Determinism: computation always continues in a uniquely determined way.

Nondeterminism: have more (or none) choices

Example:

\[ \{ \ w \in \{0,1\}^* \mid w \text{ contains 001 or 0101 as a substring} \} = A \cup B \]

\[ A = \{ \ w \in \{0,1\}^* \mid w \text{ contains 001 as a substring} \} \]

\[ B = \{ \ w \in \{0,1\}^* \mid w \text{ contains 0101 as a substring} \} \]

Diagram:

- Green: NFA for \( A \)
- Blue: NFA for \( B \)
- Green + Blue: NFA for \( A \cup B \) (many other NFA's)
Determinism: computation always continues in a uniquely determined way.

Nondeterminism: have more (or none) choices

Example:

\[ \{ w \in \{0,1\}^* \mid w \text{ contains 001 or 0101 as a substring} \} \]

Nondeterministic FA can also use \( \varepsilon \)-transitions:

[Diagram of a nondeterministic finite automaton with \( \varepsilon \)-transitions]
Nondeterminism

Example:
\[ \{ w \in \{0,1\}^* \mid w \text{ contains 1 in the third position from the end} \} \]

Does there exist a (deterministic) FA recognizing this language?

Does there exist a (deterministic) FA recognizing this language?
Example:

\{ w \in \{0\}^* \mid |w| \text{ is divisible by 2 or 3} \}
Nondeterminism

Formal definition:

A **nondeterministic finite automaton** (NFA) is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where

- $Q$ is a finite set of states
- $\Sigma$ is a (finite) alphabet
- $\delta$: $Q \times \Sigma \cup \{\varepsilon\} \rightarrow \mathcal{P}(Q)$ is the transition function
- $q_0 \in Q$ is the start state
- $F \subseteq Q$ is the set of accept states
Let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA and let $w = w_1 w_2 \ldots w_n$ where each $w_i \in \Sigma$. Then **$N$ accepts $w$** if there exists a sequence of states $r_0, r_1, r_2, \ldots, r_n$ s.t.

1) $r_0 = q_0$

2) $\delta(r_i, w_{i+1}) \subseteq r_{i+1}$  \hspace{1cm} $\forall i \in \{0, \ldots, n-1\}$

3) $r_n \in F$

The correct definition of acceptance if the NFA does not use any $\epsilon$ transitions, i.e. if $\forall q \in Q: \delta(q, \epsilon) = \emptyset$.

$\text{blue: correct also for } \epsilon\text{-transitions}$

$L(N) = \{ w \in \Sigma^* \mid N \text{ accepts } w \}$
Thm 1.39: Every NFA has an equivalent DFA.

Proof idea:
- for starters, no $\varepsilon$-transitions in the NFA

- example:

Suppose an NFA $N = (Q, \Sigma, \delta, q_0, F)$
Want a DFA $M = (Q_M, \Sigma, \delta_M, q_m, F_M)$

Let: $Q_M = P(Q)$
$q_m = \{q_0\}$
$F_M = \{ m \in Q_M \mid m \cap F \neq \emptyset \}$
$\delta_M(m, \sigma) = \bigcup_{q \in m} \delta(q, \sigma)$

Note: we did not draw the unreachable states $013$ and
Thm 1.39: Every NFA has an equivalent DFA.

Proof idea, part 2 (getting rid of ε-transitions in the NFA):
- for \( R \subseteq Q \) let:

\[
E(R) = \{ q \in Q \mid q \text{ can be reached from } R \text{ by traveling along 0 or more } \varepsilon \text{-arrows} \}
\]

after having \( N' \), we apply the subset construction with

\[ q_M = E(\{q_0, 3\}) \]
Thm 1.45 (revisited): The class of regular languages is closed under the union operation.

Idea:

Let $N_1 = \langle Q_1, \Sigma, \delta_1, q_1, F_1 \rangle$

$N_2 = \langle Q_2, \Sigma, \delta_2, q_2, F_2 \rangle$

Suppose $Q_1 \cap Q_2 = \emptyset$

Want: an NFA $N = \langle Q, \Sigma, \delta, q_0, F \rangle$

s.t. $L(N) = L(N_1) \cup L(N_2)$

Let $Q = Q_1 \cup Q_2 \cup \{q_o\}$

where $q_o \notin Q_1 \cup Q_2$

$\Sigma = \Sigma_1 \cup \Sigma_2$

$F = F_1 \cup F_2$

$\delta(q_o, \cdot) = \{ q_1, q_2 \}$

$\delta(q, \cdot) = \emptyset$ \quad $\forall q \in Q$

$\delta(q, \cdot) = \delta_1(q, \cdot)$ \quad $\forall q \in Q_1, \forall \cdot \in \Sigma$

$\delta(q, \cdot) = \delta_2(q, \cdot)$ \quad $\forall q \in Q_2, \forall \cdot \in \Sigma$

$\square$
Thm 1.47: The class of regular languages is closed under the concatenation operation.

\[ N_1 = (Q_1, \Sigma_1, \delta_1, q_{11}, F_1) \]
\[ N_2 = (Q_2, \Sigma_2, \delta_2, q_{21}, F_2) \]

\[ N = (Q, \Sigma, \delta, q_0, F) \text{ s.t. } L(N) = L(N_1) \cdot L(N_2) \]

\[ \delta(q, \sigma) = \begin{cases} 
\delta_2(q, \sigma) & \text{if } q \in Q_2, \sigma \in \Sigma_2 \\
\delta_1(q, \sigma) & \text{if } q \in Q_1 - F_1, \sigma \in \Sigma_1 \\
\delta_1(q, \sigma) & \text{if } q \in F_1, \sigma \in \Sigma_1 \\
(q_2) & \text{if } q \in F, \sigma = \varepsilon 
\end{cases} \]
Thm 1.49: The class of regular languages is closed under the star operation.

Example:

Here: \( N_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1) \)

Want: \( N = (Q, \Sigma, \delta, q_0, F) \) s.t. \( L(N) = L(N_1)^* \)

Let: \( Q = Q_1 \cup \{q_0\} \) assume \( q_0 \notin Q_1 \)
\( F = \{q_0\} \)

\( \delta(q, \sigma) = \begin{cases} 
\delta_1(q, \sigma) & \text{if } q \in Q_1, F_1, \sigma \in \Sigma \epsilon \\
\delta_1(q, \sigma) & \text{if } q \in F_1, \sigma \in \Sigma \\
\delta_1(q, \sigma) \cup \{q_0\} & \text{if } q \in F_1, \sigma = \epsilon \\
\{q_1\} & \text{if } q = q_0, \sigma = \epsilon \\
\emptyset & \text{if } q = q_0, \sigma \in \Sigma 
\end{cases} \)