Lecture 7 – Equivalence of Regular Expressions and Finite Automata

COSE215: Theory of Computation

Jihyeok Park

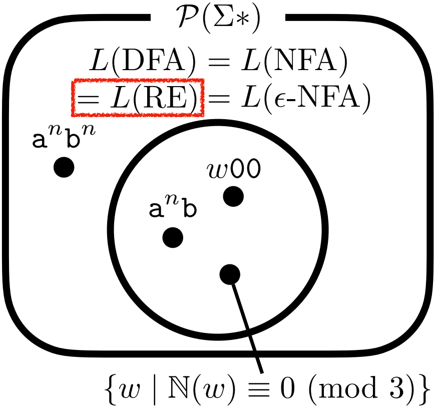
PLRG

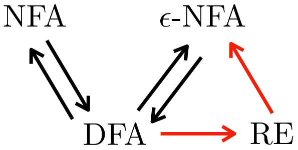
2023 Spring

Recall

1. Operations in Languages
   - Union
   - Concatenation
   - Kleene Star
2. Regular Expressions
   - Definition
   - Language of Regular Expressions
   - Extended Regular Expressions
   - Examples

Equivalence of REs and FA





Contents

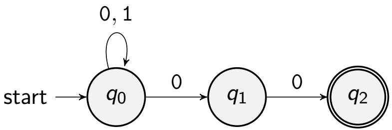
Regular Expressions to ϵ-NFA

Theorem (Regular Expressions to ϵ-NFA)

For a given regular expression R, $\exists$ ϵ-NFA N_ϵ. $L(R) = L(N_\epsilon)$.

$(0|1)^*00$

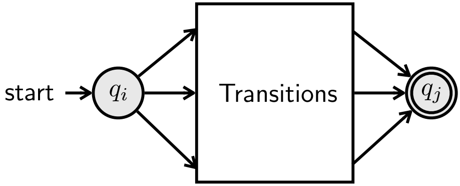
ϵ-NFA N_ϵ



Regular Expressions to ϵ-NFA

For a given regular expression R and an integer i, we will construct an ϵ-NFA $N_\epsilon = (Q, \Sigma, \delta, q_i, F)$ that accepts the language of R. It satisfies the following properties:

- Exactly one final state q_j for some j greater than i ($F = \{q_j\} \land j > i$)
- States are q_i, q_{i+1}, $\cdots$, and q_j ($Q = \{q_k \mid i \leq k \leq j\}$)
- No transition to the initial state ($\forall q \in Q.\ \forall a \in \Sigma \cup \{\epsilon\}.\ q_i \notin \delta(q, a)$)
- No transition out of the final state ($\forall a \in \Sigma \cup \{\epsilon\}.\ \delta(q_j, a) = \varnothing$)



ϵ-NFA for (R, i)

Regular Expressions to ϵ-NFA

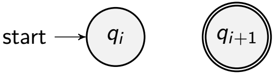
```scala
// The type definitions of states and symbols
type State = Int
type Symbol = Char

// A transition allowing epsilon
type Transition = (State, Option[Symbol], State)

// A simplified epsilon-NFA
case class SimpleENFA(from: State, trans: Set[Transition], to: State)

// Convert a regular expression to a simple epsilon-NFA with an initial state
def RE2SimpleENFA(re: RE, i: State): SimpleENFA = re match
  case REEmpty()            => ???
  case REEpsilon()          => ???
  case RESymbol(symbol)     => ???
  case REUnion(re1, re2)    => ???
  case REConcat(re1, re2) => ???
  case REStar(re)           => ???
  case REParen(re)          => ???

// Convert a simple epsilon-NFA to an epsilon-NFA
def SimpleENFA2ENFA(senfa: SimpleENFA): ENFA = ...
```

Regular Expressions to ϵ-NFA

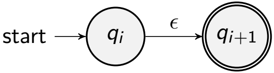
PLRG

For a given regular expression R and an integer i, the ϵ-NFA for (R, i) is:

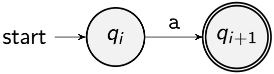
- $R = \varnothing$:



- $R = \epsilon$:



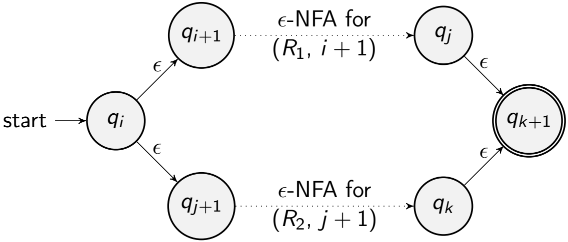
- $R = \text{a}$:




```
case REEmpty()          => SimpleENFA(
  from = i,  trans = Set(),                           to = i + 1,
)
case REEpsilon()        => SimpleENFA(
  from = i,  trans = Set((i, None, i + 1)),           to = i + 1,
)
case RESymbol(symbol) => SimpleENFA(
  from = i,  trans = Set((i, Some(symbol), i + 1)),  to = i + 1,
)
```

Regular Expressions to ϵ-NFA

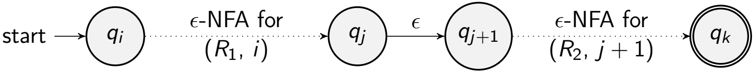
- $R = R_1 \mid R_2$:



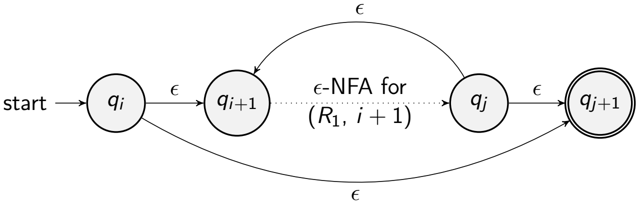

```
case REUnion(re1, re2) =>
  val SimpleENFA(_, trans1, j) = RE2SimpleENFA(re1, i + 1)
  val SimpleENFA(_, trans2, k) = RE2SimpleENFA(re2, j + 1)
  SimpleENFA(
    from  = i,
    trans = trans1 ++ trans2 ++ Set(
      (i, None, i + 1), (i, None, j + 1),
      (j, None, k + 1), (k, None, k + 1),
    ),
    to    = k + 1,
  )
```

Regular Expressions to ϵ-NFA

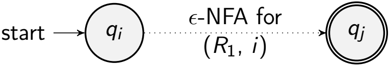
- $R = R_1 \cdot R_2$:



- $R = R_1^*$:

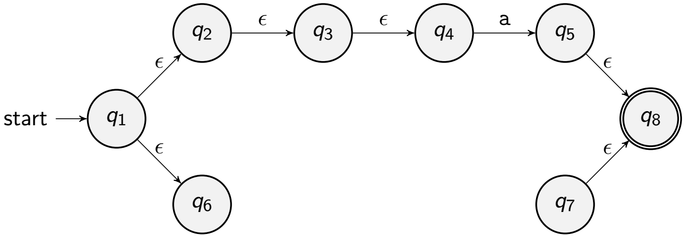


- $R = (R_1)$:



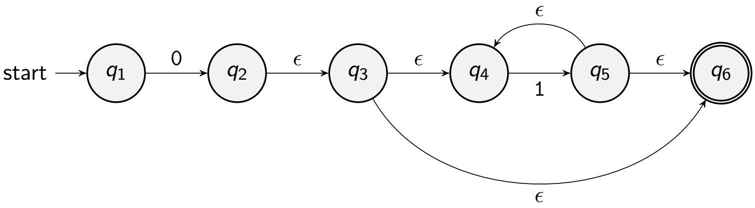
Regular Expressions to ϵ-NFA – Examples

- $R = \epsilon \cdot \mathtt{a} \,|\, \varnothing$



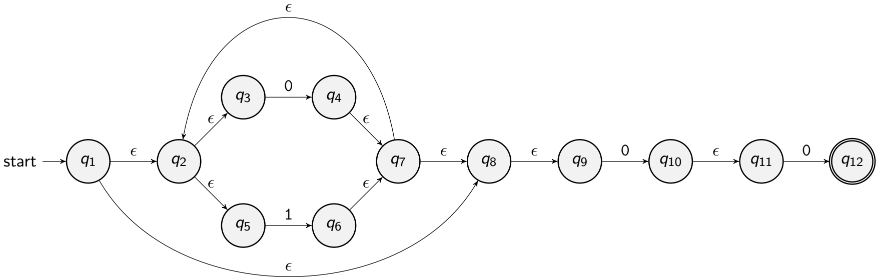
Regular Expressions to ϵ-NFA – Examples

- $R = 0 \cdot 1^*$



Regular Expressions to ϵ-NFA – Examples

- $R = (0|1)^* \cdot 0 \cdot 0$

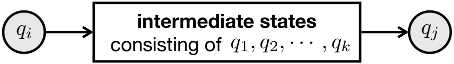


DFA to Regular Expressions

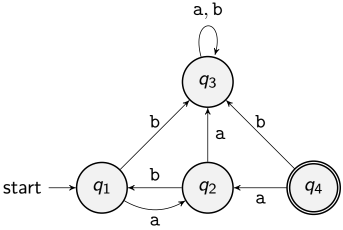
Theorem (DFA to Regular Expressions)

For a given DFA $D = (\{q_1, q_2, \cdots, q_n\}, \Sigma, \delta, q_1, F)$, $\exists$ *RE* R. $L(D) = L(R)$.

Let $R^{(k)}_{i,j}$ be the regular expression that accepts the paths from q_i to q_j whose *intermediate* states are $q_1, q_2, \cdots, q_k$. Then,



$$R = R^{(n)}_{1,f_1} \mid R^{(n)}_{1,f_2} \mid \cdots \mid R^{(n)}_{1,f_m} \quad \text{where } F = \{q_{f_1}, q_{f_2}, \cdots, q_{f_m}\}$$



$L(R^{(2)}_{1,3}) \ni$ `b`   `a a`   `a b a a`

$\not\ni$ `a`   `b a`   `a a b`

DFA to Regular Expressions

- **(Basis Case)** $k = 0$
  It means that **no intermediate states** in the path.

  - If $i \neq j$,

$$R_{i,j}^{(0)} = \mathtt{a}_1 \,|\, \mathtt{a}_2 \,|\, \cdots \,|\, \mathtt{a}_m$$

where $q_i \xrightarrow{\mathtt{a_1}} q_j, q_i \xrightarrow{\mathtt{a_2}} q_j, \cdots, q_i \xrightarrow{\mathtt{a}_m} q_j$ are transitions in D.

  - If $i = j$,

$$R_{i,j}^{(0)} = \epsilon \,|\, \mathtt{a}_1 \,|\, \mathtt{a}_2 \,|\, \cdots \,|\, \mathtt{a}_m$$

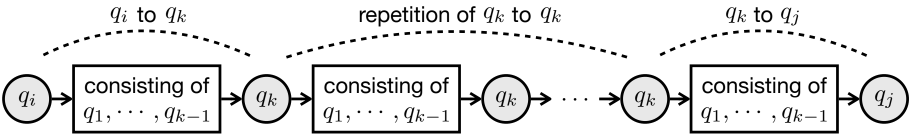
where $q_i \xrightarrow{\mathtt{a_1}} q_j, q_i \xrightarrow{\mathtt{a_2}} q_j, \cdots, q_i \xrightarrow{\mathtt{a}_m} q_j$ are transitions in D.

DFA to Regular Expressions

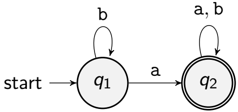
- **(Induction Case)** $R_{i,j}^{(k-1)}$ are given for all i and j.

$$R_{i,j}^{(k)} = R_{i,j}^{(k-1)} \mid R_{i,k}^{(k-1)}(R_{k,k}^{(k-1)})^* R_{k,j}^{(k-1)}$$

  - $R_{i,j}^{(k-1)}$: paths from q_i to q_j **NOT** containing q_k as intermediate states.
  - $R_{i,k}^{(k-1)}(R_{k,k}^{(k-1)})^* R_{k,j}^{(k-1)}$: paths from q_i to q_j containing q_k at least once as intermediate states.

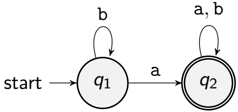


DFA to Regular Expressions – Examples



- $R_{1,1}^{(0)} =$
- $R_{1,2}^{(0)} =$
- $R_{2,1}^{(0)} =$
- $R_{2,2}^{(0)} =$

DFA to Regular Expressions – Examples

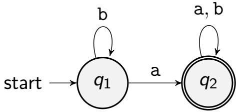


- $R^{(0)}_{1,1} = \epsilon \,|\, \mathtt{b}$
- $R^{(0)}_{1,2} = \mathtt{a}$
- $R^{(0)}_{2,1} = \varnothing$
- $R^{(0)}_{2,2} = \epsilon \,|\, \mathtt{a} \,|\, \mathtt{b}$

Note that $(\epsilon \,|\, R)^+ = R^*$, $(\epsilon \,|\, R)^* = R^*$, $\varnothing \cdot R = \varnothing$, $\varnothing \,|\, R = R$

- $R^{(1)}_{1,1} = R^{(0)}_{1,1} \mid R^{(0)}_{1,1}(R^{(0)}_{1,1})^* R^{(0)}_{1,1} = (R^{(0)}_{1,1})^+ = (\epsilon \,|\, \mathtt{b})^+ = \mathtt{b}^*$
- $R^{(1)}_{1,2} = R^{(0)}_{1,2} \mid R^{(0)}_{1,1}(R^{(0)}_{1,1})^* R^{(0)}_{1,2} = (R^{(0)}_{1,1})^* R^{(0)}_{1,2} = (\epsilon \,|\, \mathtt{b})^*\mathtt{a} = \mathtt{b}^*\mathtt{a}$
- $R^{(1)}_{2,1} = R^{(0)}_{2,1} \mid R^{(0)}_{2,1}(R^{(0)}_{1,1})^* R^{(0)}_{1,1} = R^{(0)}_{2,1}(R^{(0)}_{1,1})^* = \varnothing(\epsilon \,|\, \mathtt{b})^* = \varnothing$
- $R^{(1)}_{2,2} = R^{(0)}_{2,2} \mid R^{(0)}_{2,1}(R^{(0)}_{1,1})^* R^{(0)}_{1,2} = R^{(0)}_{2,2} \mid \varnothing = R^{(0)}_{2,2} = \epsilon \,|\, \mathtt{a} \,|\, \mathtt{b}$

DFA to Regular Expressions – Examples

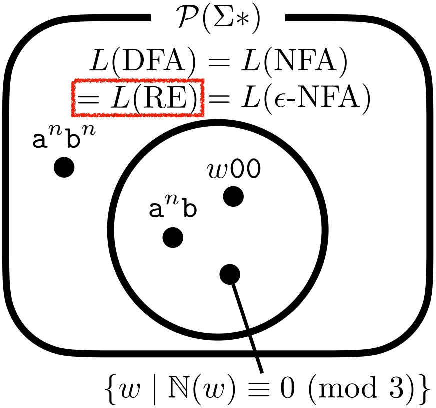


- $R_{1,1}^{(1)} = \mathtt{b}^*$
- $R_{1,2}^{(1)} = \mathtt{b}^*\mathtt{a}$
- $R_{2,1}^{(1)} = \varnothing$
- $R_{2,2}^{(1)} = \epsilon \,|\, \mathtt{a} \,|\, \mathtt{b}$

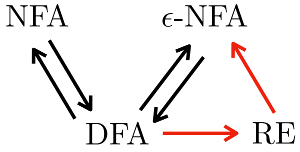
- $$\begin{aligned} R_{1,2}^{(2)} &= R_{1,2}^{(1)} \mid R_{1,2}^{(1)}(R_{2,2}^{(1)})^* R_{2,2}^{(1)} = R_{1,2}^{(1)}(R_{2,2}^{(1)})^* \\ &= \mathtt{b}^*\mathtt{a}(\epsilon \,|\, \mathtt{a} \,|\, \mathtt{b})^* \\ &= \mathtt{b}^*\mathtt{a}(\mathtt{a} \,|\, \mathtt{b})^* \end{aligned}$$

$R = R_{1,2}^{(2)} = \mathtt{b}^*\mathtt{a}(\mathtt{a} \,|\, \mathtt{b})^*$ is the regular expression for the above DFA.

Summary



$\mathcal{P}(\Sigma*)$
$L(\text{DFA}) = L(\text{NFA})$
$\boxed{= L(\text{RE})} = L(\epsilon\text{-NFA})$
$\mathtt{a}^n\mathtt{b}^n$
$w00$
$\mathtt{a}^n\mathtt{b}$
$\{w \mid \mathbb{N}(w) \equiv 0 \pmod 3\}$



NFA
ϵ-NFA
DFA
RE

Next Lecture

- Properties of Regular Languages

Jihyeok Park
jihyeok_park@korea.ac.kr
https://plrg.korea.ac.kr