Introduction to Alternating Finite Automata

Pascal Raiola

July 24th, 2013
Outline

1. Accepting with DFAs and NFAs
2. Generalization
3. Alternating finite automata (AFA)
4. Concatenation of two AFAs
Accepting with DFAs

Example: ababa

![DFA Diagram]

- Start state: $q_1$
- Transitions:
  - $a$ from $q_1$ to $q_2$
  - $b$ from $q_1$ to $b$
  - $a$ from $q_2$ to $q_1$
  - $b$ from $q_2$ to $b$

States:
- $q_1$
- $q_2$
Accepting with DFAs

Example: $q_1ababa$
Accepting with DFAs

Example: $aq_2baba$
Accepting with DFAs

Example: $abq_2aba$
Accepting with DFAs

Example: $abaq_1ba$
Accepting with DFAs

Example: $ababq_1a$
Accepting with DFAs

Example: $ababaq_2$
Accepting with DFAs

Example: \textit{ababa}q_2

\[ \text{⇒ Not accepted.} \]
Accepting with NFAs

Example: ababa

```

\begin{center}
\begin{tikzpicture}
  \node[state] (q1) at (0,0) {$q_1$};
  \node[state] (q2) at (1,0) {$q_2$};
  \path[->] (q1) edge node {a} (q2) edge[loop above] node {b} ();
  \path[->] (q2) edge[loop above] node {a} () edge node {a,b} (q1);
\end{tikzpicture}
\end{center}
```
Accepting with NFAs

Example: $\{q_1\}ababa$

Diagram:

- **$q_1$**
  - Transitions:
    - $a$ to $q_2$
    - $b$ to $q_1$
- **$q_2$**
  - Transitions:
    - $a, b$ to $q_2$

Initial state: $q_1$
Accepting with NFAs

Example: \(a\{q_2\}baba\)
Accepting with NFAs

Example: \( ab\{q_2\}aba \)
Accepting with NFAs

Example: \( aba\{q_1, q_2\}ba \)
Accepting with NFAs

Example: \(abab\{q_1, q_2\}a\)
Accepting with NFAs

Example: $ababa \{ q_1, q_2 \}$
Accepting with NFAs

Example: \textit{ababa} \{q_1, q_2\}

At least one accepting state \(\Rightarrow\) Accepted.
NFAs look more general than DFAs,
NFAs look more general than DFAs,
but accept the same class of languages.
NFAs look more general than DFAs,
but accept the same class of languages.

Can it be even more general?
Restrictions (NFA)

If we know:

\[ q_1 \xrightarrow{a} q_1 q_2 q_3 \]

and we know

\[ q_1 q_2 q_3 \xrightarrow{a} q_1 q_2 q_3 \]

Then

\[ q_1 q_2 q_3 \xrightarrow{a} q_1 q_2 q_3 \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \rightarrow q_2 \rightarrow q_3 \Rightarrow \text{Reading } a \Rightarrow \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

⇒ Reading \( a \) ⇒

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

and we know

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

Then

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \rightarrow q_2 \rightarrow q_3 \]

and we know:

\[ q_1 \rightarrow q_2 \rightarrow q_3 \]

Then

\[ q_1 \rightarrow q_2 \rightarrow q_3 \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

and we know

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

Then

\[ q_1 \xrightarrow{a} q_2 \xrightarrow{a} q_3 \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \xrightarrow{a} q_1 q_2 q_3 \]

and we know:

\[ q_1 q_2 q_3 \xrightarrow{a} q_1 q_2 q_3 \]

Then:

\[ q_1 q_2 q_3 \xrightarrow{a} q_1 q_2 q_3 \]

The transition can be more general!
Restrictions (NFA)

If we know:

\[ q_1 \quad q_2 \quad q_3 \)

⇒ Reading \( a \) ⇒

and we know

\[ q_1 \quad q_2 \quad q_3 \)

⇒ Reading \( a \) ⇒

Then

\[ q_1 \quad q_2 \quad q_3 \)

⇒ Reading \( a \) ⇒
Restrictions (NFA)

If we know:

\[ q_1 \rightarrow q_2 \rightarrow q_3 \Rightarrow \text{Reading } a \Rightarrow \]

and we know

\[ q_1 \rightarrow q_2 \rightarrow q_3 \Rightarrow \text{Reading } a \Rightarrow \]

Then

\[ q_1 \rightarrow q_2 \rightarrow q_3 \Rightarrow \text{Reading } a \Rightarrow \]
Restrictions (NFA)

If we know:

\[ q_1 \quad q_2 \quad q_3 \Rightarrow \text{Reading } a \Rightarrow \]

and we know

\[ q_1 \quad q_2 \quad q_3 \Rightarrow \text{Reading } a \Rightarrow \]

Then

\[ q_1 \quad q_2 \quad q_3 \Rightarrow \text{Reading } a \Rightarrow \]

The transition can be more general!
Acceptance condition

- DFAs accept iff the run ends in a final state.
Acceptance condition

- DFAs accept iff the run ends in a final state.
- NFAs accept iff the run ends in a set containing at least one final state.
Acceptance condition

- DFAs accept iff the run ends in a final state.
- NFAs accept iff the run ends in a set containing at least one final state.
- More general: A function $h$ deciding acceptance for each subset of $Q$:

$$h : 2^Q \rightarrow \{0, 1\}$$
Formal definition: $h$-AFA & $r$-AFA

An $h$-AFA/$r$-AFA is a 5-tuple $(Q, \Sigma, g, h, F)$, where

- $Q$ is the finite set of states,
- $\Sigma$ is the input alphabet,
Formal definition: $h$-AFA & $r$-AFA

An $h$-AFA/$r$-AFA is a 5-tuple $(Q, \Sigma, g, h, F)$, where

- $Q$ is the finite set of states,
- $\Sigma$ is the input alphabet,
- $g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}$ is the transition function,
Formal definition: *h*-AFA & *r*-AFA

An *h*-AFA/*r*-AFA is a 5-tuple \((Q, \Sigma, g, h, F)\), where
- \(Q\) is the finite set of states,
- \(\Sigma\) is the input alphabet,
- \(g: Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}\) is the transition function,
- \(h: 2^Q \rightarrow \{0, 1\}\) is the accepting function and
Formal definition: *h*-AFA & *r*-AFA

An *h*-AFA/*r*-AFA is a 5-tuple \((Q, \Sigma, g, h, F)\), where

- \(Q\) is the finite set of states,
- \(\Sigma\) is the input alphabet,
- \(g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}\) is the transition function,
- \(h : 2^Q \rightarrow \{0, 1\}\) is the accepting function and
- \(F \subseteq Q\) is the set of final states.
An $h$-AFA/$r$-AFA is a 5-tuple $(Q, \Sigma, g, h, F)$, where

- $Q$ is the finite set of states,
- $\Sigma$ is the input alphabet,
- $g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}$ is the transition function,
- $h : 2^Q \rightarrow \{0, 1\}$ is the accepting function and
- $F \subseteq Q$ is the set of final states.
- $f \in \{0, 1\}^Q$ is the to $F$ corresponding vector, e.g.
Formal definition: \( h \)-AFA & \( r \)-AFA

An \( h \)-AFA/\( r \)-AFA is a 5-tuple \((Q, \Sigma, g, h, F)\), where

- \( Q \) is the finite set of states,
- \( \Sigma \) is the input alphabet,
- \( g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\} \) is the transition function,
- \( h : 2^Q \rightarrow \{0, 1\} \) is the accepting function and
- \( F \subseteq Q \) is the set of final states.

\( f \in \{0, 1\}^Q \) is the to \( F \) corresponding vector, e.g.

\[
Q = \{q_1, q_2, q_3, q_4, q_5\}, F = \{q_2, q_3\} \\
\Rightarrow f = (0, 1, 1, 0, 0)
\]
Formal definition: $h$-AFA & $r$-AFA

The transition function $g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}$ is universalized from getting just one letter as an input to a whole word:
Formal definition: \( h\text{-AFA} \& \ r\text{-AFA} \)

- The transition function \( g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\} \) is universalized from getting just one letter as an input to a whole word:
  - \( g(q, \varepsilon, v) = v_q \)
Formal definition:  $h$-AFA & $r$-AFA

- The transition function $g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}$ is universalized from getting just one letter as an input to a whole word:
  - $g(q, \varepsilon, v) = v_q$
  - $g(q, aw, v) = g(q, a, g(w, v))$
Formal definition: $h$-AFA & $r$-AFA

- The transition function $g : Q \times \Sigma \times 2^Q \rightarrow \{0, 1\}$ is universalized from getting just one letter as an input to a whole word:
  - $g(q, \varepsilon, v) = v_q$
  - $g(q, aw, v) = g(q, a, g(w, v))$
  - Notation: $g(w, v) := (g(q, w, v))_{q \in Q}$. 
Acceptance

An input $w$ is accepted by an h-AFA iff

$$h(g(w, f)) = 1$$
Acceptance

An input $w$ is accepted by an h-AFA iff

$$h(g(w, f)) = 1$$

and by an r-AFA iff

$$h(g(w^R, f)) = 1.$$
Example: $r$-AFA

Let $A = (Q, \Sigma, g, h, F)$ be an $r$-AFA with

- $Q = \{q_1, q_2\}$,
- $\Sigma = \{a, b\}$,
- $F = \{q_2\}$, $f = (0, 1)$

$h(q_1, q_2) = q_1 \lor q_2$

and $g$ is given by

$g(a, (q_1, q_2)) = (q_1 \lor q_2, q_1 \land q_2)$

$g(b, (q_1, q_2)) = (q_1 \land q_2, q_1 \lor q_2)$
Example: $r$-AFA

Let $A = (Q, \Sigma, g, h, F)$ be an $r$-AFA with

- $Q = \{q_1, q_2\}$,
- $\Sigma = \{a, b\}$,
- $F = \{q_2\}, f = (0,1)$
- $h(q_1, q_2) = \overline{q_1} \lor q_2$
Example: $r$-AFA

Let $A = (Q, \Sigma, g, h, F)$ be an $r$-AFA with

- $Q = \{q_1, q_2\}$,
- $\Sigma = \{a, b\}$,
- $F = \{q_2\}$, $f = (0,1)$
- $h(q_1, q_2) = \overline{q_1} \lor q_2$
- and $g$ is given by

\[
\begin{align*}
g(a, (q_1, q_2)) &= (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \\
g(b, (q_1, q_2)) &= (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)
\end{align*}
\]
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

$$h(g(w^R, f))$$

$$g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2)$$

$$g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)$$

$$h(q_1, q_2) = \overline{q_1} \lor q_2, \quad f = (0, 1)$$
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

$$h(g(w^R, f)) = h(g(ba, f))$$

$$g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2)$$
$$g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)$$
$$h(q_1, q_2) = \overline{q_1} \lor q_2, \quad f = (0, 1)$$
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

$$h(g(w^R, f)) = h(g(ba, f)) = h(g(b, g(a, f)))$$

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ h(q_1, q_2) = \overline{q_1} \lor q_2, \quad f = (0, 1) \]
Example: \( r \)-AFA

Let \( w = ab \), then \( w \) is accepted by \( A \) as follows:

\[
h(g(w^R, f)) = h(g(ba, f)) \\
= h(g(b, g(a, f))) \\
= h(g(b, g(a, (0, 1))))
\]

\[
g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \\
g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \\
h(q_1, q_2) = \overline{q_1} \lor q_2, \quad f = (0, 1)
\]
Example: \( r\)-AFA

Let \( w = ab \), then \( w \) is accepted by \( A \) as follows:

\[
h(g(w^R, f)) = h(g(ba, f)) \\
= h(g(b, g(a, f))) \\
= h(g(b, g(a, (0, 1)))) \\
= h(g(b, (0 \lor \bar{1}, 0 \land \bar{1})))
\]

\[
g(a, (q_1, q_2)) = (q_1 \lor \bar{q}_2, \bar{q}_1 \land \bar{q}_2) \\
g(b, (q_1, q_2)) = (q_1 \land \bar{q}_2, \bar{q}_1 \lor q_2) \\
h(q_1, q_2) = \bar{q}_1 \lor q_2, \hspace{0.5cm} f = (0, 1)
\]
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

\[
h(g(w^R, f)) = h(g(ba, f))
= h(g(b, g(a, f)))
= h(g(b, g(a, (0, 1))))
= h(g(b, (0 \lor \overline{1}, \overline{0} \land \overline{1})))
= h(g(b, (0, 0)))
\]

\[
g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2})
\]
\[
g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)
\]
\[
h(q_1, q_2) = \overline{q_1} \lor q_2, \ f = (0, 1)
\]
Example: \textit{r-AFA}

Let $w = ab$, then $w$ is accepted by $A$ as follows:

$$h(g(w^R, f)) = h(g(ba, f))$$
$$= h(g(b, g(a, f)))$$
$$= h(g(b, g(a, (0, 1))))$$
$$= h(g(b, (0 \lor \overline{1}, \overline{0} \land \overline{1})))$$
$$= h(g(b, (0, 0)))$$
$$= h((0 \land \overline{0}, \overline{0} \lor \overline{0}))$$

$$g(a, (q_1, q_2)) = \left(q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}\right)$$
$$g(b, (q_1, q_2)) = \left(q_1 \land \overline{q_2}, \overline{q_1} \lor q_2\right)$$
$$h(q_1, q_2) = \overline{q_1} \lor q_2, \ f = (0, 1)$$
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

\[
\begin{align*}
  h(g(w^R, f)) &= h(g(ba, f)) \\
  &= h(g(b, g(a, f))) \\
  &= h(g(b, g(a, (0, 1)))) \\
  &= h(g(b, (0 \lor \overline{1}, \overline{0} \land \overline{1}))) \\
  &= h((0 \land \overline{0}, \overline{0} \lor 0)) \\
  &= h((0, 1))
\end{align*}
\]

\[
\begin{align*}
  g(a, (q_1, q_2)) &= (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2) \\
  g(b, (q_1, q_2)) &= (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \\
  h(q_1, q_2) &= \overline{q_1} \lor q_2, \quad f = (0, 1)
\end{align*}
\]
Example: \( r \)-AFA

Let \( w = ab \), then \( w \) is accepted by \( A \) as follows:

\[
\begin{align*}
    h(g(w^R, f)) &= h(g(ba, f)) \\
    &= h(g(b, g(a, f))) \\
    &= h(g(b, g(a, (0, 1)))) \\
    &= h(g(b, (0 \lor \bar{1}, \bar{0} \land \bar{1}))) \\
    &= h(g(b, (0, 0))) \\
    &= h((0 \land \bar{0}, \bar{0} \lor \overline{0})) \\
    &= h((0, 1)) \\
    &= \bar{0} \lor 1
\end{align*}
\]

\[
\begin{align*}
    g(a, (q_1, q_2)) &= (q_1 \lor \overline{q_2}, q_1 \land \overline{q_2}) \\
    g(b, (q_1, q_2)) &= (q_1 \land \overline{q_2}, q_1 \lor q_2) \\
    h(q_1, q_2) &= \overline{q_1} \lor q_2, \ f = (0, 1)
\end{align*}
\]
Example: $r$-AFA

Let $w = ab$, then $w$ is accepted by $A$ as follows:

\[
h(g(w^R, f)) = h(g(ba, f))
\]
\[
= h(g(b, g(a, f)))
\]
\[
= h(g(b, g(a, (0, 1))))
\]
\[
= h(g(b, (0 \lor 1, 0 \land 1)))
\]
\[
= h(g(b, (0, 0)))
\]
\[
= h((0 \land 0, 0 \lor 0))
\]
\[
= h((0, 1))
\]
\[
= \overline{0} \lor 1 = 1
\]

\[
g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2)
\]
\[
g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)
\]
\[
h(q_1, q_2) = \overline{q_1} \lor q_2, \quad f = (0, 1)
\]
Equivalence of DFAs and r-AFAs: “DFA ⇒ r-AFA”

Let \( A_D = (Q_D, \Sigma, \delta, s, F_D) \) be a DFA. Let \( A_A = (Q_A, \Sigma, g, h, F_A) \) be an r-AFA with:

- \( Q_A = Q_D \)
- \( F_A = \{ s \} \)
- \( g(q, a, v) = 1 \iff \exists p \in Q_D. v_p = 1 \land \delta(p, a) = q \)
- \( h(v) = 1 \iff \exists q \in F_D. v_q = 1 \)

Then \( L(A_D) = L(A_A) \).

Highly inefficient (see next talk)
Equivalence of DFAs and r-AFAs: “DFA $\Rightarrow$ r-AFA”

Let $A_D = (Q_D, \Sigma, \delta, s, F_D)$ be a DFA. Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA with:

- $Q_A = Q_D$
- $F_A = \{s\}$
Equivalence of DFAs and r-AFAs: “DFA ⇒ r-AFA”

Let $A_D = (Q_D, \Sigma, \delta, s, F_D)$ be a DFA. Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA with:

- $Q_A = Q_D$
- $F_A = \{s\}$
- $g(q, a, v) = 1 \iff \exists p \in Q_D. \; v_p = 1 \land \delta(p, a) = q$

Then $L(A_D) = L(A_A)$. Highly inefficient (see next talk)
Equivalence of DFAs and r-AFAs: “DFA ⇒ r-AFA”

Let $A_D = (Q_D, \Sigma, \delta, s, F_D)$ be a DFA. Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA with:

- $Q_A = Q_D$
- $F_A = \{s\}$
- $g(q, a, v) = 1 \iff \exists p \in Q_D. \; v_p = 1 \land \delta(p, a) = q$
- $h(v) = 1 \iff \exists q \in F_D. \; v_q = 1$

Then $L(A_D) = L(A_A)$. Highly inefficient (see next talk)
Equivalence of DFAs and r-AFAs: “DFA ⇒ r-AFA”

Let $A_D = (Q_D, \Sigma, \delta, s, F_D)$ be a DFA. Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA with:

- $Q_A = Q_D$
- $F_A = \{s\}$
- $g(q, a, v) = 1 \iff \exists p \in Q_D. v_p = 1 \land \delta(p, a) = q$
- $h(v) = 1 \iff \exists q \in F_D. v_q = 1$

Then $L(A_D) = L(A_A)$. Highly inefficient (see next talk)
Equivalence of DFAs and r-AFAs: “DFA $\Rightarrow$ r-AFA”

Let $A_D = (Q_D, \Sigma, \delta, s, F_D)$ be a DFA. Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA with:

- $Q_A = Q_D$
- $F_A = \{s\}$
- $g(q, a, v) = 1 \iff \exists p \in Q_D. v_p = 1 \land \delta(p, a) = q$
- $h(v) = 1 \iff \exists q \in F_D. v_q = 1$

Then $L(A_D) = L(A_A)$.

Highly inefficient (see next talk)
Equivalence of DFAs and r-AFAs: “r-AFA \Rightarrow DFA”:

Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA. The DFA $A_D = (Q_D, \Sigma, \delta, s, F_D)$ is defined as follows:
Equivalence of DFAs and r-AFAs: “r-AFA ⇔ DFA”:

Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA. The DFA $A_D = (Q_D, \Sigma, \delta, s, F_D)$ is defined as follows:

- $Q_D := \{0, 1\}^{Q_A}$. 

- $s := f_A$. 

- $g$ and $h$ as in the next slide.
Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA. The DFA $A_D = (Q_D, \Sigma, \delta, s, F_D)$ is defined as follows:

- $Q_D := \{0, 1\}^{Q_A}$.
- $s := f_A$. 
Equivalence of DFAs and r-AFAs: “r-AFA $\Rightarrow$ DFA”:

Let $A_A = (Q_A, \Sigma, g, h, F_A)$ be an r-AFA. The DFA $A_D = (Q_D, \Sigma, \delta, s, F_D)$ is defined as follows:

- $Q_D := \{0, 1\}^{Q_A}$.
- $s := f_A$.
- $g$ and $h$ as in the next slide.
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q}_2, \overline{q}_1 \land \overline{q}_2) \]

\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q}_2, \overline{q}_1 \lor q_2) \]

\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q}_1 \lor q_2 \]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ F = \{q_2\}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[
\begin{align*}
g(a, (q_1, q_2)) &= (q_1 \lor q_2, q_1 \land q_2) \\
g(b, (q_1, q_2)) &= (q_1 \land q_2, q_1 \lor q_2) \\
F &= \{q_2\}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2
\end{align*}
\]

\[
\begin{array}{c}
\text{\(q_1 = 0,\)} \quad \text{\(q_2 = 0\)} \\
\text{\(q_1 = 0,\)} \quad \text{\(q_2 = 1\)} \\
\text{\(q_1 = 1,\)} \quad \text{\(q_2 = 0\)} \\
\text{\(q_1 = 1,\)} \quad \text{\(q_2 = 1\)}
\end{array}
\]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ F = \{q_2\}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]

\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[
g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2})
\]

\[
g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)
\]

\[
F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2
\]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[
g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2})
\]

\[
g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2)
\]

\[
F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2
\]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]

\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land \overline{q_2}) \]
\[ g(b, (q_1, q_2)) = (q_1 \land \overline{q_2}, \overline{q_1} \lor q_2) \]
\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]
Example:

\[ g(a, (q_1, q_2)) = (q_1 \lor \overline{q_2}, \overline{q_1} \land q_2) \]
\[ g(b, (q_1, q_2)) = (q_1 \land q_2, \overline{q_1} \lor q_2) \]
\[ F = \{ q_2 \}, \quad h(q_1, q_2) = \overline{q_1} \lor q_2 \]

And so on...
Equivalence of DFAs and r-AFAs

r-AFAs $\sim$ regular languages
Equivalence of DFAs and r-AFAs

r-AFAs $\sim$ regular languages

Regular languages are closed under reversion
Equivalence of DFAs and r-AFAs

r-AFAs $\sim$ regular languages

Regular languages are closed under reversion

$\Rightarrow$ h-AFAs $\sim$ regular languages
Concatenation of two $r$-AFAs

Two $r$-AFAs:

$$A_1 = (Q_1, \Sigma, g_1, h_1, F_1), \quad A_2 = (Q_2, \Sigma, g_2, h_2, F_2)$$
Concatenation of two $r$-AFAs

Two $r$-AFAs:

$$A_1 = (Q_1, \Sigma, g_1, h_1, F_1), \quad A_2 = (Q_2, \Sigma, g_2, h_2, F_2)$$

Target: $r$-AFA $A = (Q, \Sigma, g, h, F)$ with $L(A) = L(A_1) \cdot L(A_2)$.
Concatenation of two r-AFAs: Idea

- Zacatecas
Concatenation of two r-AFAs: Idea

Z•acatecas
Concatenation of two r-AFAs: Idea

Zăcătecas
Concatenation of two r-AFAs: Idea

\[ A_1: \text{Za} \]
Concatenation of two r-AFAs: Idea

\[ A_1: Za \]
Concatenation of two r-AFAs: Idea

\[ A_1: Za \]
Concatenation of two r-AFAs: Idea

Zacat•ecas

cat•ecas

$A_1: \text{Za}$
Concatenation of two r-AFAs: Idea

$A_1: \text{Za}$

$A_1: \text{Zacat}$
Concatenation of two r-AFAs: Idea

\[ A_1: \text{Zacate} \circ \text{cas} \]

\[ A_1: \text{Zacat} \]
Concatenation of two r-AFAs: Idea

Zacate\(\bullet\)cas

cate\(\bullet\)cas

e\(\bullet\)cas

\(A_1: Za\)

\(A_1: Zacat\)

\(A_1: Zacate\)
Concatenation of two r-AFAs: Idea

Zacatec\textbullet as

catec\textbullet as

e\textbullet as

c\textbullet as

$A_1$: Za

$A_1$: Zacat

$A_1$: Zacate
Concatenation of two r-AFAs: Idea

$Zacateca_1$: Zacateca
$cateca_1$: Cateca
$eca_1$: Eca
$ca_1$: Ca

$A_1$: Za
$A_1$: Zacat
$A_1$: Zacate
Concatenation of two r-AFAs: Idea

Zacatecas

•

catecas

•

ecas

•

cas

$A_1: Z_a$

$A_1: Zacat$

$A_1: Zacate$
Concatenation of two r-AFAs: Definitions

Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$.
- $2m$ states to simulate each run in parallel - for each subset of $Q_2$ we store if there's a copy of $A_2$ in exactly these states.

Each subset $x \in 2^{Q_2}$ is associated to a state $p_x$. 
Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$. 
Concatenation of two r-AFAs: Definitions

Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$.
- $2^m$ states to simulate each run in parallel.
Concatenation of two r-AFAs: Definitions

Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$.
- $2^m$ states to simulate each run in parallel - for each subset of $Q_2$ we store if there's a copy of $A_2$ in exactly these states.
Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$.
- $2^m$ states to simulate each run in parallel - for each subset of $Q_2$ we store if there's a copy of $A_2$ in exactly these states.

$$Q = \{ q_0, \ldots, q_{n-1}, p_0, \ldots, p_{2^m-1} \}$$

- $n$ states
- $m$ states
Let $n := |Q_1|$ and $m := |Q_2|$, w.l.o.g $m \neq 0$ and $n \neq 0$. Then we need:

- $n$ states to simulate the one run of $A_1$.
- $2^m$ states to simulate each run in parallel - for each subset of $Q_2$ we store if there’s a copy of $A_2$ in exactly these states.

$$Q = \{q_0, \cdots, q_{n-1}, p_0, \cdots, p_{2^m-1}\}$$

Each subset $x \in 2^{Q_2}$ is associated to a state $p_x$. 
Concatenation of two r-AFAs: Where to start?

If $\epsilon \notin L(A_1)$ $A$ should be forced to start with $A_1$: $F = F_1$, 

Concatenation of two r-AFAs: Where to start?

If $\epsilon \notin L(A_1)$, $A$ should be forced to start with $A_1$: $F = F_1$, otherwise it can also directly launch a copy of $A_2$. 
If $\varepsilon \notin L(A_1)$ $A$ should be forced to start with $A_1$: $F = F_1$, otherwise it can also directly launch a copy of $A_2$, formally:

$$F = F_1 \cup \{p_{f_2}\}$$
Concatenation of two r-AFAs: Accepting

$h$ has only to care for $A_2$, formally:

$A_1$: Zacat
Concatenation of two r-AFAs: Accepting

$h$ has only to care for $A_2$, formally:

\[ h(v) = 1 \iff \exists x \in [0, 2^m - 1]. \ v_{n+x} = 1 \land h_2(x) = 1 \]
Concatenation of two r-AFAs: $g$ on the first $n$ states

$A$ has to run $A_1$ on the whole input word without any possibility of interruption:

$$g(a, v)|_{Q_1} = g_1(a, v|_{Q_1})$$
Concatenation of two r-AFAs: $g$ on the last $2^m$ states

Copies of $A_2$ should work parallel on the states $p_k$ ($k \geq 0$)
Concatenation of two r-AFAs: $g$ on the last $2^m$ states

Copies of $A_2$ should work parallel on the states $p_k$ ($k \geq 0$)

$\Rightarrow p_k$ should be reached from $p_j$

iff

formally:

For all $k \geq 0$,

$k \neq f_2 g(p_k, a, v) = 1 \iff \exists j \in [0, 2^m - 1]. v^{n+j} = 1 \land g_2(a, j) = k$
Concatenation of two r-AFAs: \( g \) on the last \( 2^m \) states

Copies of \( A_2 \) should work parallel on the states \( p_k \) \( (k \geq 0) \)

\[ \Rightarrow p_k \text{ should be reached from } p_j \]
\[ \text{iff} \]
\[ A_2 \text{ reaches } k \text{ from } j. \]
Concatenation of two r-AFAs: \( g \) on the last \( 2^m \) states

Copies of \( A_2 \) should work parallel on the states \( p_k \) \( (k \geq 0) \)

\[ \Rightarrow p_k \text{ should be reached from } p_j \]

iff

\[ A_2 \text{ reaches } k \text{ from } j. \]

formally:

For all \( k \geq 0, k \neq f_2 \)

\[ g(p_k, a, v) = 1 \iff \exists j \in [0, 2^m - 1]. \ v_{n+j} = 1 \land g_2(a, j) = k \]
Concatenation of two r-AFAs: Special treatment for $p_{f_2}$

The state $p_{f_2}$ can be reached:

$\exists j \in [0, 2m - 1]. v^n + j = 1 \land g_2(a, j) = f_2$
The state $p_{f_2}$ can be reached:

- as before and
Concatenation of two r-AFAs: Special treatment for $p_{f_2}$

The state $p_{f_2}$ can be reached:

- as before and
- if $A_1$ accepts a substring.
Concatenation of two r-AFAs: Special treatment for $p_{f_2}$

The state $p_{f_2}$ can be reached:

- as before and
- if $A_1$ accepts a substring.

Formally:

$$g(p_{f_2}, a, v) = 1 \iff (\exists j \in [0, 2^m - 1]. v_{n+j} = 1 \land g_2(a, j) = f_2)$$
$$\lor h_1(g(a, v)|_{Q_1}) = 1$$
Concatenation of two r-AFAs

Then \( L(A) = L(A_1) \cdot L(A_2) \).
Sources

Literature:


▶ Implementing Reversed Alternating Finite Automaton (r-AFA) Operations, S. Huerter, K. Salomaa, Xiuming Wu, S. Yu
Sources

Literature:
- Implementing Reversed Alternating Finite Automaton (r-AFA) Operations, S. Huerter, K. Salomaa, Xiuming Wu, S. Yu

Pictures:
- $A_1$: Larry D. Moore CC BY-SA 3.0.
- $A_2$: Disney/Pixar
Thank you!
Additional operations: Union and intersection

Acceptance-checking for AFAs allows working with multiple states in parallel.
Additional operations: Union and intersection

Acceptance-checking for AFAs allows working with multiple states in parallel.
⇒ For both the intersection and the union of two AFAs $A_1$ and $A_2$, one can run both AFAs in one AFA $A = (Q, \Sigma, g, h, F)$:
Additional operations: Union and intersection

Acceptance-checking for AFAs allows working with multiple states in parallel.
⇒ For both the intersection and the union of two AFAs $A_1$ and $A_2$, one can run both AFAs in one AFA $A = (Q, \Sigma, g, h, F)$:

- $Q = Q_1 \cup Q_2$
- $g(q, a, u) = \begin{cases} g_1(q, a, u | Q_1) & q \in Q_1 \\ g_2(q, a, u | Q_2) & q \in Q_2 \end{cases}$
Additional operations: Union and intersection

Acceptance-checking for AFAs allows working with multiple states in parallel.
⇒ For both the intersection and the union of two AFAs $A_1$ and $A_2$, one can run both AFAs in one AFA $A = (Q, \Sigma, g, h, F)$:

- $Q = Q_1 \cup Q_2$
- $g(q, a, u) = \begin{cases} 
  g_1(q, a, u|Q_1) & q \in Q_1 \\
  g_2(q, a, u|Q_2) & q \in Q_2 
\end{cases}$
- $h = h_1 \lor h_2$ (union) resp. $h = h_1 \land h_2$ (intersection).
- $F = F_1 \cup F_2$
For the complement $B$ of an AFA $A$, define $h_B = \overline{h_A}$. 