Determinism: computation always continues in a uniquely determined way.

Nondeterminism: have more (or none) choices

Example:

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

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

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

Example:

\[ \{ w \in \{0\}^* \mid |w| \text{ is divisible by 2 or 3} \} \text{ with at most } 8 \text{ sinks} \]
**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

For DFA:
\[\delta: Q \times \Sigma \rightarrow Q\]
Let $N=(Q, \Sigma, \delta, q_0, F)$ be an NFA and let $w=w_1w_2...w_n$ where each $w_i \in \Sigma$. Then $N$ accepts $w$ if there exists a sequence of states $r_0, r_1, ..., r_n$ such that:

1) $r_0 = q_0$

2) $r_{i+1} \in \delta(r_i, w_{i+1}) \quad \forall i \in \{0, 1, ..., n-1\}$

3) $r_n \in F$
Thm 1.39: Every NFA has an equivalent DFA.

Proof idea:
- for starters, no $\epsilon$-transitions in the NFA
- example:
Thm 1.39: Every NFA has an equivalent DFA.

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

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

Where:
- $Q_M = \mathcal{P}(Q_N)$
- $q_{0M} = \varepsilon(q_{0N})$
- $F_M = \{ s \subseteq Q_N \mid s \cap F_N \neq \emptyset \}$
- $\delta_M(s, a) = E\left( \bigcup_{s' \subseteq s} \delta_N(s', a) \right)$

Example:
- $E(\{q_2\}) = \{q_3, q_5\}$
- $E(\{q_0\}) = \{q_0, q_1, q_2, q_3, q_5\}$
- $E(\{q_1, q_4\}) = \{q_1, q_2, q_3, q_5, q_4, q_6\}$
Thm 1.45 (revisited): The class of regular languages is closed under the union operation.

Let \( N_1 = (Q_1, \Sigma_1, \delta_1, q_{1i}, F_1) \) and \( N_2 = (Q_2, \Sigma_2, \delta_2, q_{2i}, F_2) \) be two NFAs.

We want to construct \( N = (Q, \Sigma, \delta, q_0, F) \) such that \( L(N) = L(N_1) \cup L(N_2) \).

Let:
- \( Q = Q_1 \cup Q_2 \cup \{ q_{\text{uni}} \} \)
- \( \Sigma = \Sigma_1 \cup \Sigma_2 \)
- \( q_0 = q_{\text{uni}} \)
- \( F = F_1 \cup F_2 \)
- \( \delta(q, a) = \begin{cases} \delta_1(q, a) & \text{if } q \in Q_1, a \in \Sigma_1 \cup \Sigma_2 \\ \delta_2(q, a) & \text{if } q \in Q_2, a \in \Sigma_1 \cup \Sigma_2 \\ \{ q_{\text{uni}} \} & \text{if } q = q_{\text{uni}} , a \in \Sigma_1 \cup \Sigma_2 \\ \emptyset & \text{if } q \notin Q_1 \cup Q_2 \end{cases} \)
Closure under regular operations

**Thm 1.47**: The class of regular languages is closed under the concatenation operation.

- For any two regular languages, their concatenation is also a regular language.
- A language is said to be regular iff there exists a (D)FA for the language.
- We know that for every NFA there exists an equivalent DFA and vice versa.

Assume \( N_1 = (Q_1, \Sigma, \delta_1, q_1^0, F_1) \) are two NFAs, i.e \{1,2\}.

Given two NFAs, \( N_1 \) and \( N_2 \), we can construct their concatenation \( N_1 N_2 \). The accept states of \( N_1 \) are not accept states anymore.

Do formal description on your own!
Thm 1.49: The class of regular languages is closed under the star operation.

Let $N_i = (Q_i, \Sigma_i, \delta_i, q_{i0}, F_i)$ be an NFA. We want to construct an NFA $N = (Q, \Sigma, \delta, q_0, F)$ such that $L(N) = L(N_i)^*$.

Given words $w_1, w_2, \ldots, w_k$ where each $w_i \in L(N)$, it is supposed to be accepted by $N$. 

Diagram: 

- $N_i$ is the initial NFA.
- $\epsilon$ transitions are shown.
- The construction step is illustrated by extending paths with $\epsilon$ transitions to include all $w_i$ words.

Diagram: 

- $N$ is the constructed NFA.
- $\epsilon$ transitions are shown.
- The extended paths cover all possible combinations of $w_i$ words.