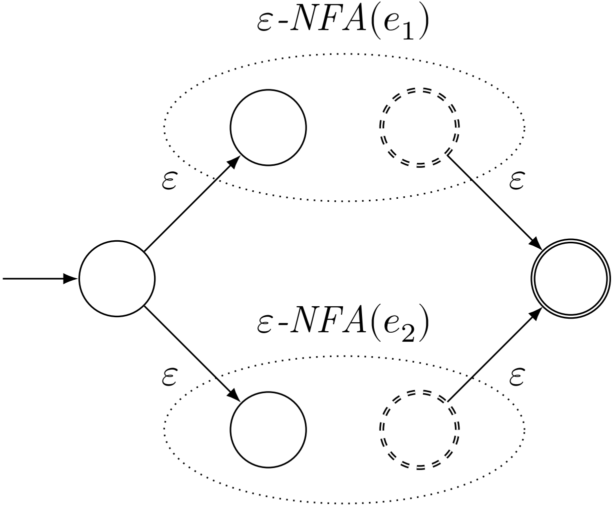
A symbol

$\varepsilon\text{-}NFA(a) =$



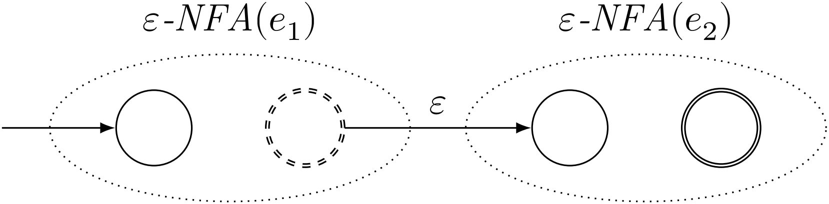
Alternation

$\varepsilon\text{-}NFA(e_1 + e_2) =$



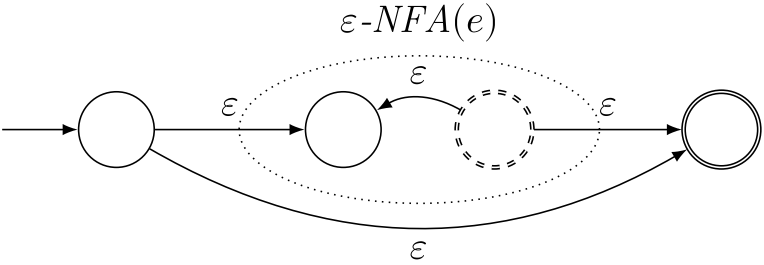
Sequencing

$\varepsilon\text{-}NFA(e_1 e_2) =$

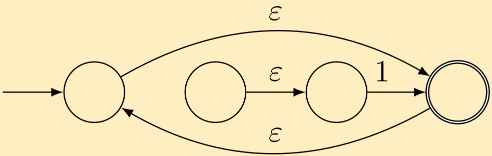


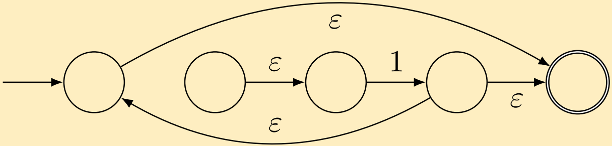
Kleene star

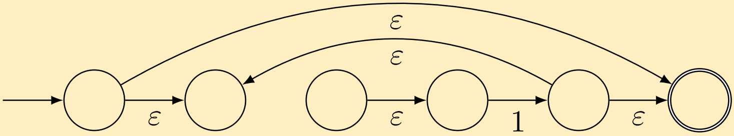
$\varepsilon\text{-}NFA(e^*) =$



Which of the following ε-NFAs is equal to $\varepsilon\text{-}NFA((\emptyset 1)^*)$ (ignoring the alphabet and the names of the states)?

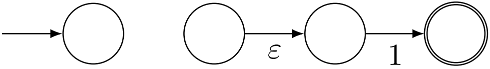
1. 

2. 

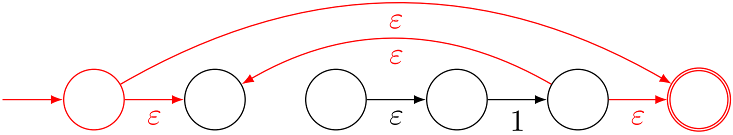
3. 

Respond at `https://pingo.coactum.de/375102`.

$\varepsilon\text{-}NFA(\emptyset 1) =$



$\varepsilon\text{-}NFA((\emptyset 1)^*) =$



Regular languages

- Recall that a language $M \subseteq \Sigma^*$ is regular if there is some DFA A with alphabet Σ such that $L(A) = M$.
- A language $M \subseteq \Sigma^*$ is also regular if and only if there is some *regular expression* $e \in RE(\Sigma)$ such that $L(e) = M$.

More about regular expression algebra

Discovering and proving laws

- In the last lecture I mentioned that $(\varepsilon + e)^* = e^*$.
- How can you figure out that this holds?
- And how can you prove it?

Proving laws

- Recall that $e_1 = e_2$ means that $L(e_1) = L(e_2)$.
- We can prove $L(e_1) = L(e_2)$ by proving $L(e_1) \subseteq L(e_2)$ and $L(e_2) \subseteq L(e_1)$, i.e. that $\forall w \in L(e_1).\ w \in L(e_2)$ and $\forall w \in L(e_2).\ w \in L(e_1)$.

Let $e \in RE(\Sigma)$. Then $(\varepsilon + e)^* = e^*$.

$L((\varepsilon + e)^*) \subseteq L(e^*)$:

- If $w \in L((\varepsilon + e)^*)$, then there is some $n \in \mathbb{N}$ such that $w = w_1 \cdots w_n$ and each string w_i is either ε or a member of $L(e)$.
- Remove the strings w_i that are equal to ε.
- We get a string $w' = w_{k_1} \cdots w_{k_m}$, for some natural numbers m and $k_1 < \cdots < k_m$.
- Because all strings w_{k_i} belong to $L(e)$ we get that $w' \in L(e^*)$.
- Furthermore $w = w'$, so $w \in L(e^*)$.

Let $e \in RE(\Sigma)$. Then $(\varepsilon + e)^* = e^*$.

$L(e^*) \subseteq L((\varepsilon + e)^*)$:

- We have that $L(e) \subseteq L(\varepsilon) \cup L(e) = L(\varepsilon + e)$.
- The result follows by monotonicity of $-^*$.

Monotonicity

- $M \subseteq N$ implies that $M^* \subseteq N^*$.
- $M_1 \subseteq N_1$ and $M_2 \subseteq N_2$ imply:
  - $M_1 \cup M_2 \subseteq N_1 \cup N_2$.
  - $M_1 \cap M_2 \subseteq N_1 \cap N_2$.
  - $M_1 M_2 \subseteq N_1 N_2$.

Discovering (and proving) laws

- A regular expression proposition $e_1 = e_2$ is valid iff the equation obtained by replacing each variable e by a *fresh* symbol a is true.
- Examples:
  - $(\varepsilon + e)^* = e^*$ is valid iff $(\varepsilon + 1)^* = 1^*$ is true.
  - $e_1 1 e_2 = e_2 1 e_1$ is valid iff $012 = 210$ is true.
- Next lecture: An algorithm for checking if two regular languages are equal.

Which of the following regular expression equivalences are valid?

1. $\emptyset^* e = e$.
2. $(e_1 + e_2)^* = e_1^* + (e_1 e_2)^* + e_2^*$.
3. $e_1(e_2 e_1)^* = (e_1 e_2)^* e_1$.
4. $(e_1 + e_2)^* = (e_1^* e_2)^* e_1^*$.
5. $(e_1 + e_2)^* = e_1^*(e_2 e_1^* e_2)^* e_1^*$.

Respond at `https://pingo.coactum.de/375102`.

The shifting and denesting rules

1. The shifting rule: $e_1(e_2e_1)^* = (e_1e_2)^*e_1$.
2. The denesting rule: $(e_1 + e_2)^* = (e_1^*e_2)^*e_1^*$.

The denesting rule

Consider the following equations:

$$e_1 = e_2$$
$$e_2 = 0e_1 + 1e_2 + \varepsilon$$

One way to find a solution for e_1, using Arden's lemma:

$$e_2 = (0+1)e_2 + \varepsilon$$
$$e_2 = (0+1)^*\varepsilon = (0+1)^*$$
$$e_1 = (0+1)^*$$

Another way:

$$e_2 = 1^*(0e_1 + \varepsilon)$$
$$e_1 = 1^*0e_1 + 1^*$$
$$e_1 = (1^*0)^*1^*$$

One can combine methods

Is it the case that $((\varepsilon + e)^*)^* \subseteq (1 + e)^*$?

- We know that $(\varepsilon + e)^* = e^*$, so $((\varepsilon + e)^*)^* = (e^*)^*$.
- We also have $e \subseteq 1 + e$, and thus, by monotonicity, $e^* \subseteq (1 + e)^*$.
- We can conclude if $(e^*)^* = e^*$.
- This holds if $(1^*)^* = 1^*$.
- We have $1^* \subseteq (1^*)^*$.
- We also have $(1^*)^* \subseteq 1^*$, because a string in $(1^*)^*$ consists of an arbitrary number of 1s, and is thus a member of 1^*.

More laws related to the Kleene star

1. $e^* = \varepsilon + ee^*$.
2. $e^*e^* = e^*$.
3. $(e^*)^* = e^*$.

The pumping lemma

The pumping lemma for regular languages

For every regular language L
over the alphabet Σ:

$$
\begin{aligned}
&\exists m \in \mathbb{N}.\\
&\quad \forall w \in L.\ |w| \geq m \Rightarrow\\
&\qquad \exists t, u, v \in \Sigma^*.\\
&\qquad\quad w = tuv \wedge |tu| \leq m \wedge u \neq \varepsilon \wedge\\
&\qquad\quad \forall n \in \mathbb{N}.\ tu^n v \in L
\end{aligned}
$$

The pumping lemma for regular languages

For every regular language L
over the alphabet Σ:

$$
\begin{aligned}
&\exists m \in \mathbb{N}. \\
&\quad \forall w \in L.\ |w| \geq m \Rightarrow \\
&\quad\quad \exists t, u, v \in \Sigma^*. \\
&\quad\quad\quad w = tuv \wedge |tu| \leq m \wedge u \neq \varepsilon \wedge \\
&\quad\quad\quad \forall n \in \mathbb{N}.\ tu^n v \in L
\end{aligned}
$$

The pumping lemma for regular languages

Proof sketch:

- There is at least one DFA $A = (Q, \Sigma, \delta, q_0, F)$ such that $L(A) = L$.
- Let $m = |Q|$.
- If a string $w \in \Sigma^*$ with $|w| \geq |Q|$ is accepted by A, then, by the pigeonhole principle, $\hat{\delta}(q_0, w_1 \cdots w_i) = \hat{\delta}(q_0, w_1 \cdots w_j)$ for some $i, j \in \{0, ..., |Q|\}$, $i < j$.
- Let $t = w_1 \cdots w_i$, $u = w_{i+1} \cdots w_j$, $v = w_{j+1} \cdots w_{|w|}$.
- Note that $tuv = w$, $|tu| \leq |Q|$ and $u \neq \varepsilon$.
- Furthermore $tv \in L$, $tu^2v \in L$, $tu^3v \in L$, ...

Is the language $P \subseteq \{\,(,)\,\}^*$ regular?

$$\frac{}{\varepsilon \in P} \qquad\qquad \frac{w \in P}{(w) \in P}$$

1. Yes.
2. No.

Respond at `https://pingo.coactum.de/375102`.

$$\frac{}{\varepsilon \in P} \qquad \frac{w \in P}{(w) \in P}$$

No, P is not regular:

- Assume that P is regular.
- By the pumping lemma there is some $m \in \mathbb{N}$ such that, for all $w \in P$ for which $|w| \geq m$, there are strings $t, u, v \in \{\, (,)\, \}^*$ such that $w = tuv$, $|tu| \leq m$, $u \neq \varepsilon$ and, for all $n \in \mathbb{N}$, $tu^n v \in P$.
- Let w be the string $(^m)^m$.
- Note that $w \in P$ and $|w| \geq m$.

$$\frac{}{\varepsilon \in P} \qquad\qquad \frac{w \in P}{(w) \in P}$$

- We get that there are strings $t, u, v \in \{\, (,)\, \}^*$ such that $(^m)^m = tuv$, $|tu| \leq m$, $u \neq \varepsilon$ and, for all $n \in \mathbb{N}$, $tu^n v \in P$.
- Because $(^m)^m = tuv$ and $|tu| \leq m$ we know that u consists only of left parentheses, and because $u \neq \varepsilon$ we know that u consists of at least one left parenthesis.
- We also know that $tv \in P$. However, this is contradictory, because tv contains more right parentheses than left parentheses.

Necessary, not sufficient

- I have seen students try to use the pumping lemma to prove that a language *is* regular.
- However, there are non-regular languages that satisfy the pumping lemma's formula ("$\exists m \in \mathbb{N}.$ …").

Closure properties

Closure properties

Let $M, N \subseteq \Sigma^*$ be regular languages. Then

- M^* is regular,
- MN is regular,
- $M \cup N$ is regular,
- $M \cap N$ is regular,
- $\Sigma^* \setminus N$ is regular, and
- $M \setminus N$ is regular.
  (Note that $M \setminus N = M \cap (\Sigma^* \setminus N)$.)

For which of the following definitions of M is $M \setminus \{ 1^n \mid n \in \mathbb{N}, n > 0 \}$ regular?

1. $M = \{ 1 \} \cup L((21)^*)$.
2. $$\frac{}{\varepsilon \in M} \qquad \frac{w \in M}{1w2 \in M}$$
3. $$\frac{}{\varepsilon \in M} \qquad \frac{w \in M}{1w1 \in M}$$

Respond at `https://pingo.coactum.de/375102`.

Today

- Translation from regular expressions to finite automata.
- More about regular expression algebra.
- The pumping lemma for regular languages.
- Some closure properties for regular languages.

Next lecture

- Various algorithms.
- Equivalence of states.