Non-deterministic Finite Automata (NFA)

NFA *versus* DFA

- In a DFA, at every state q, for every symbol a, there is a unique a-transition i.e. there is a unique q' such that $q \xrightarrow{a} q'$.

  This is not necessarily so in an NFA. At any state, an NFA may have multiple a-transitions, or none.

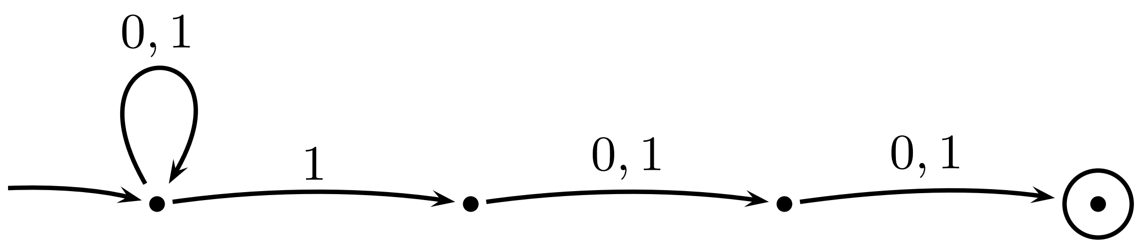
- In a DFA, transition arrows are labelled by symbols from Σ; in an NFA, they are labelled by symbols from $\Sigma \cup \{ \epsilon \}$. I.e. an NFA may have ϵ-transitions.

- We may think of the non-determinism as a kind of parallel computation wherein several processes can be running concurrently.

  When the NFA splits to follow several choices, that corresponds to a process "forking" into several children, each proceeding separately. If at least one of these accepts, then the entire computation accepts.

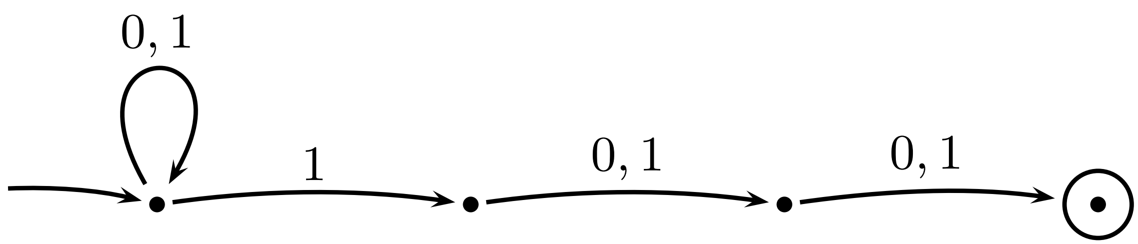
Example: All strings containing a 1 in third position from the end

NFA:

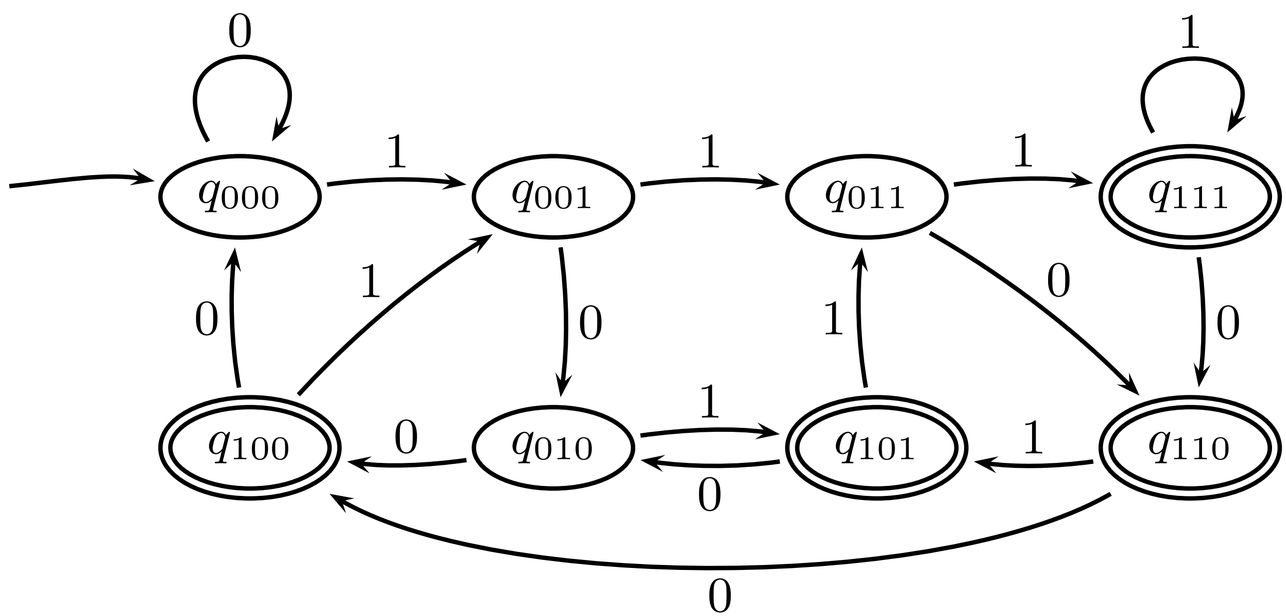


Example: All strings containing a 1 in third position from the end

NFA:

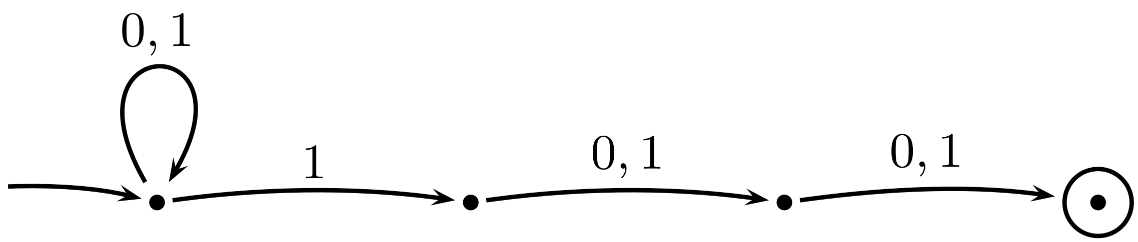


DFA:

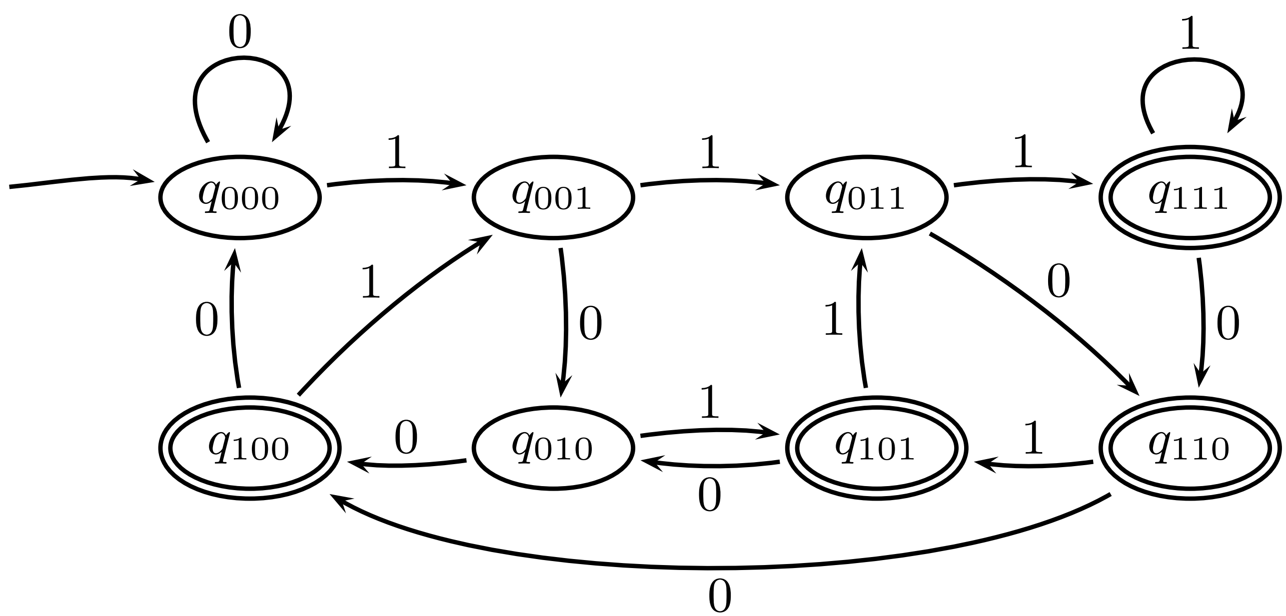


Example: All strings containing a 1 in third position from the end

NFA:

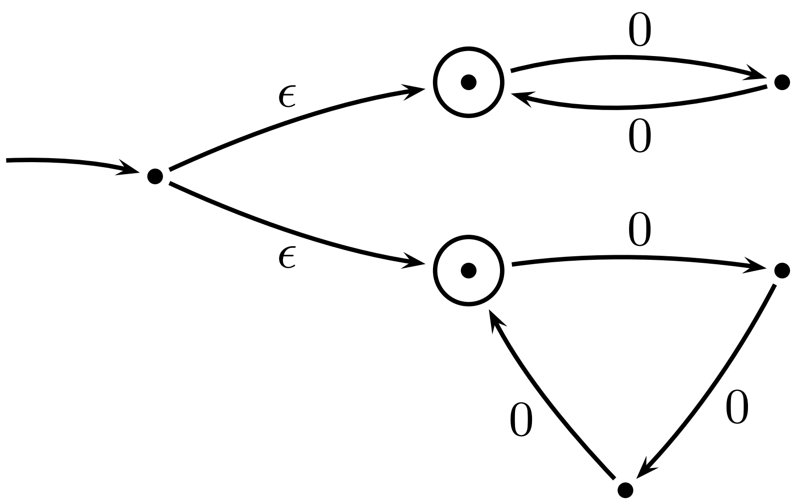


DFA:



NFAs are more compact - they generally require fewer states to recognize a language.

Example: $\{\, 0^k : k$ is a multiple of 2 or 3 $\}$



Using ϵ-transitions and non-determinism, a language defined by an NFA can be easier to understand.

Definition: NFA

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

(i) Q is a finite set of states

(ii) Σ is a finite alphabet

(iii) $q_0 \in Q$ is the start state

(iv) $\delta : Q \times (\Sigma \cup \{ \epsilon \}) \longrightarrow \mathcal{P}(Q)$ is the transition function

(v) $F \subseteq Q$ is the set of final states.

Note: $\mathcal{P}(Q) \stackrel{\text{def}}{=} \{ X : X \subseteq Q \}$ is the *power set* of Q. Equivalently δ can be presented as a relation, i.e. a subset of $(Q \times (\Sigma \cup \{ \epsilon \})) \times Q$.

For $a \in \Sigma \cup \{ \epsilon \}$ we define $q \stackrel{a}{\longrightarrow} q' \stackrel{\text{def}}{=} q' \in \delta(q, a)$.

Some definitions and notations

Fix an NFA $N = (Q, \Sigma, \delta, q_0, F)$.

$L(N)$, the *language accepted by N*, consists of all strings w over Σ satisfying $q_0 \stackrel{w}{\Longrightarrow} q$ where q is a final state. Here $\cdot \stackrel{-}{\Longrightarrow} \cdot$ is defined by:

- $q \stackrel{\epsilon}{\Longrightarrow} q'$ iff $q = q'$ or there is a sequence $q \xrightarrow{\epsilon} \cdots \xrightarrow{\epsilon} q'$ of one or more ϵ-transitions in N from q to q'.
- For $w = a_1 \cdots a_{n+1}$ where each $a_i \in \Sigma$, $q \stackrel{w}{\Longrightarrow} q'$ iff there are $q_1, q_1', \cdots, q_{n+1}, q_{n+1}'$ (not necessarily all distinct) such that

$$q \stackrel{\epsilon}{\Longrightarrow} q_1 \xrightarrow{a_1} q_1' \stackrel{\epsilon}{\Longrightarrow} q_2 \xrightarrow{a_2} q_2' \stackrel{\epsilon}{\Longrightarrow} \cdots q_n' \stackrel{\epsilon}{\Longrightarrow} q_{n+1} \xrightarrow{a_{n+1}} q_{n+1}' \stackrel{\epsilon}{\Longrightarrow} q'$$

Intuitively $q \stackrel{w}{\Longrightarrow} q'$ means:

> "There is a sequence of transitions from q to q' in N in which the symbols in w occur in the correct order, but with 0 or more ϵ-transitions before or after each one".

We shall sometimes write $\hat{\delta}(q, w) = \{ q' \in Q : q \stackrel{w}{\Longrightarrow} q' \}$, for $w \in \Sigma^*$.

Note: In case N is a DFA, for any $q \in Q$ and $w \in \Sigma^*$, there is a *unique* q' such that $q \stackrel{w}{\Longrightarrow} q'$ (thus, by abuse of notation, we write $\hat{\delta}(q, w) = q'$).

Exercise. Writing $w = a_1 \cdots a_{n+1}$, we have $q \stackrel{w}{\Longrightarrow} q'$ is equivalent to: there exist $q_1, \cdots, q_n$ such that

$$q \stackrel{a_1}{\Longrightarrow} q_1 \stackrel{a_2}{\Longrightarrow} \cdots\cdots \stackrel{a_{n+1}}{\Longrightarrow} q'$$

Equivalence of NFAs and DFAs: The Subset Construction

Observation. Every DFA *is* an NFA!

Say two automata are *equivalent* if they accept the same language.

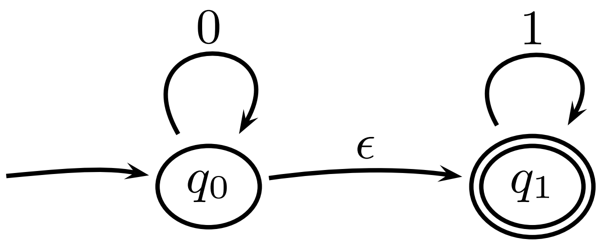
Theorem (Determinization). Every NFA has an equivalent DFA.

Proof. Fix an NFA $N = (Q_N, \Sigma_N, \delta_N, q_N, F_N)$, we construct an equivalent DFA $\mathcal{P}N = (Q_{\mathcal{P}N}, \Sigma_{\mathcal{P}N}, \delta_{\mathcal{P}N}, q_{\mathcal{P}N}, F_{\mathcal{P}N})$ such that $L(N) = L(\mathcal{P}N)$:

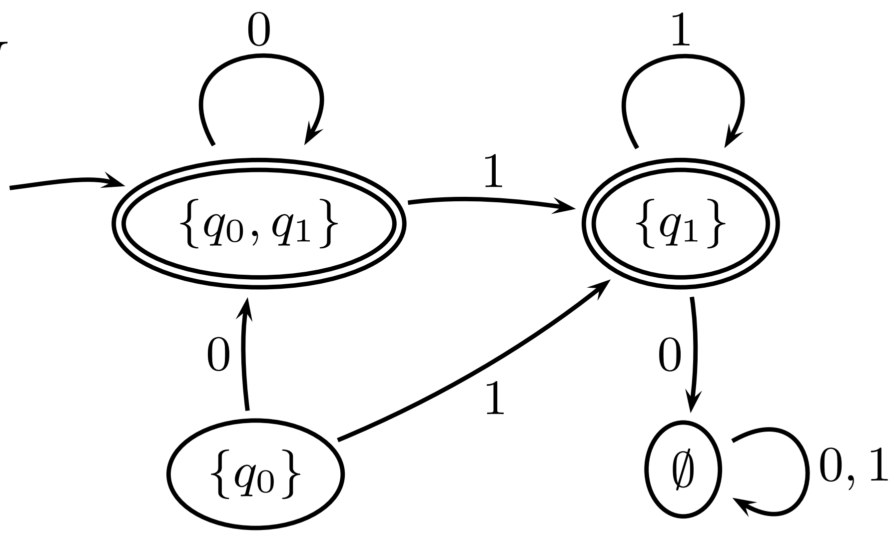
- $Q_{\mathcal{P}N} \stackrel{\text{def}}{=} \{ S : S \subseteq Q_N \}$
- $\Sigma_{\mathcal{P}N} \stackrel{\text{def}}{=} \Sigma_N$
- $S \xrightarrow{a} S'$ in $\mathcal{P}N$ iff $S' = \{ q' : \exists q \in S.(q \stackrel{a}{\Longrightarrow} q' \text{ in } N) \}$
- $q_{\mathcal{P}N} \stackrel{\text{def}}{=} \{ q : q_N \stackrel{\epsilon}{\Longrightarrow} q \}$
- $F_{\mathcal{P}N} \stackrel{\text{def}}{=} \{ S \in Q_{\mathcal{P}N} : F_N \cap S \neq \emptyset \}$

Example. All words that begin with a string of 0's followed by a string of 1's.

NFA N



DFA $\mathcal{P}N$



Note. State $\{q_0\}$ is redundant.

Proof of "$L(N) \subseteq L(\mathcal{P}N)$":

Suppose $\epsilon \in L(N)$. Then $q_N \stackrel{\epsilon}{\Longrightarrow} q'$ for some $q' \in F_N$. Hence $q' \in q_{\mathcal{P}N}$, and so, $q_{\mathcal{P}N} = \{\, q'' : q_N \stackrel{\epsilon}{\Longrightarrow} q'' \,\} \in F_{\mathcal{P}N}$ i.e. $\epsilon \in L(\mathcal{P}N)$.

Now take any non-null $u = a_1 \cdots a_n$. Suppose $u \in L(N)$. Then there is a sequence of N-transitions

$$q_N \stackrel{a_1}{\Longrightarrow} q_1 \stackrel{a_2}{\Longrightarrow} \cdots\cdots \stackrel{a_n}{\Longrightarrow} q_n \in F_N \tag{1}$$

Since $\mathcal{P}N$ is deterministic, feeding $a_1, \cdots, a_n$ to it results in the sequence of $\mathcal{P}N$-transitions

$$q_{\mathcal{P}N} \xrightarrow{a_1} S_1 \xrightarrow{a_2} \cdots\cdots \xrightarrow{a_n} S_n \tag{2}$$

where

$$\begin{array}{rcl} S_1 & = & \{\, q' : \exists q \in q_{\mathcal{P}N}.(q \stackrel{a_1}{\Longrightarrow} q' \text{ in } N) \,\} \\ S_2 & = & \{\, q' : \exists q \in S_1.(q \stackrel{a_2}{\Longrightarrow} q' \text{ in } N) \,\} \\ \vdots & & \vdots \end{array}$$

By definition of $\delta_{\mathcal{P}N}$, from (1), we have $q_1 \in S_1$, and so $q_2 \in S_2$, $\cdots$, and so

$q_n \in S_n$, and hence $S_n \in F_{\mathcal{P}N}$ because $q_n \in F_N$. Thus (2) shows that $u \in L(\mathcal{P}N)$.

Proof of "$L(\mathcal{P}N) \subseteq L(N)$":

Suppose $\epsilon \in L(\mathcal{P}N)$. Then $q_{\mathcal{P}N} \in F_{\mathcal{P}N}$ i.e. $F_N \cap \{\, q : q_N \stackrel{\epsilon}{\Longrightarrow} q \,\} \neq \emptyset$, or equivalently, for some $q' \in F_N$, $q_N \stackrel{\epsilon}{\Longrightarrow} q'$. Hence $\epsilon \in L(N)$.

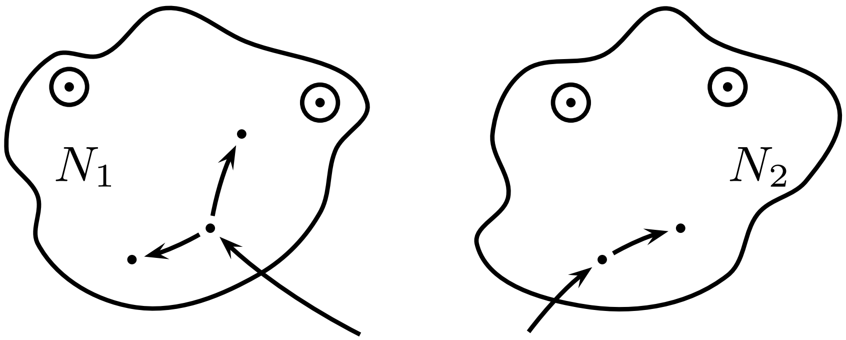
Now suppose some non-null $u = a_1 \cdots a_n \in L(\mathcal{P}N)$. I.e. there is a sequence of $\mathcal{P}N$-transitions of the form (2) with $S_n \in F_{\mathcal{P}N}$ i.e. with S_n containing some $q_n \in F_N$. Now since $q_n \in S_n$, by definition of $\delta_{\mathcal{P}N}$, there is some $q_{n-1} \in S_{n-1}$ with $q_{n-1} \stackrel{a_n}{\Longrightarrow} q_n$ in N. Working backwards in this way, we can build up a sequence of N-transitions like (1), until we deduce that $q_N \stackrel{a_1}{\Longrightarrow} q_1$. Thus we get a sequence of N-transitions with $q_n \in F_N$, and hence $u \in L(N)$.

$\square$

Closure under regular operations revisited

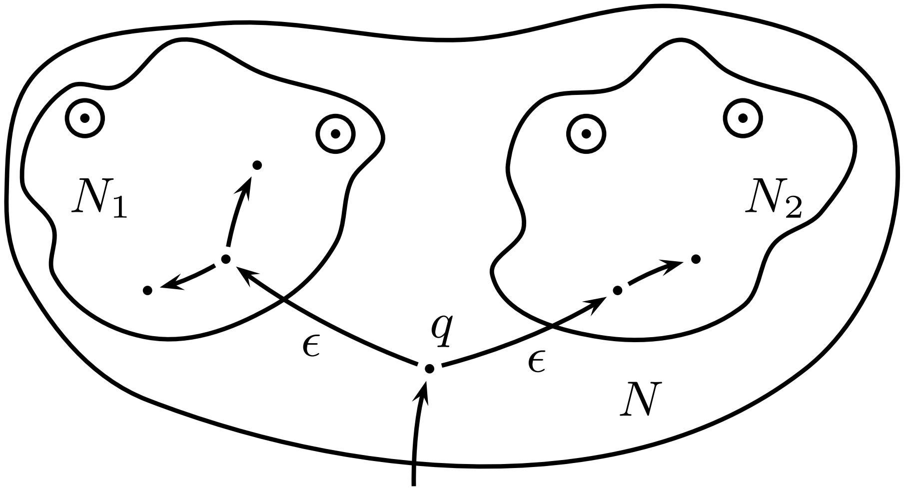
Using nondeterminism makes *some* proofs much easier.

Theorem. Regular languages are closed under union.



Take NFAs N_1 and N_2.

Define N that accepts $L(N_1) \cup L(N_2)$ by adding a new start state q to the disjoint union of (the respective state transition graphs of) N_1 and N_2, and a ϵ-transition from q to each start state of N_1 and N_2.



Regular languages are closed under union (cont'd)

More formally, given NFAs $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$, we define $N = (Q, \Sigma, \delta, q, F)$ by

$$
\begin{aligned}
Q &= Q_1 \times \{1\} \cup Q_2 \times \{2\} \cup \{q\} \\
F &= F_1 \times \{1\} \cup F_2 \times \{2\} \\[1ex]
\delta(q, \epsilon) &= \{(q_1, 1), (q_2, 2)\} \\
\delta((r, 1), a) &= \{(r', 1) \mid r' \in \delta_1(r, a)\} \\
\delta((r, 2), a) &= \{(r', 2) \mid r' \in \delta_2(r, a)\}
\end{aligned}
$$

Regular languages are closed under union (cont'd)

Proof of "$L(N_1) \cup L(N_2) \subseteq L(N)$":

Suppose $w = a_1 \cdots a_n \in L(N_1)$ then there exist $r_1, \ldots, r_n \in Q_1$ such that

$$q_1 \xRightarrow{a_1} r_1 \xRightarrow{a_2} \cdots \xRightarrow{a_n} r_n$$

with $r_n \in F_1$. Then in N we got the sequence

$$q \xrightarrow{\epsilon} (q_1, 1) \xRightarrow{a_1} (r_1, 1) \xRightarrow{a_2} \cdots \xRightarrow{a_n} (r_n, 1)$$

with $(r_n, 1) \in F$. Hence $w \in L(N)$.

Similarly we can show that $w \in L(N_2)$ implies $w \in L(N)$.

Regular languages are closed under union (cont'd)

Proof of "$L(N) \subseteq L(N_1) \cup L(N_2)$":

Suppose $w = a_1 \cdots a_n \in L(N)$ then there exist $i \in \{1, 2\}$ and $r_1, \ldots, r_n \in Q_i$ such that

$$q \xrightarrow{\epsilon} (q_i, i) \overset{a_1}{\Longrightarrow} (r_1, i) \overset{a_2}{\Longrightarrow} \cdots \overset{a_n}{\Longrightarrow} (r_n, i)$$

with $(r_n, i) \in F$. But then, in N_i, we have the sequence

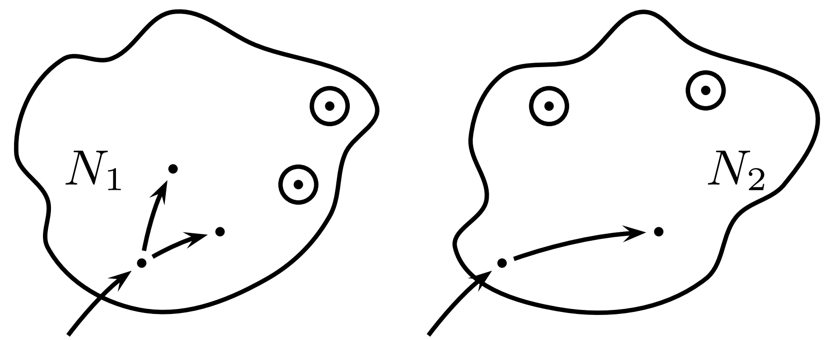
$$q_i \overset{a_1}{\Longrightarrow} r_1 \overset{a_2}{\Longrightarrow} \cdots \overset{a_n}{\Longrightarrow} r_n$$

with $r_n \in F_i$. Hence $w \in L(N_i)$.

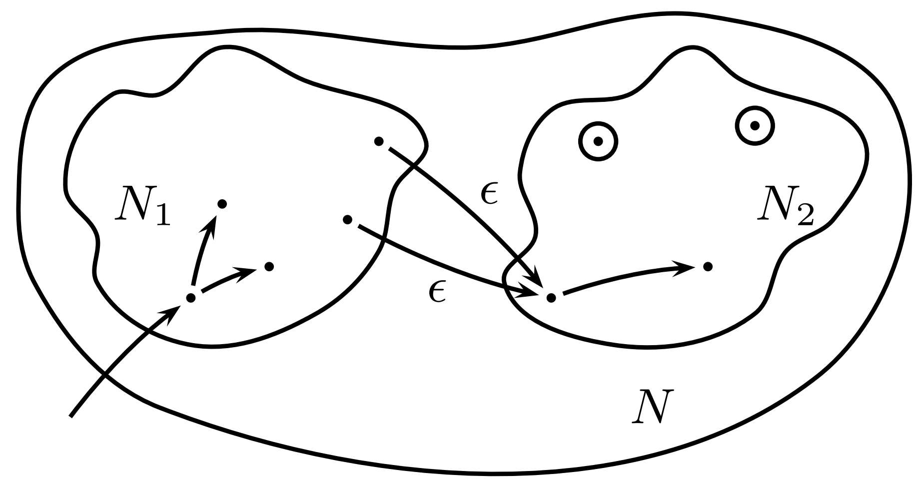
$\square$

Theorem. Regular languages are closed under concatenation.

Take NFAs N_1 and N_2.



An NFA N that accepts $L(N_1) \cdot L(N_2)$ can be obtained from the disjoint union of N_1 and N_2 by making the start state of N_1 the start state of N, and by adding an ϵ-transition from each accepting state of N_1 to the start state of N_2. The accepting states of N are those of N_2.



Regular languages are closed under concatenation (cont'd)

More formally, given NFAs $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $N_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$, we define $N = (Q, \Sigma, \delta, (q_1, 1), F)$ by

$$
\begin{array}{lcll}
Q & = & Q_1 \times \{1\} \quad \cup \quad Q_2 \times \{2\} & \\
F & = & F_2 \times \{2\} & \\
\delta((r,1),a) & = & \{(r',1) \mid r' \in \delta_1(r,a)\} & \text{for } a \neq \epsilon \text{ or } r \notin F_1 \\
\delta((r,1),\epsilon) & = & \{(r',1) \mid r' \in \delta_1(r,\epsilon)\} \cup \{(q_2,2)\} & \text{for } r \in F_1 \\
\delta((r,2),a) & = & \{(r',2) \mid r' \in \delta_2(r,a)\} &
\end{array}
$$

Regular languages are closed under concatenation (cont'd)

Proof of "$L(N_1)\,L(N_2) \subseteq L(N)$":

Suppose $w \in L(N_1)\,L(N_2)$ then there exist $u = a_1 \cdots a_n \in L(N_1)$ and $v = b_1 \cdots b_m \in L(N_2)$ with $w = uv$.

Therefore there exist $r_1, \ldots, r_n \in Q_1$ with $r_n \in F_1$ such that, in N_1,

$$q_1 \xRightarrow{a_1} r_1 \xRightarrow{a_2} \cdots \xRightarrow{a_n} r_n$$

$s_1, \ldots, s_m \in Q_2$ with $s_m \in F_2$ such that, in N_2,

$$q_2 \xRightarrow{b_1} s_1 \xRightarrow{b_2} \cdots \xRightarrow{b_m} s_m$$

Then in N we got the sequence

$$(q_1, 1) \xRightarrow{a_1} (r_1, 1) \xRightarrow{a_2} \cdots \xRightarrow{a_n} (r_n, 1) \xrightarrow{\epsilon} (q_2, 2) \xRightarrow{b_1} (s_1, 2) \xRightarrow{b_2} \cdots \xRightarrow{b_m} (s_m, 2)$$

with $(s_m, 2) \in F$. Hence $w \in L(N)$.

Regular languages are closed under concatenation (cont'd)

Proof of "$L(N) \subseteq L(N_1)\, L(N_2)$":

Suppose $w = a_1 \cdots a_k \in L(N)$, then there exist $r_1, \ldots, r_k \in Q$ such that,

$$(q_1, 1) \xRightarrow{a_1} r_1 \xRightarrow{a_2} \cdots \xRightarrow{a_k} r_k$$

with $r_k \in F$. By definition of F there is an $s_k \in F_2$ such that $r_k = (s_k, 2)$. The definition of δ implies that there is exactly one ϵ-transition to get from the first to the second component, i.e. there are states $s_1, \ldots, s_n \in Q_1$ and $s_{n+1}, \ldots, s_{k-1} \in Q_2$ such that

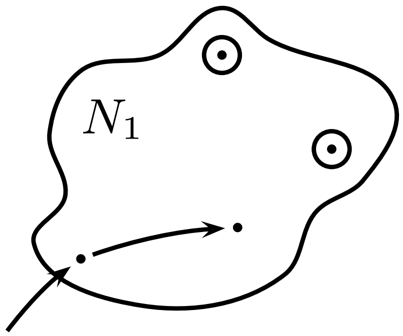
$$(q_1, 1) \xRightarrow{a_1} (s_1, 1) \xRightarrow{a_2} \cdots \xRightarrow{a_n} (s_n, 1) \xrightarrow{\epsilon} (q_2, 2) \xRightarrow{a_{n+1}} (s_{n+1}, 2) \xRightarrow{a_{n+2}} \cdots \xRightarrow{a_k} (s_k, 2)$$

Then, in N_1, we have the sequence $q_1 \xRightarrow{a_1} s_1 \xRightarrow{a_2} \cdots \xRightarrow{a_n} s_n$ with $s_n \in F_1$ and in N_2 we have the sequence $q_2 \xRightarrow{a_{n+1}} s_{n+1} \xRightarrow{a_{n+2}} \cdots \xRightarrow{a_k} s_k$ with $s_k \in F_2$. Hence $u = a_1 \cdots a_n \in L(N_1)$ and $v = a_{n+1} \cdots a_k \in L(N_2)$, and therefore $w = uv \in L(N_1)\, L(N_2)$. $\square$

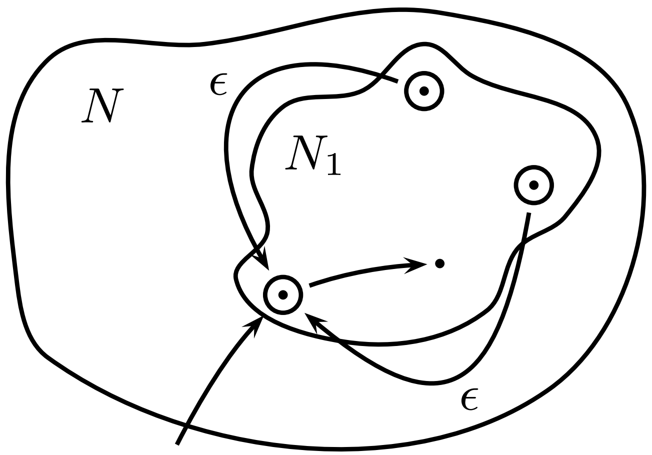
Theorem. Regular languages are closed under star.

First attempt:

Take an NFA $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ that accepts A_1. Construct N that accepts $A_1^* = \{\epsilon\} \cup A_1 \cup A_1\,A_1 \cup \cdots$.



Obtain N from N_1 by making the start state accepting, and by adding a new ϵ-transition from each accepting state to the start state.



What is wrong with this?

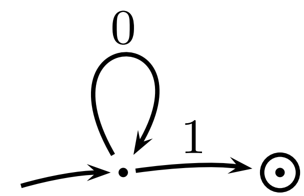
Consider the two-node two-edge NFA N_1

that accepts $\{\, 0^i 1 : i \geq 0 \,\}$

(namely the language

defined by the regular expression 0^*1).

N_1



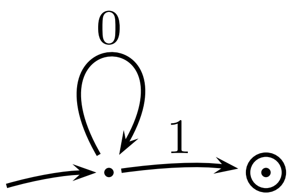
Consider the two-node two-edge NFA N_1

that accepts $\{ 0^i 1 : i \geq 0 \}$

(namely the language

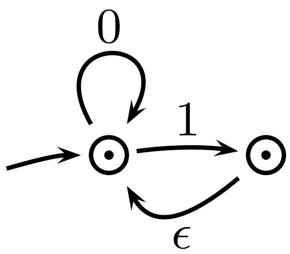
defined by the regular expression 0^*1).

N_1



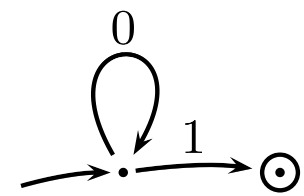
The above construction gives the NFA N

N



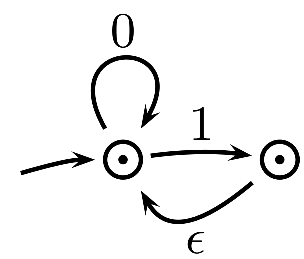
Consider the two-node two-edge NFA N_1
that accepts $\{\, 0^i 1 : i \geq 0 \,\}$
(namely the language
defined by the regular expression 0^*1).

N_1



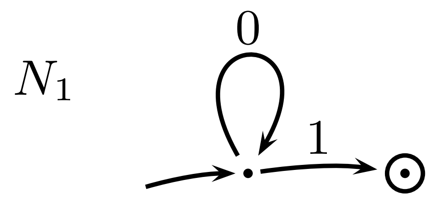
The above construction gives the NFA N

N

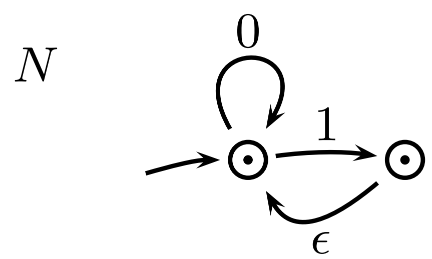


But N accepts $(0 + 0^*1)^* = (0+1)^* \neq (0^*1)^*$.
E.g. N accepts 010 which is not in $L(N_1)^*$

Consider the two-node two-edge NFA N_1
that accepts $\{ 0^i 1 : i \geq 0 \}$
(namely the language
defined by the regular expression 0^*1).

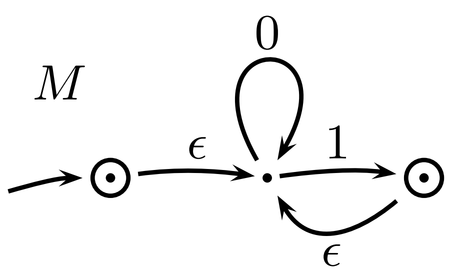


The above construction gives the NFA N



But N accepts $(0 + 0^*1)^* = (0 + 1)^* \neq (0^*1)^*$.
E.g. N accepts 010 which is not in $L(N_1)^*$

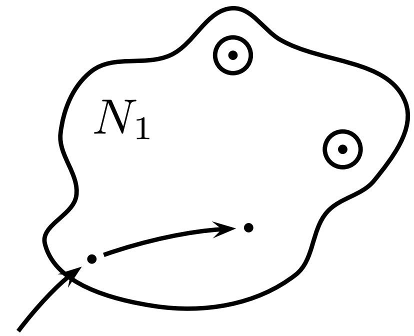
The NFA M accepts $L(N_1)^*$



Proof: Regular languages are closed under star

Second (correct) attempt:

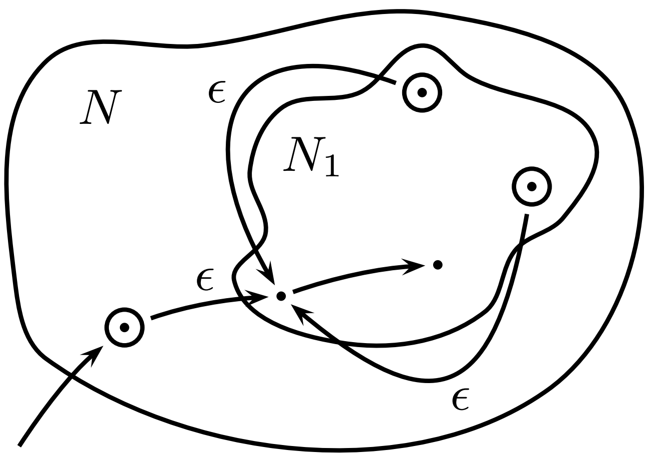
Take an NFA $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ that accepts A_1.



Define $N = (Q_1 \cup \{ q_0 \}, \Sigma, \delta, q_0, F_1 \cup \{ q_0 \})$

where

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & q \in Q_1 \text{ and } q \notin F_1 \\ \delta_1(q,a) & q \in F_1 \text{ and } a \neq \epsilon \\ \delta_1(q,a) \cup \{ q_1 \} & q \in F_1 \text{ and } a = \epsilon \\ \{ q_1 \} & q = q_0 \text{ and } a = \epsilon \\ \emptyset & q = q_0 \text{ and } a \neq \epsilon \end{cases}$$



Regular languages are closed under star (cont'd)

Proof of "$L(N_1)^* \subseteq L(N)$":

Obviously $\epsilon \in L(N)$ because $q_0 \in F$.

Suppose $w \in L(N_1)^*$ and $w \neq \epsilon$ then there exist $k \geq 1$ and $v_1, \ldots, v_k$ such that $w = v_1 \ldots v_k$ and $v_i = a_{i1} \ldots a_{in_i} \in L(N_1)$ for each i. Then for each i there exist $r_{i1}, \ldots, r_{in_i} \in Q_1$ such that

$$q_1 \stackrel{a_{i1}}{\Longrightarrow} r_{i1} \stackrel{a_{i2}}{\Longrightarrow} \cdots \stackrel{a_{in_i}}{\Longrightarrow} r_{in_i}$$

with $r_{in_i} \in F_1$. Then in N we got the sequence

$$q_0 \stackrel{\epsilon}{\longrightarrow} q_1 \stackrel{a_{11}}{\Longrightarrow} r_{11} \stackrel{a_{12}}{\Longrightarrow} \cdots \stackrel{a_{1n_1}}{\Longrightarrow} r_{1n_1} \stackrel{\epsilon}{\longrightarrow} q_1 \stackrel{a_{21}}{\Longrightarrow} \cdots \stackrel{a_{kn_k}}{\Longrightarrow} r_{kn_k}$$

with $r_{kn_k} \in F$. Hence $w \in L(N)$.

Regular languages are closed under star (cont'd)

Proof of "$L(N) \subseteq L(N_1)^*$":

If $w = \epsilon$, $w \in L(N_1)^*$ by definition of star.

Suppose $w = a_1 \cdots a_n \in L(N)$ then there exist $r_1, \ldots, r_n \in Q$ such that

$$q \xrightarrow{\epsilon} q_1 \xRightarrow{a_1} r_1 \xRightarrow{a_2} \cdots \xRightarrow{a_n} r_n$$

with $r_n \in F$.

Let $k - 1$ be the number of occurences of the "new" ϵ-transitions $r_j \xrightarrow{\epsilon} q_1$ with $r_j \in F_1$. If we split the transition sequence at these transitions, we get k transition sequences $q_1 \xRightarrow{v_i} r_i$ such that $w = v_1 \ldots v_k$ and for each i $v_i \in L(N_1)$.

Hence $w = v1 \ldots v_k \in L(N_1)^*$

$\square$