Advanced Logic

http://www-sop.inria.fr/members/Martin.Avanzini/teaching/2022/AL/

Martin Avanzini (martin.avanzini@inria.fr)
Etienne Lozes (etienne.lozes@unice.fr)

MASTER INFORMATIQUE
UNIVERSITÉ CÔTE D'AZUR

2nd Semester M1, 2022

Last Lecture

1. The class $REG(\Sigma)$ of regular languages is the *smallest* class (i.e., set of) languages s.t.
   1.1 $\emptyset \in REG(\Sigma)$ and $\{a\} \in REG(\Sigma)$ for every $a \in \Sigma$; and
   1.2 if $L, M \in REG(\Sigma)$ then $L \cup M \in REG(\Sigma)$, $L \cdot M \in REG(\Sigma)$ and $L^* \in REG(\Sigma)$.
2. Kleene's Theorem: The class of languages recognized by NFAs (DFAs) coincide with REG
3. finite automata yield decidable decision procedures

	Word	Emptyness	Universality	Inclusion	Equivalence
DFA	PTIME	PTIME	PTIME	PTIME	PTIME
NFA	PTIME	PTIME	PSPACE	PSPACE	PSPACE

   – state-space explosion through determinisation cannot be avoided

Today's Lecture

- non-determinism
- alternative finite automata
- relationship with regular languages

Non-Determinism

Angelican vs Demonic Non-Determinism

What is a non-deterministic machine (or automaton)?

- a machine which admits several executions on the same input
- put otherwise, during processing, several choices are possible

Angelican vs Demonic Non-Determinism

What is a non-deterministic machine (or automaton)?

- ⋆ a machine which admits several executions on the same input
- ⋆ put otherwise, during processing, several choices are possible
- ⋆ such choices can be resolved in favor (anglican non-determinism) or against (demonic non-determinism) a positive outcome (e.g. acceptance, termination, etc)

Angelican vs Demonic Non-Determinism

What is a non-deterministic machine (or automaton)?

- ⋆ a machine which admits several executions on the same input
- ⋆ put otherwise, during processing, several choices are possible
- ⋆ such choices can be resolved in favor (anglican non-determinism) or against (demonic non-determinism) a positive outcome (e.g. acceptance, termination, etc)
  - Anglican: an angel resolves choices

    $\Longrightarrow$ it is sufficient to have one "good" execution path, to have a positive outcome
  - Demonic: a demon resolves choices

    $\Longrightarrow$ all execution paths must be "good", to have a positive outcome

Angelican vs Demonic Non-Determinism

What is a non-deterministic machine (or automaton)?

- ⋆ a machine which admits several executions on the same input
- ⋆ put otherwise, during processing, several choices are possible
- ⋆ such choices can be resolved in favor (anglican non-determinism) or against (demonic non-determinism) a positive outcome (e.g. acceptance, termination, etc)
  - Anglican: an angel resolves choices

    ⟹ it is sufficient to have one "good" execution path, to have a positive outcome
  - Demonic: a demon resolves choices

    ⟹ all execution paths must be "good", to have a positive outcome

Example

- ⋆ NFAs are based on anglican non-determinism
- ⋆ worst-case complexity analysis assumes demonic non-determinism

NFAs with Demonic Choice

- NFAs incorporate angelic non-determinism because, in order for $w \in \mathsf{L}(\mathcal{A})$, only one accepting run of w has to exists

NFAs with Demonic Choice

- NFAs incorporate angelic non-determinism because, in order for $w \in \mathsf{L}(\mathcal{A})$, only one accepting run of w has to exists

- demonic non-determinism introduced by re-formulating the acceptance condition

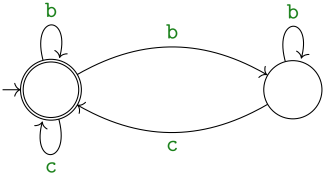
$$\mathsf{L}^{-}(\mathcal{A}) \triangleq \{w \mid \text{all runs on } w \text{ are accepting}\}$$

NFAs with Demonic Choice

- NFAs incorporate angelic non-determinism because, in order for $w \in L(\mathcal{A})$, only one accepting run of w has to exists
- demonic non-determinism introduced by re-formulating the acceptance condition

$$L^{-}(\mathcal{A}) \triangleq \{w \mid \text{all runs on } w \text{ are accepting}\}$$

Example



- $L(\mathcal{A}) = (b \cup c)^*$
- $L^{-}(\mathcal{A}) = \epsilon \cup (b \cup c)^* \cdot c$

Duality of Non-Determinism

- ⋆ recall that for each NFA $\mathcal{A}$, its dual $\overline{\mathcal{A}}$ is given by complementing final states
- ⋆ in general, only when $\mathcal{A}$ is deterministic, then $L(\overline{\mathcal{A}}) = \overline{L(\mathcal{A})}$

Duality of Non-Determinism

- ⋆ recall that for each NFA $\mathcal{A}$, its dual $\overline{\mathcal{A}}$ is given by complementing final states
- ⋆ in general, only when $\mathcal{A}$ is deterministic, then $\mathrm{L}(\overline{\mathcal{A}}) = \overline{\mathrm{L}(\mathcal{A})}$

Proposition

$$w \in \mathrm{L}(\mathcal{A}) \quad \Longleftrightarrow \quad w \notin \mathrm{L}^{-}(\overline{\mathcal{A}})$$

Duality of Non-Determinism

- recall that for each NFA $\mathcal{A}$, its dual $\overline{\mathcal{A}}$ is given by complementing final states
- in general, only when $\mathcal{A}$ is deterministic, then $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proposition

$$w \in \mathsf{L}(\mathcal{A}) \quad \iff \quad w \notin \mathsf{L}^{-}(\overline{\mathcal{A}})$$

- regime to resolve non-determinism has no effect on expressiveness of NFAs
- although potentially on the conciseness of the language description through NFAs

Duality of Non-Determinism

- ⋆ recall that for each NFA $\mathcal{A}$, its dual $\overline{\mathcal{A}}$ is given by complementing final states
- ⋆ in general, only when $\mathcal{A}$ is deterministic, then $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proposition

$$w \in \mathsf{L}(\mathcal{A}) \quad \Longleftrightarrow \quad w \notin \mathsf{L}^{-}(\overline{\mathcal{A}})$$

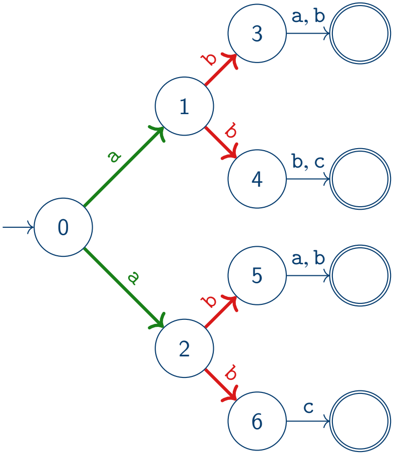
- ⋆ regime to resolve non-determinism has no effect on expressiveness of NFAs
- ⋆ although potentially on the conciseness of the language description through NFAs

what happens if we leave regime internal to the automata?

Alternating Finite Automata

Alternating Finite Automata

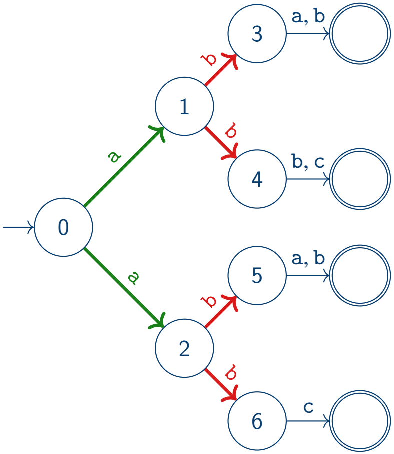
- General Idea: mix Anglican an Demonic choice on the level of individual transitions



$$\delta(0, \mathtt{a}) = 1 \vee 2$$
$$\delta(1, \mathtt{b}) = 3 \wedge 4$$
$$\delta(2, \mathtt{b}) = 5 \wedge 6$$
$$\vdots$$

Alternating Finite Automata

- ⋆ General Idea: mix Anglican an Demonic choice on the level of individual transitions



$$\delta(0,\mathrm{a}) = 1 \vee 2$$
$$\delta(1,\mathrm{b}) = 3 \wedge 4$$
$$\delta(2,\mathrm{b}) = 5 \wedge 6$$
$$\vdots$$

$$L(\mathcal{A}) = \mathrm{a}\left(\overbrace{\mathrm{b}(\overbrace{\mathrm{a} \cup \mathrm{b}}^{L(3)}) \cap \mathrm{b}(\overbrace{\mathrm{b} \cup \mathrm{c}}^{L(4)})}^{L(1)}\right)$$
$$\cup\ \mathrm{a}\left(\underbrace{\mathrm{b}(\underbrace{\mathrm{a} \cup \mathrm{b}}_{L(5)}) \cap \mathrm{b}\ \underbrace{\mathrm{c}}_{L(6)}}_{L(2)}\right)$$
$$= \mathrm{abb} \cup \varnothing$$
$$= \mathrm{abb}$$

Alternating Finite Automata, Formally

Positive Boolean Formulas

- let $A = \{a, b, \dots\}$ be a set of atoms
- the positive Boolean formulas $\mathbb{B}^+(A)$ over atoms A are given by the following grammar:

$$\phi, \psi ::= a \;\Big|\; \phi \wedge \psi \;\Big|\; \phi \vee \psi$$

  - such formulas are called positive because negation is missing

Alternating Finite Automata, Formally

Positive Boolean Formulas

- ⋆ let $A = \{a, b, \ldots\}$ be a set of atoms
- ⋆ the positive Boolean formulas $\mathbb{B}^+(A)$ over atoms A are given by the following grammar:

$$\phi, \psi ::= a \;\big|\; \phi \wedge \psi \;\big|\; \phi \vee \psi$$

  - such formulas are called positive because negation is missing

- ⋆ a set $M \subseteq A$ is a model of ϕ if $M \vDash \phi$ where

$$M \vDash a :\Longleftrightarrow a \in M \qquad M \vDash \phi \wedge \psi :\Longleftrightarrow M \vDash \phi \textbf{ and } M \vDash \psi \qquad M \vDash \phi \vee \psi :\Longleftrightarrow M \vDash \phi \textbf{ or } M \vDash \psi$$

Alternating Finite Automata, Formally

Positive Boolean Formulas

- ⋆ let $A = \{a, b, \dots\}$ be a set of atoms
- ⋆ the positive Boolean formulas $\mathbb{B}^+(A)$ over atoms A are given by the following grammar:

$$\phi, \psi ::= a \;\Big|\; \phi \wedge \psi \;\Big|\; \phi \vee \psi$$

  - such formulas are called positive because negation is missing

- ⋆ a set $M \subseteq A$ is a model of ϕ if $M \models \phi$ where

$$M \models a :\Longleftrightarrow a \in M \qquad M \models \phi \wedge \psi :\Longleftrightarrow M \models \phi \textbf{ and } M \models \psi \qquad M \models \phi \vee \psi :\Longleftrightarrow M \models \phi \textbf{ or } M \models \psi$$

Example

consider $\phi = a \wedge (b \vee c)$, then

$$\{a, b\} \models \phi \qquad \{a, c\} \models \phi \qquad \{a\} \not\models \phi \qquad \{b, c\} \not\models \phi$$

Alternating Finite Automata, Formally (II)

an alternating finite automata (AFA) is a tuple $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ where all components are identical to an NFA except that

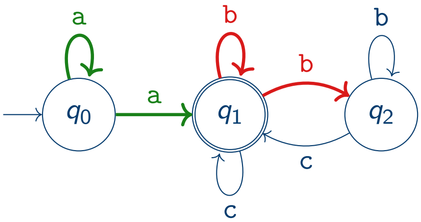
$$\delta : Q \times \Sigma \to \mathbb{B}^+(Q)$$

Alternating Finite Automata, Formally (II)

an alternating finite automata (AFA) is a tuple $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ where all components are identical to an NFA except that

$$\delta : Q \times \Sigma \to \mathbb{B}^+(Q)$$

Example



δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$

Runs in an AFA

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ be an AFA

- an execution for a word $w = \mathtt{a}_1 \ldots \mathtt{a}_n \in \Sigma^*$ is a tree T_w whose nodes are labeled by states Q s.t.:
  1. the root node of T_w is labeled by the initial state q_I
  2. for all nodes v on the ith layer ($i = 0, \ldots, n-1$) with successors $v_1, \ldots, v_k$ on layer $i+1$, labeled by $q_1, \ldots, q_k$, respectively:

$$\{q_1, \ldots, q_k\} \models \delta(q, \mathtt{a}_{i+1})$$

Runs in an AFA

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ be an AFA

- ★ an execution for a word $w = \mathtt{a}_1 \ldots \mathtt{a}_n \in \Sigma^*$ is a tree T_w whose nodes are labeled by states Q s.t.:
  1. the root node of T_w is labeled by the initial state q_I
  2. for all nodes v on the ith layer ($i = 0, \ldots, n-1$) with successors $v_1, \ldots, v_k$ on layer $i+1$, labeled by $q_1, \ldots, q_k$, respectively:

$$\{q_1, \ldots, q_k\} \models \delta(q, \mathtt{a}_{i+1})$$

- ★ an execution is accepting if all leafs are labeled by final states

Runs in an AFA

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$ be an AFA

- ★ an execution for a word $w = \mathtt{a}_1 \dots \mathtt{a}_n \in \Sigma^*$ is a tree T_w whose nodes are labeled by states Q s.t.:
  1. the root node of T_w is labeled by the initial state q_I
  2. for all nodes v on the ith layer ($i = 0, \dots, n-1$) with successors $v_1, \dots, v_k$ on layer $i+1$, labeled by $q_1, \dots, q_k$, respectively:

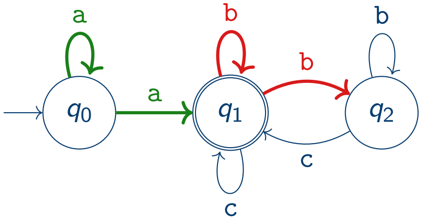
$$\{q_1, \dots, q_k\} \models \delta(q, \mathtt{a}_{\mathtt{i+1}})$$

- ★ an execution is accepting if all leafs are labeled by final states

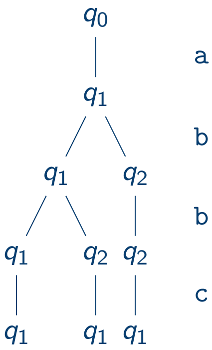
- ★ the language recognized by $\mathcal{A}$ is given by

$$\mathrm{L}(\mathcal{A}) \triangleq \{w \mid \text{there exists an accepting execution } T_w \text{ for } w\}$$

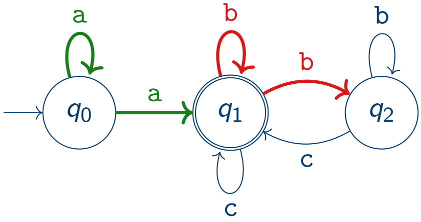
Example of Accepting Execution for $w = \texttt{abbc}$



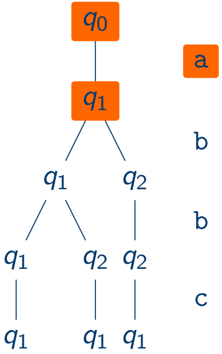
δ	a	b	c
q_0	$q_0 \lor q_1$	$q_\bot$	$q_\bot$
q_1	$q_\bot$	$q_1 \land q_2$	q_1
q_2	$q_\bot$	q_2	q_1
$q_\bot$	$q_\bot$	$q_\bot$	$q_\bot$



Example of Accepting Execution for $w = \texttt{abbc}$

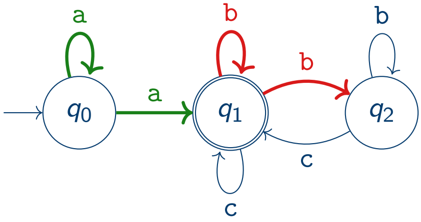


δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$

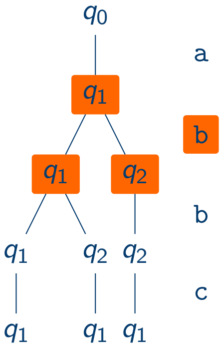


$$\{q_1\} \models q_0 \vee q_1$$

Example of Accepting Execution for $w = \texttt{abbc}$

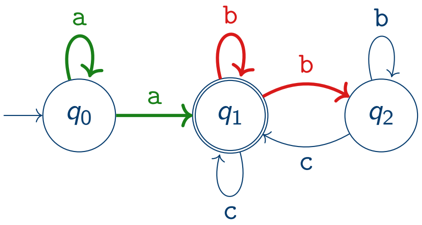


δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$

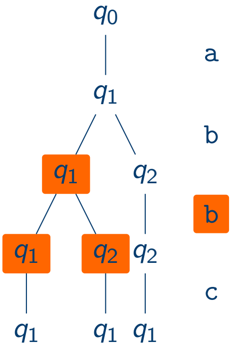


$$\{q_1, q_2\} \models q_1 \wedge q_2$$

Example of Accepting Execution for $w = \texttt{abbc}$

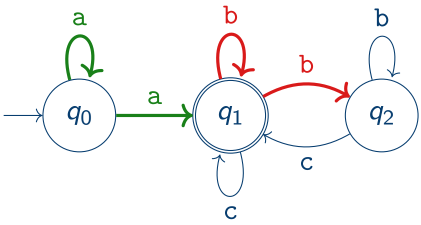


δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$

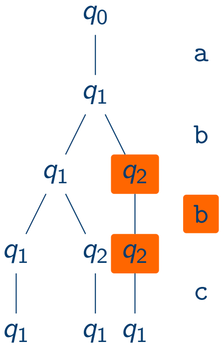


$$\{q_1, q_2\} \models q_1 \wedge q_2$$

Example of Accepting Execution for w = abbc

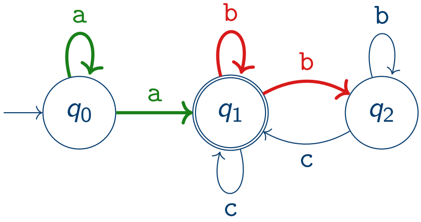


δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$

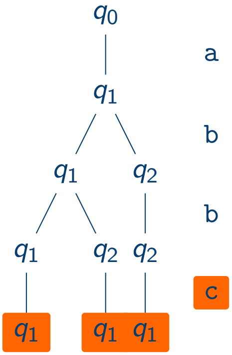


$\{q_2\} \models q_2$

Example of Accepting Execution for $w = \mathtt{abbc}$

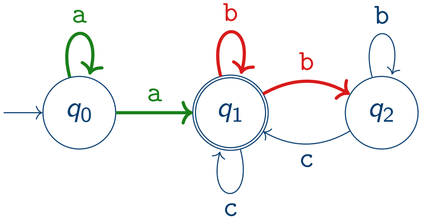


δ	a	b	c
q_0	$q_0 \lor q_1$	$q_\bot$	$q_\bot$
q_1	$q_\bot$	$q_1 \land q_2$	q_1
q_2	$q_\bot$	q_2	q_1
$q_\bot$	$q_\bot$	$q_\bot$	$q_\bot$

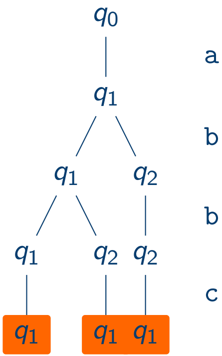


$\{q_1\} \models q_1$

Example of Accepting Execution for $w = \texttt{abbc}$



δ	a	b	c
q_0	$q_0 \vee q_1$	$q_\perp$	$q_\perp$
q_1	$q_\perp$	$q_1 \wedge q_2$	q_1
q_2	$q_\perp$	q_2	q_1
$q_\perp$	$q_\perp$	$q_\perp$	$q_\perp$



$\{q_1, q_1, q_1\} \subseteq F$

Extended Transition Function

the extended transition function

$$\hat{\delta} : \mathbb{B}^+(Q) \times \Sigma^* \to \mathbb{B}^+(Q)$$

is recursively defined by:

$$\hat{\delta}(q, \epsilon) \triangleq q \qquad \hat{\delta}(\phi \vee \psi, w) = \hat{\delta}(\phi, w) \vee \hat{\delta}(\psi, w)$$

$$\hat{\delta}(q, \mathrm{a} \cdot w) \triangleq \hat{\delta}(\delta(q, \mathrm{a}), w) \qquad \hat{\delta}(\phi \wedge \psi, w) = \hat{\delta}(\phi, w) \wedge \hat{\delta}(\psi, w)$$

Extended Transition Function

the extended transition function

$$\hat{\delta} : \mathbb{B}^+(Q) \times \Sigma^* \rightarrow \mathbb{B}^+(Q)$$

is recursively defined by:

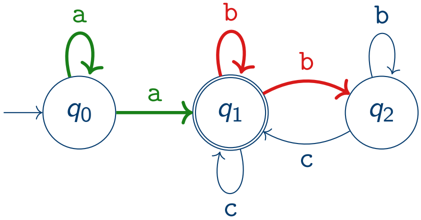
$$\hat{\delta}(q, \epsilon) \triangleq q \qquad \hat{\delta}(\phi \vee \psi, w) = \hat{\delta}(\phi, w) \vee \hat{\delta}(\psi, w)$$

$$\hat{\delta}(q, \mathrm{a} \cdot w) \triangleq \hat{\delta}(\delta(q, \mathrm{a}), w) \qquad \hat{\delta}(\phi \wedge \psi, w) = \hat{\delta}(\phi, w) \wedge \hat{\delta}(\psi, w)$$

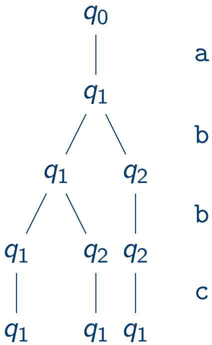
Lemma

$$\mathrm{L}(\mathcal{A}) = \{w \mid F \vDash \hat{\delta}(q_I, w)\}$$

Example of Accepting Execution for $w = \texttt{abbc}$ (II)



δ	a	b	c
q_0	$q_0 \lor q_1$	$q_\bot$	$q_\bot$
q_1	$q_\bot$	$q_1 \land q_2$	q_1
q_2	$q_\bot$	q_2	q_1
$q_\bot$	$q_\bot$	$q_\bot$	$q_\bot$



$$
\begin{aligned}
\hat{\delta}(q_0, \texttt{abbc}) &= \hat{\delta}(q_0 \lor q_1, \texttt{bbc}) \\
&= \hat{\delta}(q_0, \texttt{bbc}) \lor \hat{\delta}(q_1, \texttt{bbc}) \\
&= \hat{\delta}(q_\bot, \texttt{bc}) \lor (\hat{\delta}(q_1, \texttt{bc}) \land \hat{\delta}(q_2, \texttt{bc})) \\
&= \hat{\delta}(q_\bot, \texttt{c}) \lor (\hat{\delta}(q_1, \texttt{c}) \land \hat{\delta}(q_2, \texttt{c})) \\
&= \hat{\delta}(q_\bot, \epsilon) \lor \hat{\delta}(q_1, \epsilon) \\
&= q_\bot \lor q_1
\end{aligned}
$$

$$\{q_1\} \models q_\bot \lor q_1$$

Comparison to NFAs and DFAs

- AFAs generalise NFAs
  - every DFA is a NFA is an AFA

Comparison to NFAs and DFAs

- AFAs generalise NFAs
  - every DFA is a NFA is an AFA
- AFAs allow often more succinct encoding / automata constructions

Comparison to NFAs and DFAs

- AFAs generalise NFAs
  - every DFA is a NFA is an AFA
- AFAs allow often more succinct encoding / automata constructions

Example

- let $\mathcal{A}^{(m)} = (Q^{(m)}, \{\mathrm{a}\}, \delta^{(m)}, q_I^{(m)}, F^{(m)})$ be an NFA with $\mathrm{L}(\mathcal{A}^{(m)}) = \{w \mid |w| = 0 \bmod m\}$
  - this NFA has at least m states

Comparison to NFAs and DFAs

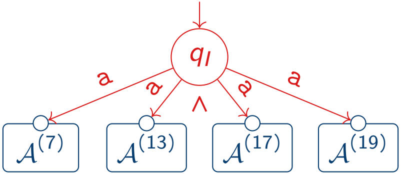
- AFAs generalise NFAs
  - every DFA is a NFA is an AFA

- AFAs allow often more succinct encoding / automata constructions

Example

- let $\mathcal{A}^{(m)} = (Q^{(m)}, \{a\}, \delta^{(m)}, q_I^{(m)}, F^{(m)})$ be an NFA with $L(\mathcal{A}^{(m)}) = \{w \mid |w| = 0 \bmod m\}$
  - this NFA has at least m states

- consider the AFA $\mathcal{A}$ defined from $\mathcal{A}^{(m)}$ for primes $m = 7, 13, 17, 19$ by



Comparison to NFAs and DFAs

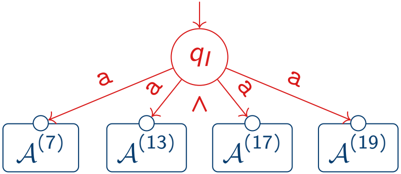
- AFAs generalise NFAs
  - every DFA is a NFA is an AFA

- AFAs allow often more succinct encoding / automata constructions

Example

- let $\mathcal{A}^{(m)} = (Q^{(m)}, \{a\}, \delta^{(m)}, q_I^{(m)}, F^{(m)})$ be an NFA with $L(\mathcal{A}^{(m)}) = \{w \mid |w| = 0 \bmod m\}$
  - this NFA has at least m states

- consider the AFA $\mathcal{A}$ defined from $\mathcal{A}^{(m)}$ for primes $m = 7, 13, 17, 19$ by



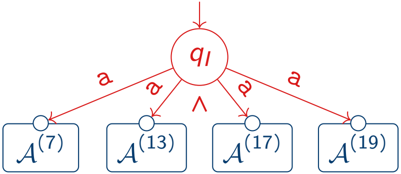
  - $L(\mathcal{A}) = \{w \mid |w| = 1 \bmod 29393\}$ since $29393 = 7 \cdot 13 \cdot 17 \cdot 19$

Comparison to NFAs and DFAs

- AFAs generalise NFAs
  - every DFA is a NFA is an AFA
- AFAs allow often more succinct encoding / automata constructions

Example

- let $\mathcal{A}^{(m)} = (Q^{(m)}, \{a\}, \delta^{(m)}, q_I^{(m)}, F^{(m)})$ be an NFA with $L(\mathcal{A}^{(m)}) = \{w \mid |w| = 0 \bmod m\}$
  - this NFA has at least m states
- consider the AFA $\mathcal{A}$ defined from $\mathcal{A}^{(m)}$ for primes $m = 7, 13, 17, 19$ by



  - $L(\mathcal{A}) = \{w \mid |w| = 1 \bmod 29393\}$ since $29393 = 7 \cdot 13 \cdot 17 \cdot 19$
  - AFA $\mathcal{A}$ has $57 = 1 + 7 + 13 + 17 + 19$, whereas a corresponding NFA needs 29393 states

Complementation

- recall: NFA-complementation may blow-up automata sizes by an exponential

Lemma

For every AFA $\mathcal{A}$ there exists an AFA $\overline{\mathcal{A}}$ of equal size such that $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Complementation

- recall: NFA-complementation may blow-up automata sizes by an exponential

Lemma

For every AFA $\mathcal{A}$ there exists an AFA $\overline{\mathcal{A}}$ of equal size such that $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proof Outline.

- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- define the dual formula $\overline{\phi}$ of $\phi \in \mathbb{B}^+(Q)$ following De Morgans rules

$$\overline{q} \triangleq q \qquad \overline{\phi \vee \psi} \triangleq \overline{\phi} \wedge \overline{\psi} \qquad \overline{\phi \wedge \psi} \triangleq \overline{\phi} \vee \overline{\psi}$$

  - morally, $q \in Q$ re-used for their "negation"; we have (i) $M \models \phi$ iff $Q \setminus M \not\models \overline{\phi}$

Complementation

- recall: NFA-complementation may blow-up automata sizes by an exponential

Lemma

For every AFA $\mathcal{A}$ there exists an AFA $\overline{\mathcal{A}}$ of equal size such that $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proof Outline.

- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- define the dual formula $\overline{\phi}$ of $\phi \in \mathbb{B}^+(Q)$ following De Morgans rules

$$\overline{q} \triangleq q \qquad \overline{\phi \vee \psi} \triangleq \overline{\phi} \wedge \overline{\psi} \qquad \overline{\phi \wedge \psi} \triangleq \overline{\phi} \vee \overline{\psi}$$

  - morally, $q \in Q$ re-used for their “negation”; we have (i) $M \vDash \phi$ iff $Q \backslash M \nvDash \overline{\phi}$
- we now define $\overline{\mathcal{A}} \triangleq (Q, \Sigma, \overline{\delta}, q_I, Q \backslash F)$ where $\overline{\delta}(q, \mathrm{a}) \triangleq \overline{\delta(q, \mathrm{a})}$ for all $q \in Q$, $\mathrm{a} \in \Sigma$

Complementation

- recall: NFA-complementation may blow-up automata sizes by an exponential

Lemma

For every AFA $\mathcal{A}$ there exists an AFA $\overline{\mathcal{A}}$ of equal size such that $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proof Outline.

- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- define the dual formula $\overline{\phi}$ of $\phi \in \mathbb{B}^+(Q)$ following De Morgans rules

$$\overline{q} \triangleq q \qquad \overline{\phi \vee \psi} \triangleq \overline{\phi} \wedge \overline{\psi} \qquad \overline{\phi \wedge \psi} \triangleq \overline{\phi} \vee \overline{\psi}$$

  - morally, $q \in Q$ re-used for their "negation"; we have (i) $M \vDash \phi$ iff $Q \backslash M \nvDash \overline{\phi}$
- we now define $\overline{\mathcal{A}} \triangleq (Q, \Sigma, \overline{\delta}, q_I, Q \backslash F)$ where $\overline{\delta}(q, \mathrm{a}) \triangleq \overline{\delta(q, \mathrm{a})}$ for all $q \in Q$, $\mathrm{a} \in \Sigma$
  - by induction on $|w|$ it can now be shown that (ii) $\hat{\overline{\delta}}(q_I, w) = \overline{\hat{\delta}(q, w)}$

Complementation

- recall: NFA-complementation may blow-up automata sizes by an exponential

Lemma

For every AFA $\mathcal{A}$ there exists an AFA $\overline{\mathcal{A}}$ of equal size such that $\mathsf{L}(\overline{\mathcal{A}}) = \overline{\mathsf{L}(\mathcal{A})}$

Proof Outline.

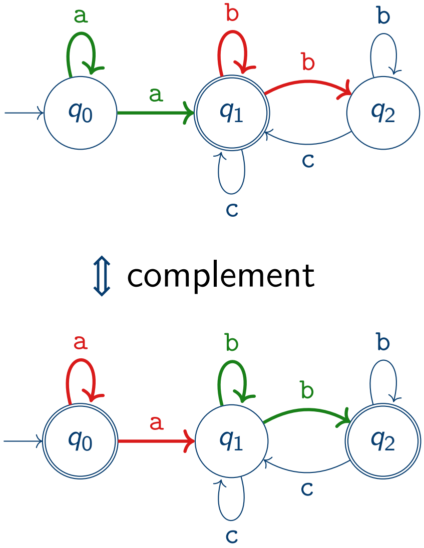
- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- define the dual formula $\overline{\phi}$ of $\phi \in \mathbb{B}^+(Q)$ following De Morgans rules

$$\overline{q} \triangleq q \qquad \overline{\phi \vee \psi} \triangleq \overline{\phi} \wedge \overline{\psi} \qquad \overline{\phi \wedge \psi} \triangleq \overline{\phi} \vee \overline{\psi}$$

  - morally, $q \in Q$ re-used for their "negation"; we have (i) $M \vDash \phi$ iff $Q \backslash M \nvDash \overline{\phi}$
- we now define $\overline{\mathcal{A}} \triangleq (Q, \Sigma, \overline{\delta}, q_I, Q \backslash F)$ where $\overline{\delta}(q, \mathsf{a}) \triangleq \overline{\delta(q, \mathsf{a})}$ for all $q \in Q$, $\mathsf{a} \in \Sigma$
  - by induction on $|w|$ it can now be shown that (ii) $\hat{\overline{\delta}}(q_I, w) = \overline{\hat{\delta}(q, w)}$
  - overall, we have

$$w \notin \mathsf{L}(\mathcal{A}) \overset{def.}{\iff} F \nvDash \hat{\delta}(q_I, w) \overset{(i)}{\iff} Q \backslash F \vDash \overline{\hat{\delta}(q_I, w)} \overset{(ii)}{\iff} Q \backslash F \vDash \hat{\overline{\delta}}(q_I, w) \overset{def.}{\iff} w \in \mathsf{L}(\overline{\mathcal{A}})$$

Example



Relationship with Regular Languages

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ *there exist a DFA* $\mathcal{B}$ *with* $O(2^{2^{|\mathcal{A}|}})$ *states such that* $L(\mathcal{A}) = L(\mathcal{B})$.

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ *there exist a DFA* $\mathcal{B}$ *with* $O(2^{2^{|\mathcal{A}|}})$ *states such that* $L(\mathcal{A}) = L(\mathcal{B})$.

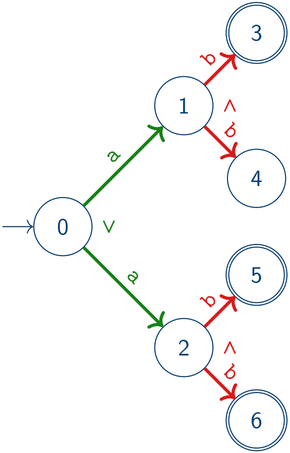
Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

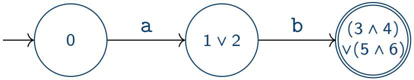
Idea:

- the states of $\mathcal{B}$ are formulas
- $\phi \xrightarrow{\mathtt{a}} \psi$ in $\mathcal{B}$ if $\hat{\delta}(\phi, \mathtt{a}) = \psi$
  - Example: $\delta(p, \mathtt{a}) = q \wedge r$ and $\delta(q, \mathtt{a}) = r \implies p \vee q \xrightarrow{\mathtt{a}} (q \wedge r) \vee r$
  - a run $q_I \xrightarrow{\mathtt{a}_1} \cdots \xrightarrow{\mathtt{a}_n} \phi$ thus models $\hat{\delta}(q_I, \mathtt{a}_1 \ldots \mathtt{a}_n) = \phi$
- the formula q_I is the initial state
- the formulas modeled by F are final

Example



the initial AFA



the translated DFA

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ there exist a DFA $\mathcal{B}$ with $\mathrm{O}(2^{2^{|\mathcal{A}|}})$ states such that $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$.

Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

Idea:

- ⋆ the states of $\mathcal{B}$ are formulas
- ⋆ $\phi \xrightarrow{\mathrm{a}} \psi$ in $\mathcal{B}$ if $\hat{\delta}(\phi, \mathrm{a}) = \psi$
  - Example: $\delta(p, \mathrm{a}) = q \wedge r$ and $\delta(q, \mathrm{a}) = r \implies p \vee q \xrightarrow{\mathrm{a}} (q \wedge r) \vee r$
  - a run $q_I \xrightarrow{\mathrm{a}_1} \cdots \xrightarrow{\mathrm{a}_\mathrm{n}} \phi$ thus models $\hat{\delta}(q_I, \mathrm{a}_1 \ldots \mathrm{a}_\mathrm{n}) = \phi$
- ⋆ the formula q_I is the initial state
- ⋆ the formulas modeled by F are final
- ⋆ to keep the construction finite, we'll identify equivalent formulas

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ *there exist a DFA* $\mathcal{B}$ *with* $\mathrm{O}(2^{2^{|\mathcal{A}|}})$ *states such that* $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$*.*

Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

Formally:

- the equivalence $\sim$ on $\mathbb{B}^+(Q)$ is given by $\phi \sim \psi$ if $\{M \mid M \models \phi\} = \{M \mid M \models \psi\}$
  - $q \sim q \vee q \sim q \wedge q$ but $q \not\sim p \vee q \not\sim p \wedge q$

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ there exist a DFA $\mathcal{B}$ with $\mathrm{O}(2^{2^{|\mathcal{A}|}})$ states such that $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$.

Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

Formally:

- the equivalence $\sim$ on $\mathbb{B}^+(Q)$ is given by $\phi \sim \psi$ if $\{M \mid M \models \phi\} = \{M \mid M \models \psi\}$
  - $q \sim q \vee q \sim q \wedge q$ but $q \not\sim p \vee q \not\sim p \wedge q$
- the equivalence class $[\phi]_\sim$ can be simply conceived as the formula ϕ, with equivalent formulas $\phi \sim \psi$ identified
  - $[q \vee q]_\sim = \{q, q \vee q, q \wedge q, \ldots\}$

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ *there exist a DFA* $\mathcal{B}$ *with* $O(2^{2^{|\mathcal{A}|}})$ *states such that* $L(\mathcal{A}) = L(\mathcal{B})$.

Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

Formally:

- ⋆ the equivalence $\sim$ on $\mathbb{B}^+(Q)$ is given by $\phi \sim \psi$ if $\{M \mid M \models \phi\} = \{M \mid M \models \psi\}$
  - $q \sim q \vee q \sim q \wedge q$ but $q \not\sim p \vee q \not\sim p \wedge q$
- ⋆ the equivalence class $[\phi]_\sim$ can be simply conceived as the formula ϕ, with equivalent formulas $\phi \sim \psi$ identified
  - $[q \vee q]_\sim = \{q, q \vee q, q \wedge q, \dots\}$
- ⋆ the set of all such equivalence classes $\mathbb{B}^+(Q)/\sim$ contains $O(2^{2^{|Q|}})$ elements

AFAs Recognize *REG*

Theorem

For every AFA $\mathcal{A}$ *there exist a DFA* $\mathcal{B}$ *with* $\mathrm{O}(2^{2^{|\mathcal{A}|}})$ *states such that* $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$.

Proof Outline.

let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$

Formally:

- the equivalence $\sim$ on $\mathbb{B}^+(Q)$ is given by $\phi \sim \psi$ if $\{M \mid M \models \phi\} = \{M \mid M \models \psi\}$
  - $q \sim q \vee q \sim q \wedge q$ but $q \not\sim p \vee q \not\sim p \wedge q$
- the equivalence class $[\phi]_\sim$ can be simply conceived as the formula ϕ, with equivalent formulas $\phi \sim \psi$ identified
  - $[q \vee q]_\sim = \{q, q \vee q, q \wedge q, \ldots\}$
- the set of all such equivalence classes $\mathbb{B}^+(Q)/\sim$ contains $\mathrm{O}(2^{2^{|Q|}})$ elements
- $\mathcal{B} \triangleq (\mathbb{B}^+(Q)/\sim, \Sigma, q_I, \delta_\sim, \{[\phi]_\sim \mid F \models \phi\})$ where $\delta_\sim([\phi]_\sim, \mathtt{a}) \triangleq [\hat{\delta}(\phi, \mathtt{a})]_\sim$ recognises $\mathrm{L}(\mathcal{A})$

From AFAs to NFA

Theorem

For every AFA $\mathcal{A}$ *there exist a NFA* $\mathcal{B}$ *with* $\mathrm{O}(2^{|\mathcal{A}|})$ *states such that* $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$.

Proof Outline.

- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- idea: rather then "recording" to be validated formulas as in the DFA construction, the corresponding NFA "records" valuations
  - the construction is simpler, at the expense of non-determinism

From AFAs to NFA

Theorem

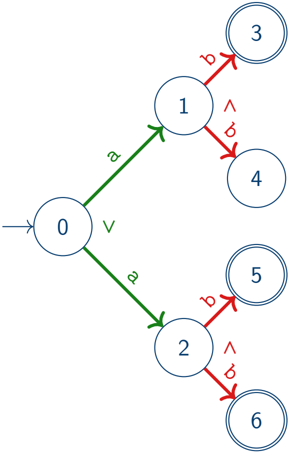
For every AFA $\mathcal{A}$ *there exist a NFA* $\mathcal{B}$ *with* $\mathrm{O}(2^{|\mathcal{A}|})$ *states such that* $\mathrm{L}(\mathcal{A}) = \mathrm{L}(\mathcal{B})$.

Proof Outline.

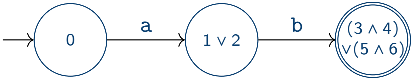
- let $\mathcal{A} = (Q, \Sigma, q_I, \delta, F)$
- idea: rather then "recording" to be validated formulas as in the DFA construction, the corresponding NFA "records" valuations
  - the construction is simpler, at the expense of non-determinism
- the NFA is given by $\mathcal{B} \triangleq (2^Q, \Sigma, \{q_I\}, \delta', \{M \mid M \subseteq F\})$ where

$$N \in \delta'(M, \mathrm{a}) \quad :\Longleftrightarrow \quad N \vDash \bigwedge_{q \in M} \delta(q, \mathrm{a})$$

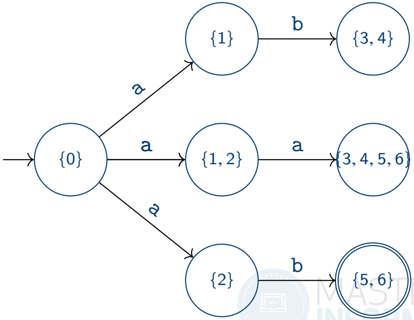
Example (II)



the initial AFA



the translated DFA



the translated NFA