THEORY OF AUTOMATA & FORMAL LANGUAGES

Introduction to ε-NFA

Transitions without consuming input

Understand → Close → Move → Verify

What is an ε-transition?

A transition that consumes no input symbol.

Ordinary transition

Read a symbol, for example a.

Follow an edge labelled a.

Advance the input position by one.

If no matching edge exists, that path stops.

Example: $q_0 \xrightarrow{a} q_1$.

Epsilon transition

Follow an edge labelled ε.

Do not advance the input position.

It may be taken before, between or after symbols.

It is optional, not compulsory.

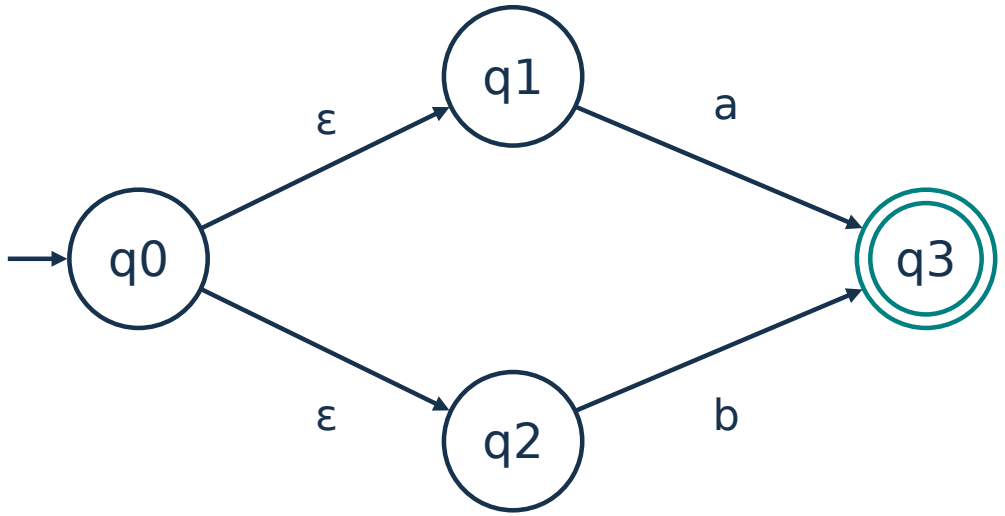
ε is not an input alphabet symbol.

An ε-transition changes the state but leaves the unread input unchanged.

A first ε-NFA

Language: L = {a, b}.

Complete machine



How the machine works

q_0 is the initial state.

Choose q_1 or q_2 without reading input.

From q_1, read a to reach q_3.

From q_2, read b to reach q_3.

Only q_3 is accepting.

The machine accepts a or b, but not ε, ab or ba.

Formal mathematical model

$M = (Q, \Sigma, \delta, q_0, F)$

Five components

$Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{a,b\}$

$\delta: Q \times (\Sigma \cup \{\varepsilon\}) \to \mathscr{P}(Q)$

Initial state: q_0

$F = \{q_3\}$

Transition table: first example

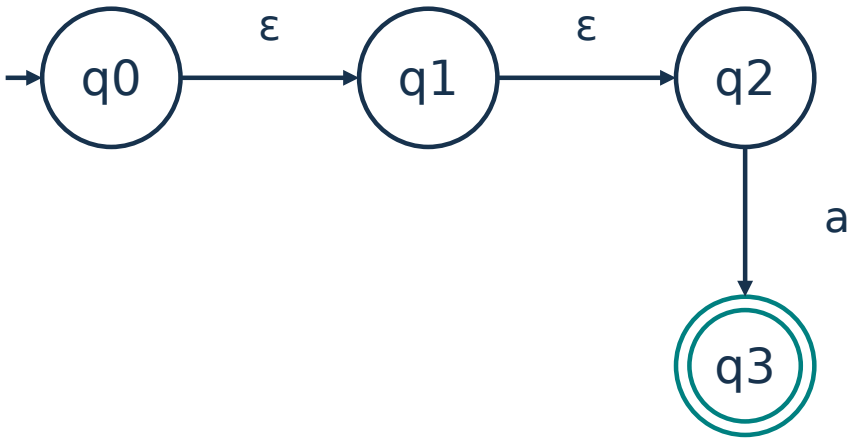
State	a	b	ε
q_0	∅	∅	$\{q_1,q_2\}$
q_1	$\{q_3\}$	∅	∅
q_2	∅	$\{q_3\}$	∅
q_3	∅	∅	∅

A missing transition is represented by the empty set ∅.

Understanding ε-closure

E(q) = states reachable from q using zero or more ε-edges.

Example diagram



Compute each closure

$E(q_0) = \{q_0, q_1, q_2\}$

$E(q_1) = \{q_1, q_2\}$

$E(q_2) = \{q_2\}$

$E(q_3) = \{q_3\}$

$E(\{q_1, q_3\}) = \{q_1, q_2, q_3\}$

Include the starting state: zero ε-transitions are always allowed.

Processing a string

Close before reading input and after every symbol.

General algorithm

$S = E(\{q_0\})$

Read the next symbol x.

T = union of δ(q,x) for every q in S.

Set S = E(T); repeat for remaining input.

Accept exactly when $S \cap F \neq \emptyset$.

Worked example: first machine

Input: b

Initial $S = \{q_0, q_1, q_2\}$

Move on b: $T = \{q_3\}$

Close: $S = E(\{q_3\}) = \{q_3\}$

End of input: $q_3 \in F \rightarrow$ accept.

The closure of a set is the union of the closures of its states.

Acceptance and rejection

Use the first ε-NFA for L = {a,b}.

Accepted: a

Initial closure = $\{q_0,q_1,q_2\}$

Move on a: $\{q_3\}$

Closure: $\{q_3\}$

No input remains.

Accept: q_3 is a final state.

Rejected: ab

Initial closure = $\{q_0,q_1,q_2\}$

Move on a, then close: $\{q_3\}$

Move on b: $\emptyset$

Closure of $\emptyset$ is $\emptyset$.

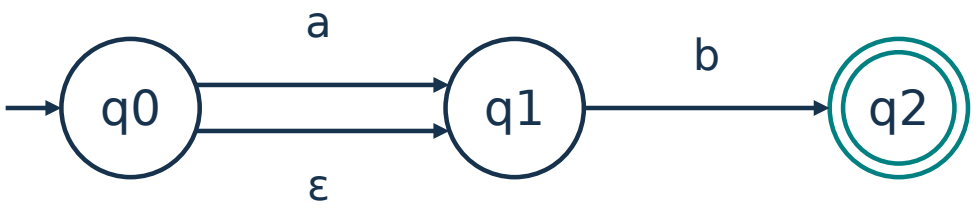
Reject: no complete accepting path.

Reaching a final state early is not sufficient; all input must be consumed.

Optional blocks using ε

Language: L = {b, ab}; expression (ε | a)b.

Complete machine



Manual verification

Input b: q_0 —ε→ q_1 —b→ q_2.

Input ab: q_0 —a→ q_1 —b→ q_2.

Input ε: closure {q_0,q_1}; reject.

Input a: ends at q_1; reject.

Input bb: no complete path; reject.

An ε-edge can bypass an optional part of a language.

Practice and summary

Apply the closure rule before attempting conversions.

Practice questions

1. Does an ε-edge consume a symbol?

2. Is q always included in E(q)?

3. Find E(q_0) for the chain on slide 5.

4. Does the first machine accept ε?

5. Does the optional-block machine accept ab?

Explained answers

1. No; input position stays unchanged.

2. Yes; zero moves are permitted.

3. {q_0,q_1,q_2}, via the ε-chain.

4. No; initial closure has no final state.

5. Yes; read a then b to reach q_2.

Next: removing ε-transitions and converting NFA to DFA.