THEORY OF AUTOMATA & FORMAL LANGUAGES

Direct Conversion of ε-NFA to DFA

Combining epsilon closure with subset construction

Close → Move → Close → Verify

Why convert directly?

Combine ε-closure and subset construction in one procedure.

ε-NFA

ε-transitions consume no input.

Several states may be active at once.

Before reading, include all ε-reachable states.

After reading, take ε-closure again.

Goal: preserve the accepted language.

Equivalent DFA

One state represents a closed subset.

Each input has exactly one destination.

No ε-transitions remain.

Accept if the subset contains a final state.

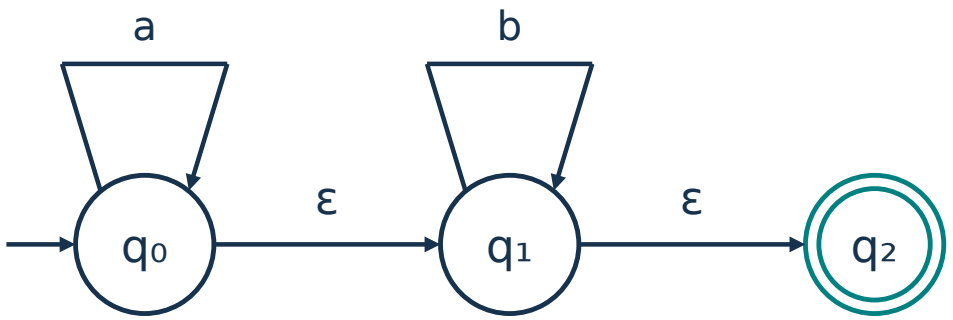
Include the empty subset if reached.

No intermediate ordinary NFA is required.

Worked example: original ε-NFA

Σ = {a,b}; initial q_0; F = {q_2}; language: a*b*.

Original state diagram



Original transition table

State	a	b	ε
q_0	{q_0}	∅	{q_1}
q_1	∅	{q_1}	{q_2}
q_2	∅	∅	∅

The machine may skip either block.

ε-paths may be followed before or after consuming a symbol.

Step 1 — Compute ε-closures

E(S) contains every state reachable from S using zero or more ε-steps.

Closure of each state

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

$q_0 \xrightarrow{\varepsilon} q_1 \xrightarrow{\varepsilon} q_2$

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

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

Every state belongs to its own closure.

Direct subset rule

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

$\delta^D(S,x) = E(\text{move}(S,x))$

move(S,x) = union of δ(q,x) for $q \in S$.

All discovered subsets are ε-closed.

A subset is final if it intersects F.

A is accepting because it contains q_2; therefore ε is accepted.

Step 2 — Process initial subset A

A = $\{q_0,q_1,q_2\}$; compute both destinations.

Read a from A

move(A,a) = $\{q_0\} \cup \varnothing \cup \varnothing$

= $\{q_0\}$

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

This is the existing subset A.

Record A —a→ A.

Read b from A

move(A,b) = $\varnothing \cup \{q_1\} \cup \varnothing$

= $\{q_1\}$

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

Name this new subset B.

Record A —b→ B.

Always apply ε-closure after taking the union of symbol transitions.

Step 3 — Process B and the empty subset

$B = \{q_1, q_2\}$; $C = \emptyset$.

Process B

On a: $\text{move}(B,a) = \emptyset \cup \emptyset$

$E(\emptyset) = \emptyset = C$

On b: $\text{move}(B,b) = \{q_1\} \cup \emptyset$

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

B is accepting; C is not.

Process C

On a: $\text{move}(\emptyset,a) = \emptyset$

$E(\emptyset) = \emptyset = C$

On b: $\text{move}(\emptyset,b) = \emptyset$

$E(\emptyset) = \emptyset = C$

No new subsets remain.

C is a rejecting sink: once reached, no input can leave it.

Step 4 — Draw the equivalent DFA

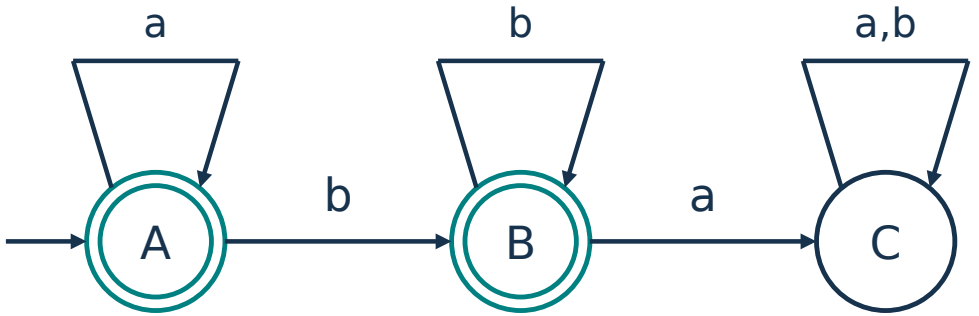
A and B are accepting because both contain q_2.

Complete DFA transition table

State	a	b
→ * A	A	B
* B	C	B
C	C	C

A={q_0,q_1,q_2}; B={q_1,q_2}; C=∅

Resulting DFA



The DFA accepts zero or more a symbols followed by zero or more b symbols.

Step 5 — Verify the conversion

Trace the ε-closed subsets and the DFA on the same input.

Accepted example: aab

Start: A = $\{q_0, q_1, q_2\}$

Read a: A = $\{q_0, q_1, q_2\}$

Read a: A = $\{q_0, q_1, q_2\}$

Read b: B = $\{q_1, q_2\}$

B contains q_2 → accept.

Rejected example: aba

Start: A = $\{q_0, q_1, q_2\}$

Read a: A = $\{q_0, q_1, q_2\}$

Read b: B = $\{q_1, q_2\}$

Read a: C = $\emptyset$

C has no final state → reject.

The traces illustrate equivalence; the subset rule preserves all possible paths.

Practice and summary

Use close → move → close for every input symbol.

Practice questions

1. What is the initial DFA state?

2. Why is A accepting?

3. Does the machine accept bbb?

4. Does the machine accept ba?

5. Why must C have two loops?

Explained answers

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

2. It contains original final state q_2.

3. Yes: A → B → B → B.

4. No: A → B → C.

5. The DFA must be total.

Next: DFA minimization and equivalent states.