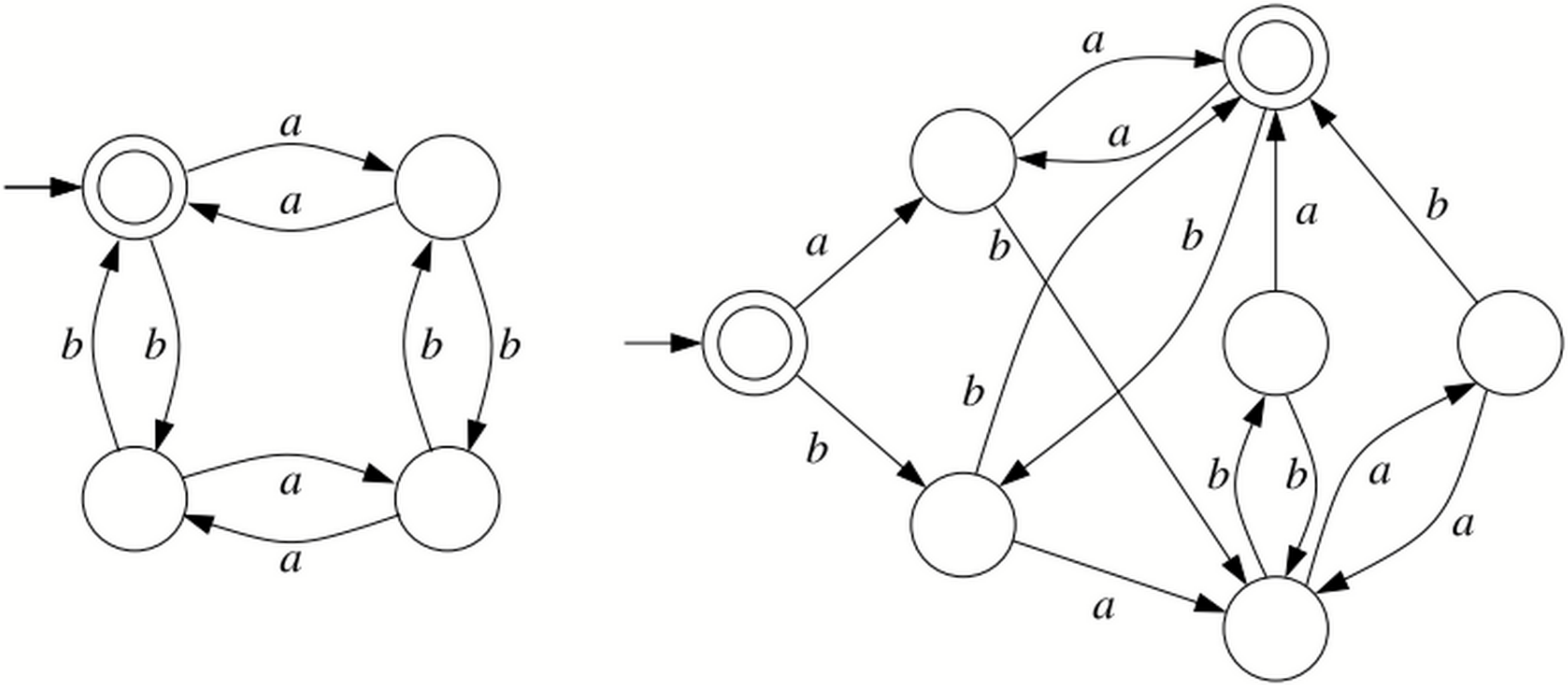
Minimization and Reduction



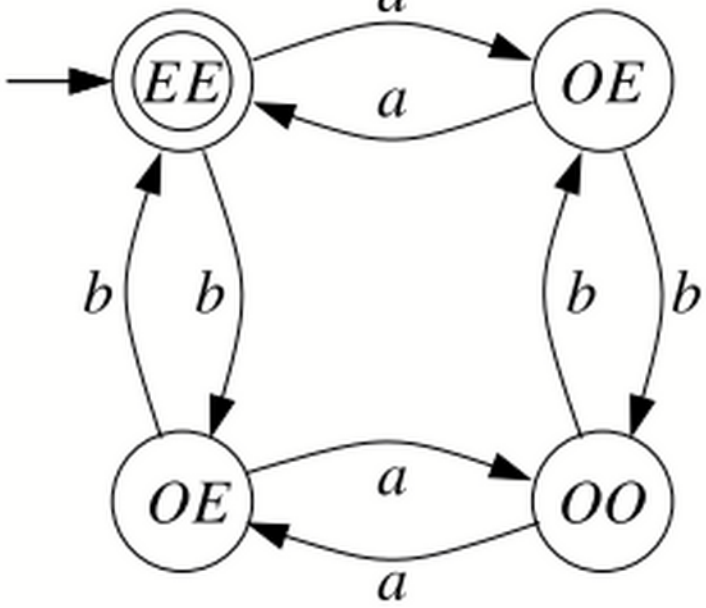
Residual

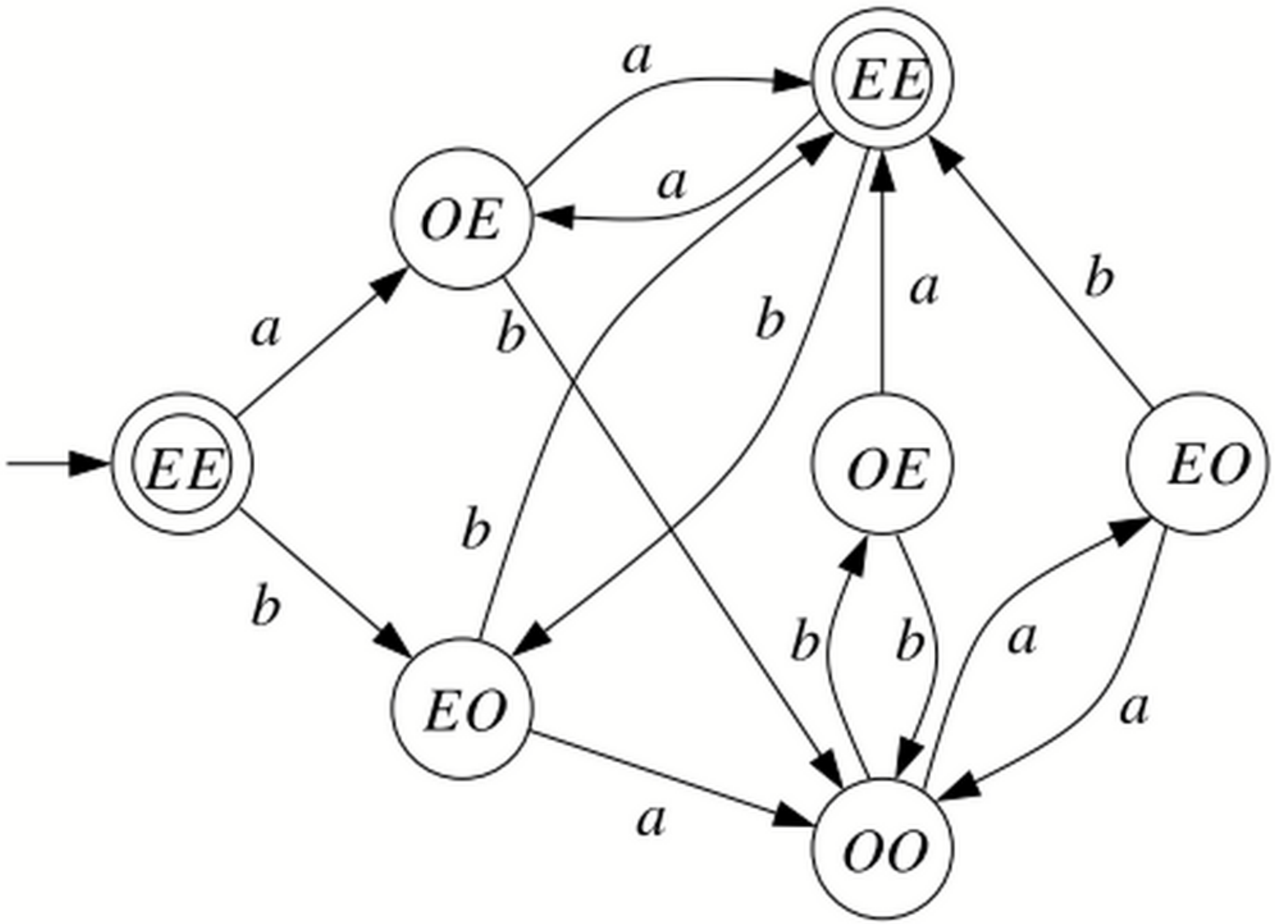
Definition 3.1 *Given a language $L \subseteq \Sigma^*$ and $w \in \Sigma^*$, the w-*residual *of L is the language $L^w = \{u \in \Sigma^* \mid wu \in L\}$. A language $L' \subseteq \Sigma^*$ is a* residual *of L if $L' = L^w$ for at least one $w \in \Sigma^*$.*

Observe : $(L^w)^u = L^{wu}$

Relation between residuals and states

- Let A be a (finite or infinite) deterministic automaton.
- **Def:** The language of a state q of A, denoted by $L_A(q)$ or just $L(q)$, is the language recognized by A with q as initial state.
- **Observation 1**: State-languages are residuals.
  - For every state q of A: $L(q)$ is a residual of $L(A)$.
- **Observation 2**: Residuals are state-languages.
  - For every residual R of $L(A)$: there is a state q such that $R = L(A)$.


EE
OE
OE
OO
a
a
a
a
b
b
b
b


EE
EE
OE
OE
EO
EO
OO
a
a
a
a
a
a
a
b
b
b
b
b
b
b
b

- Important consequence:

  A regular language has finitely many residuals

  or, equivalently

  Languages with infinitely many residuals are not regular

Canonical DFA for a regular language

Definition 3.4 *Let $L \subseteq \Sigma^*$ be a language. The* canonical DA *for L is the DA $C_L = (Q_L, \Sigma, \delta_L, q_{0L}, F_L)$, where:*

- *Q_L is the set of residuals of L; i.e., $Q_L = \{L^w \mid w \in \Sigma^*\}$;*
- *$\delta(K, a) = K^a$ for every $K \in Q_L$ and $a \in \Sigma$;*
- *$q_{0L} = L$; and*
- *$F_L = \{K \in Q_L \mid \varepsilon \in K\}$.*

Example 1: The language EE $\subseteq \{a, b\}^*$

$Q_{EE} =$

$q_{0EE} =$

$F_{EE} =$

$\delta_{EE} =$

Example 2: The language a*b*

$Q_{a^*b^*} =$

$q_{0\,a^*b^*} =$

$F_{a^*b^*} =$

$\delta_{a^*b^*} =$

Proposition 3.6 *the canonical DA for L recognizes L.*

Proof: Let C_L be the canonical DA for L. We prove $L(C_L) = L$.

Let $w \in \Sigma^*$. We prove by induction on $|w|$ that $w \in L$ iff $w \in L(C_L)$.

$$\begin{array}{lll} & \varepsilon \in L & (w = \epsilon) \\ \Leftrightarrow & L \in F_L & (\text{definition of } F_L) \\ \Leftrightarrow & q_{0L} \in F_L & (q_{0L} = L) \\ \Leftrightarrow & \varepsilon \in L(C_L) & (q_{0L} \text{ is the initial state of } C_L) \end{array}$$

$$\begin{array}{lll} & aw' \in L & \\ \Leftrightarrow & w' \in L^a & (\text{definition of } L^a) \\ \Leftrightarrow & w' \in L(C_{L^a}) & (\text{induction hypothesis}) \\ \Leftrightarrow & aw' \in L(C_L) & (\delta_L(L, a) = L^a) \end{array}$$

Theorem 3.7 *If L is regular, then C_L is the unique minimal DFA up to isomorphism recognizing L.*

Proof:

1. C_L is a DFA for L with a minimal number of states.
   - C_L has as many states as residuals, and
   - every DFA for L has at least as many states as residuals
2. Every minimal DFA for L is isomorphic to C_L.
   Let A be an arbitrary minimal DFA for L. Then:
   - The states of A are in bijection with the residuals of L.
   - The transitions of A are completely determined by this bijection: if $q \leftrightarrow L^w$, then $\delta(q, a) \leftrightarrow L^{wa}$
   - The initial state is the state in bijection with L.
   - The final states are those in bijection with residuals containing ϵ.

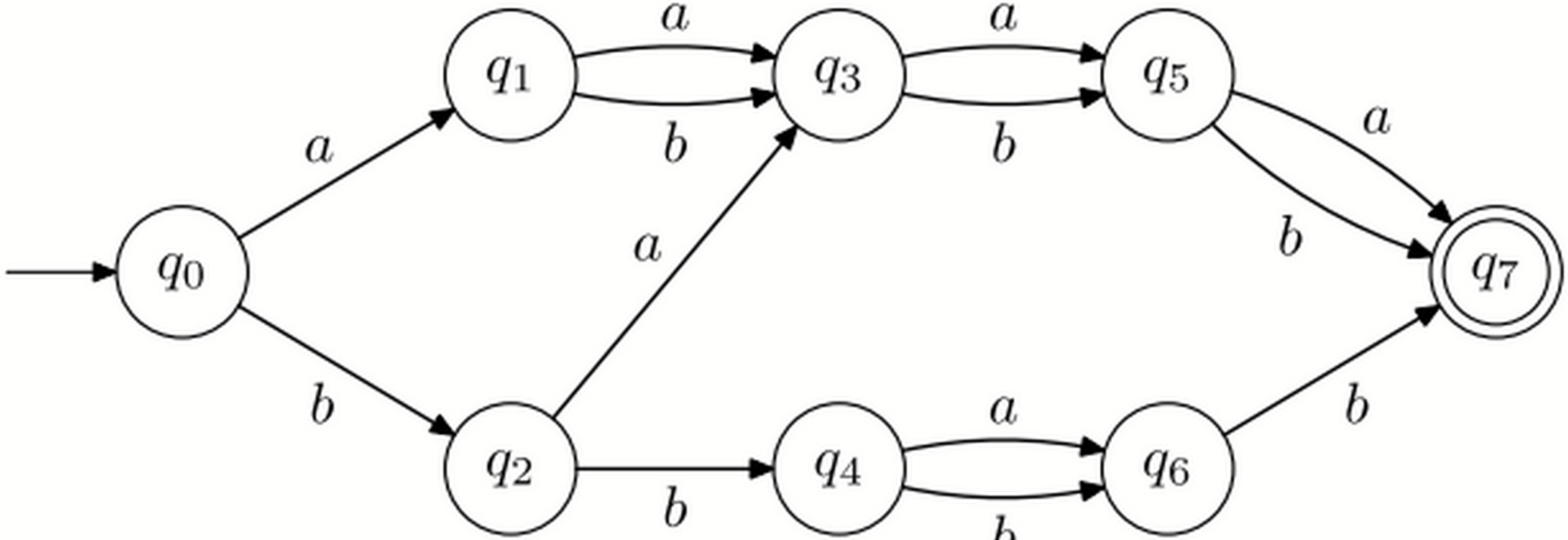
Corollary 3.8 *A DFA is minimal if and only if* $L(q) \neq L(q')$ *for every two distinct states* q *and* q'.

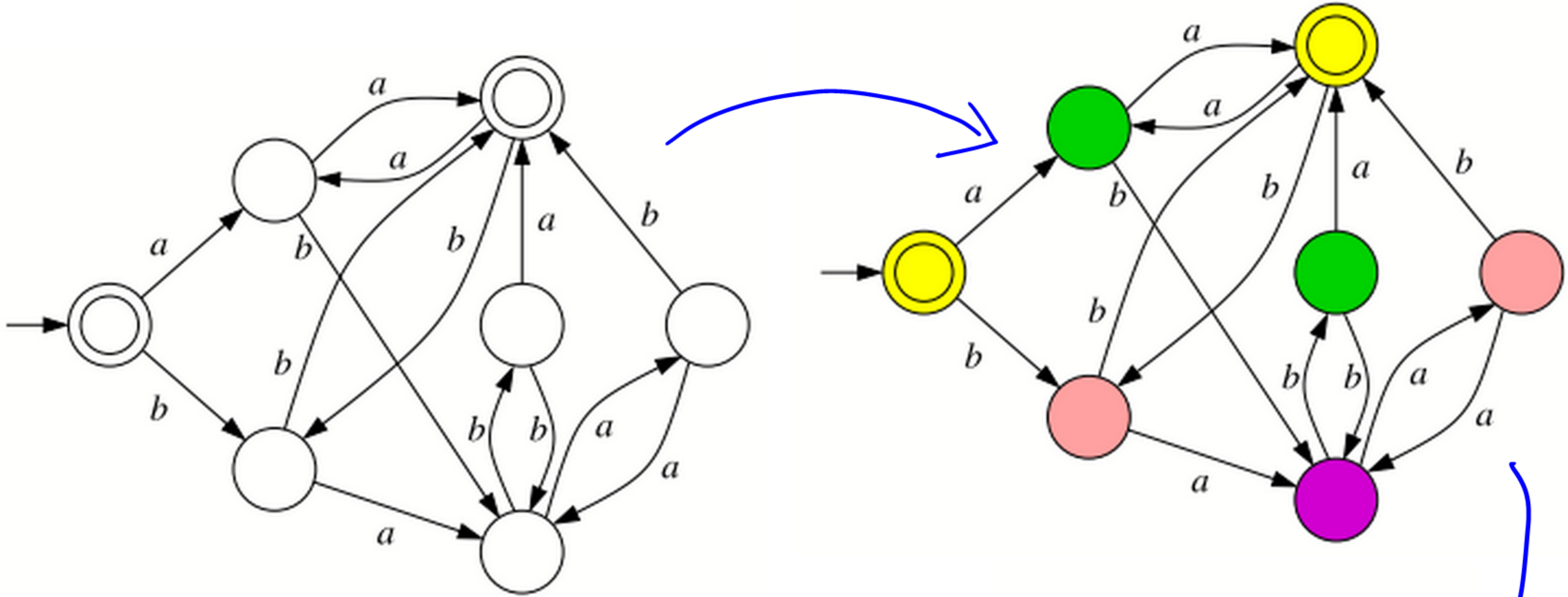
Proof:

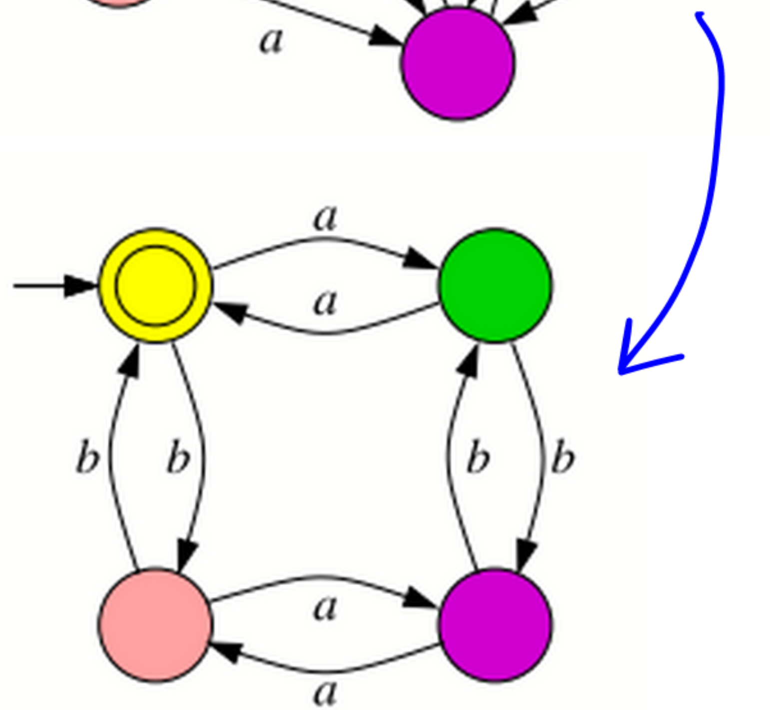
(→): Let A be a minimal DFA.
Every residual of $L(A)$ is recognized by at least one state of A (holds for every DFA).
Since A is minimal, it has as many states as C_L, and so its number of states is equal to the number of residuals of $L(A)$.
Therefore: distinct states of A recognize distinct residuals of $L(A)$.

(→): Let A be a DFA such that distinct states recognize distinct languages.
Since every state of A recognizes a residual of $L(A)$, and every residual of $L(A)$ is recognized by some state of A (holds for every DFA), the number of states of A is equal to the number of residuals of $L(A)$.
So A has as many states as C_L, and so it is minimal.

Is it minimal?







1. Computing the language partition
2. Quotienting
3. Thm: the result is the canonical DFA

Computing the language partition

State partitions

- Block: set of states.
- Partition: set of blocks such that each state belongs to exactly one block.
- Partition P refines partition P' if every block of P is contained in some block of P'.
- If P refines P', then we say that P is finer than P', and P' is coarser than P.
- Language partition: the partition in which two states belong to the same block iff they recognize the same language.

Computing the language partition

- Start with the partition containing the following two blocks:
  - Final states (which recognize ε)
  - Non-final states (which do not recognize ε)
- Iteratively split blocks, ensuring that states recognizing the same language always stay in the same block.
- Blocks that contain at least two states recognizing different languages are called unstable.

Finding an unstable block

If two states q_1, q_2 belong to the same block B but there is a letter a such that $\delta(q_1, a)$ and $\delta(q_2, a)$ belong to different blocks, then B is unstable.

Splitting an unstable block

If two states q_1, q_2 belong to the same block B and some block B' contains $\delta(q_1, a)$ but not $\delta(q_2, a)$, then B is unstable and we say that (a, B') splits B.

$Ref_P(B, a, B')$ denotes the result of splitting block B of partition P into the two following parts:

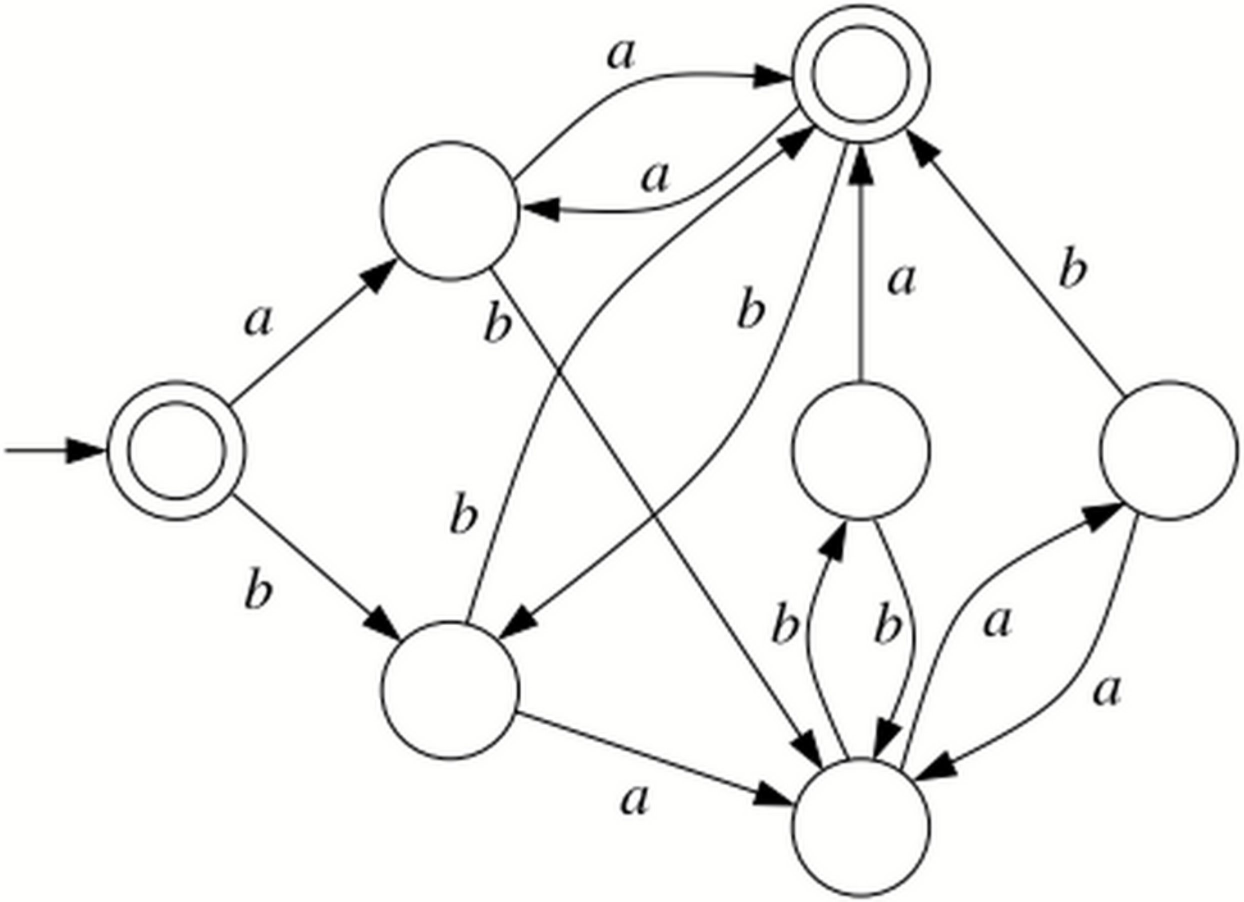
- States whose a-transition leads to B' (e.g. q_1)
- States whose a-transition leads to elsewhere (e.g. q_2)

LanPar(A)

Input: DFA $A = (Q, \Sigma, \delta, q_0, F)$

Output: The language partition P_ℓ for $L = \mathcal{L}(A)$.

if $F = \emptyset$ or $Q \setminus F = \emptyset$ **then return** $\{Q\}$
else $P \leftarrow \{F, Q \setminus F\}$
while P is unstable **do**
    pick $B, B' \in P$ and $a \in \Sigma$ such that (a, B') splits B
    $P \leftarrow Ref_P[B, a, B']$
return P


a
a
a
b
b
a
b
b
b
b
b
a
a
a

Correctness

- The algorithm terminates.

  Every execution of the loop increases the number of blocks by 1, and the number of blocks is bounded by the number of states.

- After termination, two states belong to the same block iff they recognize the same language.

  We show:

  (1) If two states belong to different blocks, they recognize different languages.

  (2) If two states recognize different languages, they belong to different blocks.

(1) If two states q_1 and q_2 belong to different blocks, they recognize different languages.

By induction on the number k of splittings carried out until q_1 and q_2 are put into different blocks.

- $k = 0$. Then q_1 is final and q_2 is non-final, or vice versa, and we are done.
- $k \rightarrow k + 1$. Then there are states q_1', q_2' such that $q_1 \xrightarrow{a} q_1'$, $q_2 \xrightarrow{a} q_2'$, and q_1', q_2' already belong to different blocks before q_1, q_2 are put into different blocks. By induction hypothesis q_1' and q_2' recognize different languages. Since the automaton is a DFA, q_1 and q_2 also recognize different languages

(2) If two states q_1 and q_2 recognize different languages, they belong to different blocks.

Let w be a shortest word that belongs to, say, $L(q_1)$ but not to $L(q_2)$. By induction on the length of w.

- $|w| = 0$. Then $w = \varepsilon$, q_1 is final, and q_2 is non-final. So q_1 and q_2 belong to different blocks from the start.
- $|w| > 0$. Then $w = aw'$ for some a, w'. Let $q_1' = \delta(q_1, a)$ and $q_2' = \delta(q_2, a)$. Then $L(q_1') \neq L(q_2')$ by the DFA property. By induction hypothesis q_1', q_2' are put at some some point into different blocks. If at this point q_1 and q_2 still belong to the same block B, then B becomes unstable and is eventually split.

Quotienting

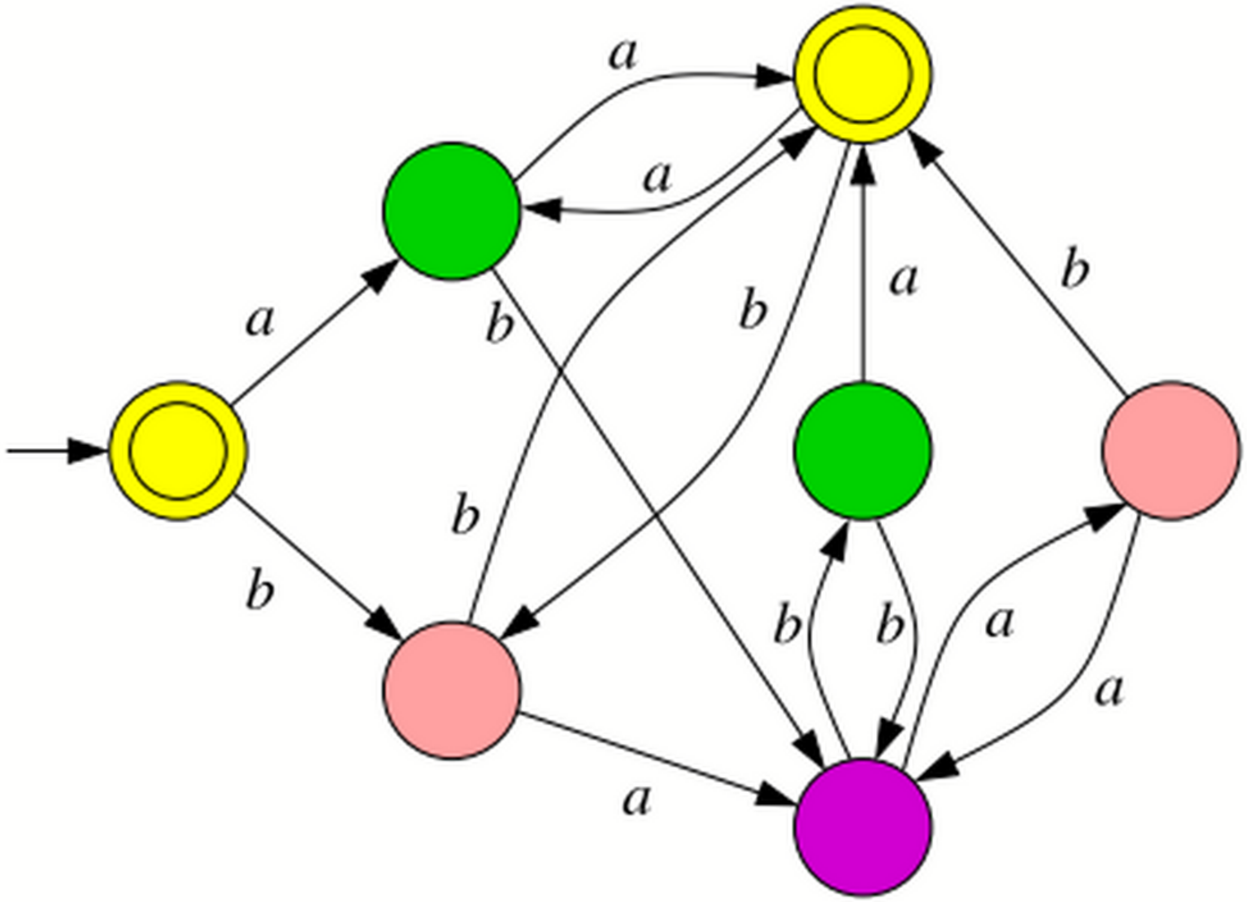
Quotient w.r.t. a partition

- Definition: The quotient of a NFA $A = (Q, \Sigma, \delta, q_0, F)$ with respect to a partition P is the NFA

$$A/P=(Q_P, \Sigma, \delta_P, q_{0P}, F_P)$$

where

  - $Q_P = P$
  - $(B, a, B') \in \delta_P$ iff $(q, a, q') \in \delta$ for some $q \in B$ and some $q' \in B'$
  - q_{0P} is the blcok containing q_0
  - F_P is the set of blocks that contain some state of F


a
a
a
b
b
a
b
b
b
b
b
a
a
a

We prove:

The quotient of a DFA with respect to its language partition is the canonical DFA.

The proof has two parts:

(1) A DFA and its quotient w.r.t. the language partition recognize the same language.
(2) The quotient is minimal (and therefore the canonical DFA).

(1) A DFA and its quotient w.r.t. the language partition recognize the same language.

We prove a more general result:

Lemma: Let A be a NFA, and let P be any partition that refines the language partition P_l.

a) For every state q of A: $L_A(q) = L_{A/P}(B)$, where B is the block containing q.
b) If A is a DFA and $P = P_l$, then A/P is also a DFA.

a) For every state q of A: $L_A(q) = L_{A/P}(B)$,

where B is the block containing q.

We prove that for every word $w \in \Sigma$:

$$w \in L_A(q) \Leftrightarrow w \in L_{A/P}(B).$$

Proof by induction on $|w|$.

Base: $|w| = 0$, i.e., $w = \varepsilon$. Then:

$$\begin{array}{lll} & \varepsilon \in L_A(q) & \\ \text{iff} & q \in F & \\ \text{iff} & B \subseteq F & (P \text{ refines } P_\ell\text{, and so also } P_0) \\ \text{iff} & B \in F_P & \\ \text{iff} & \varepsilon \in L_{A/P}(B) & \end{array}$$

a) For every state q of A: $L_A(q) = L_{A/P}(B)$, where B is the block containing q.

Step: $|w| > 0$. Then $w = aw'$ for some a, w'.

So $w \in L_A(q)$ iff there is $(q, a, q') \in \delta$ with $w' \in L_A(q')$.

Let B' be the block containing q'. We have:

	$aw' \in L_A(q)$	
iff	$w' \in L_A(q')$	(definition of q')
iff	$w' \in L_{A/P}(B')$	(induction hypothesis)
iff	$aw' \in L_{A/P}(B)$	($(B, a, B') \in \delta_P$)

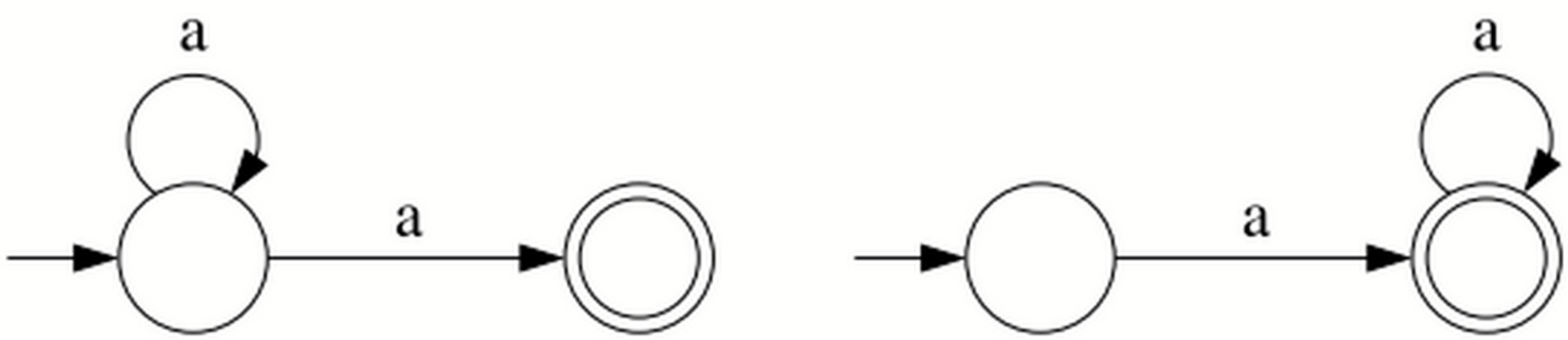
b) If A is a DFA and $P = P_l$, then A/P is also a DFA.

We show: If A/P has transitions (B, a, B_1) and (B, a, B_2), then $B_1 = B_2$.

- There are $q, q' \in B$, $q_1 \in B_1$, $q_2 \in B_2$ such that (q, a, q_1) and (q', a, q_2) are transitions of A.
- Since $P = P_l$, q and q' recognize the same language.
- Since A is a DFA, q_1 and q_2 recognize the same language.
- Since $P = P_l$, $B_1 = B_2$.

Reduction of NFAs

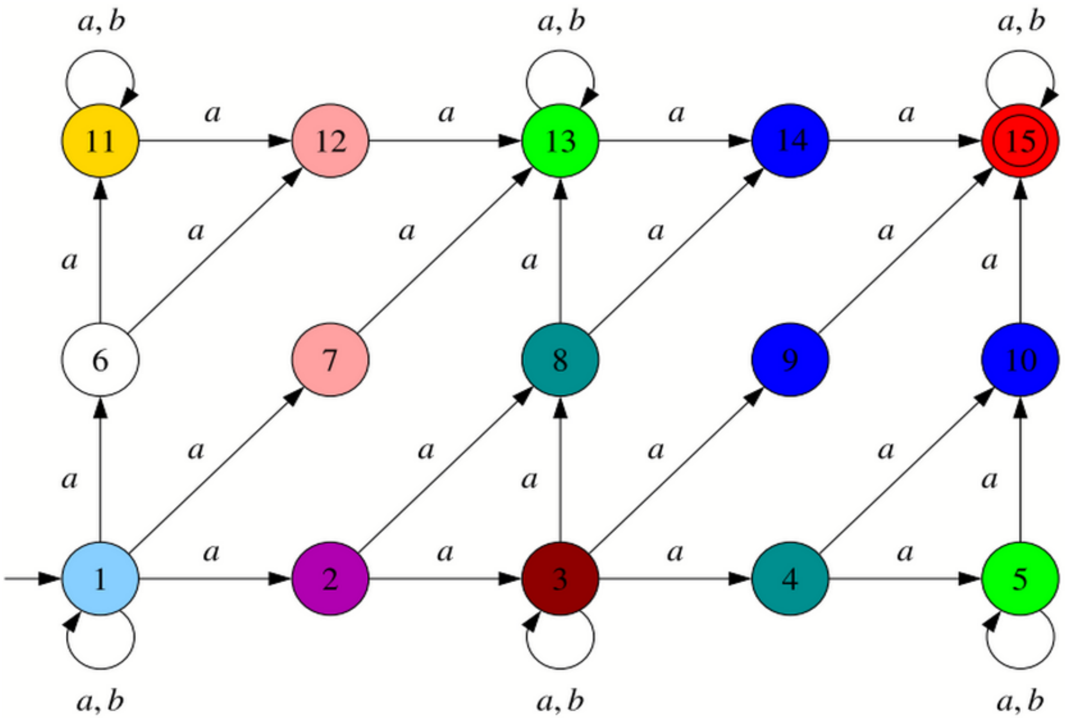
The minimal NFA is not unique

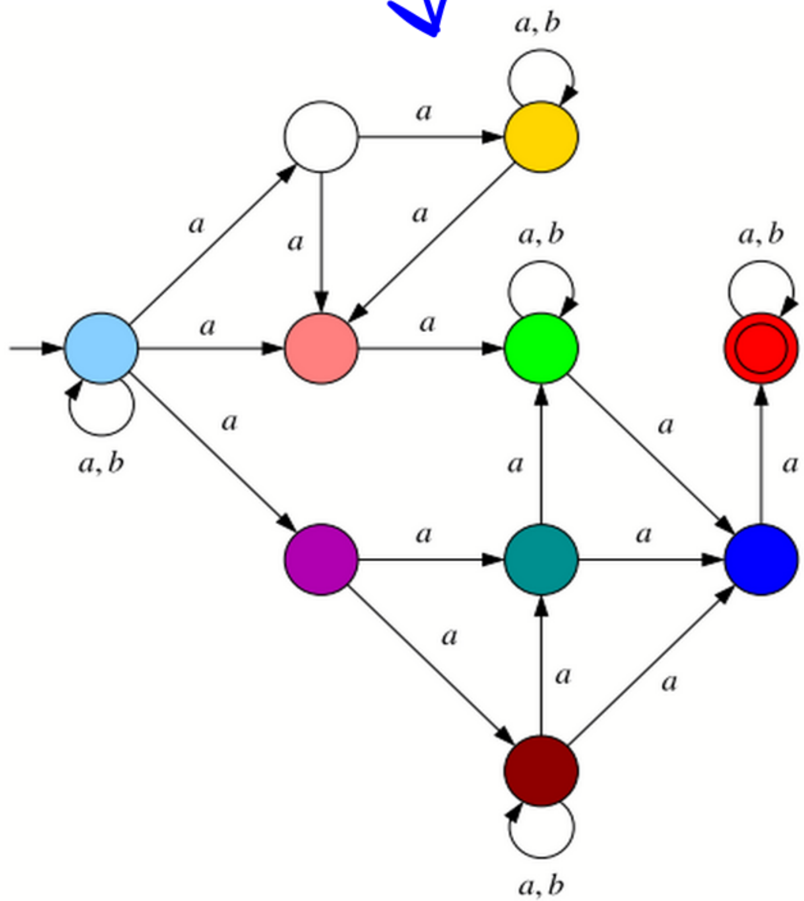


Minimal NFAs are hard to compute

Theorem 3.18 *The following problem is PSPACE-complete: given a NFA A and a number k* ≥ 1, *decide if there is a NFA equivalent to A with at most k states.*

Reducing NFAs





1. Computing a suitable partition
2. Quotienting

What is a „suitable“ partition?

- The quotient w.r.t. the partition must recognize the same language as the original NFA.
- So, by the Lemma, we can take any partition that refines the language partition.
- A partition refines the language partition iff states in the same block recognize the same language (states in different blocks may not recognize different langauges, though!).
- Such partitions necessarily refine the partition $\{F, Q \setminus F\}$.

Computing a suitable partition

- Idea: use the same algorithm as for DFA, but with new notions of unstable block and block splitting.
- We must guarantee:

  after termination, states of a block recognize the same language

  or, equivalently

  after termination, states recognizing different languages belong to different blocks

Key observation:

If $L(q_1) \neq L(q_2)$ then either

- one of q_1, q_2 is final and the other non-final, or
- one of q_1, q_2, say q_1, has a transition $q_1 \overset{\text{a}}{\rightarrow} q_1'$ such that every a-transition $q_2 \overset{\text{a}}{\rightarrow} q_2'$ satisfies: $L(q_1') \neq L(q_2')$.

This suggests the following definition:

Definition: Let B, B' blocks of a partition P, and let $a \in \Sigma$.
The pair (a, B') splits B if there are states $q_1, q_2 \in B$ such that

$$\delta(q_1, a) \cap B' = \emptyset \quad \text{and} \quad \delta(q_2, a) \cap B' \neq \emptyset$$

The result of the split is the partition

$$Ref_P^{NFA}[B, a, B] = (P \setminus \{B\}) \cup \{B_0, B_1\}$$

where

$$B_0 = \{q \in B \mid \delta(q, a) \cap B' = \emptyset\}$$
$$B_1 = \{q \in B \mid \delta(q, a) \cap B' \neq \emptyset\}$$

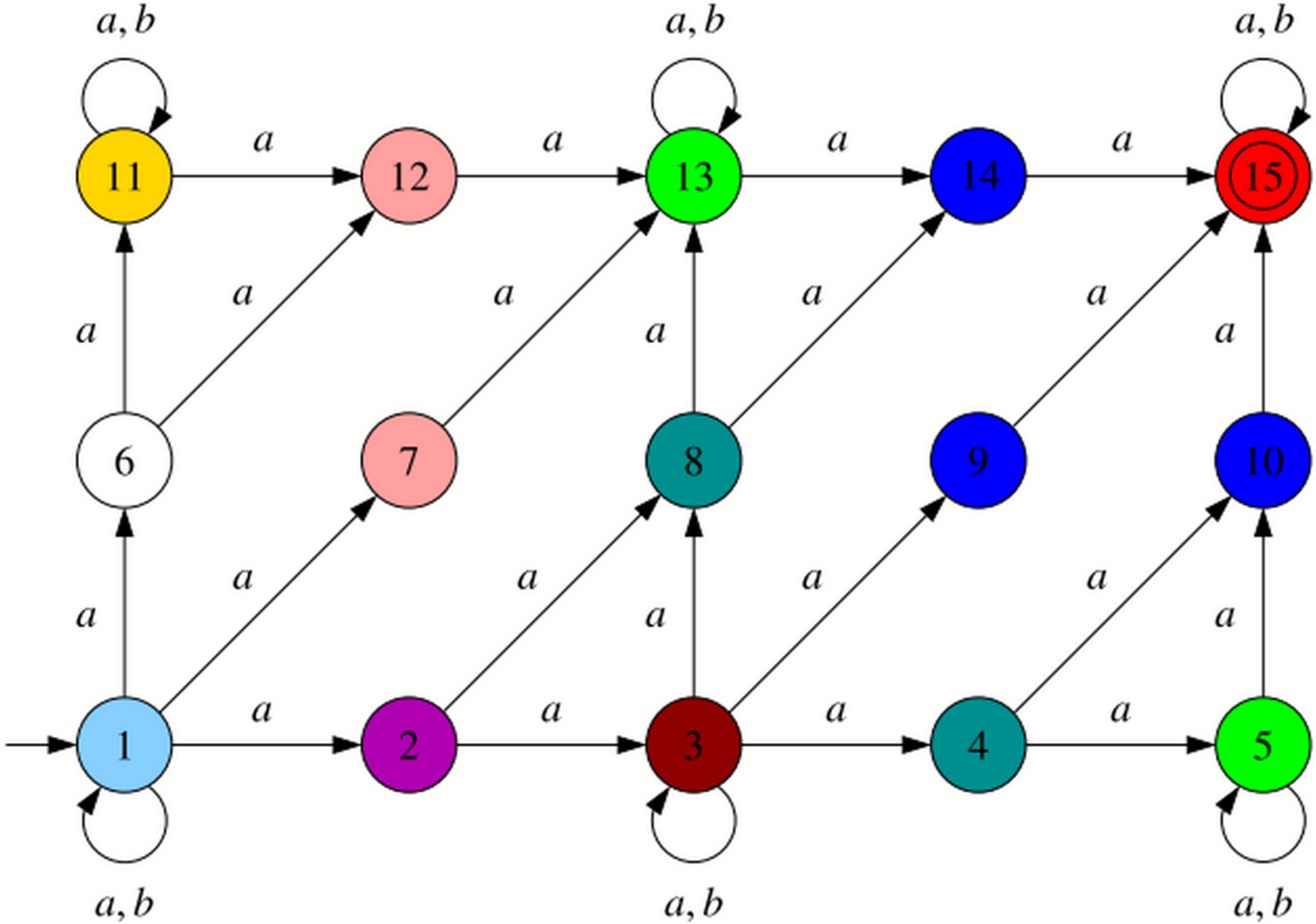
A partition is unstable if there are B, a, B' such that (a, B') splits B , otherwise it is stable.

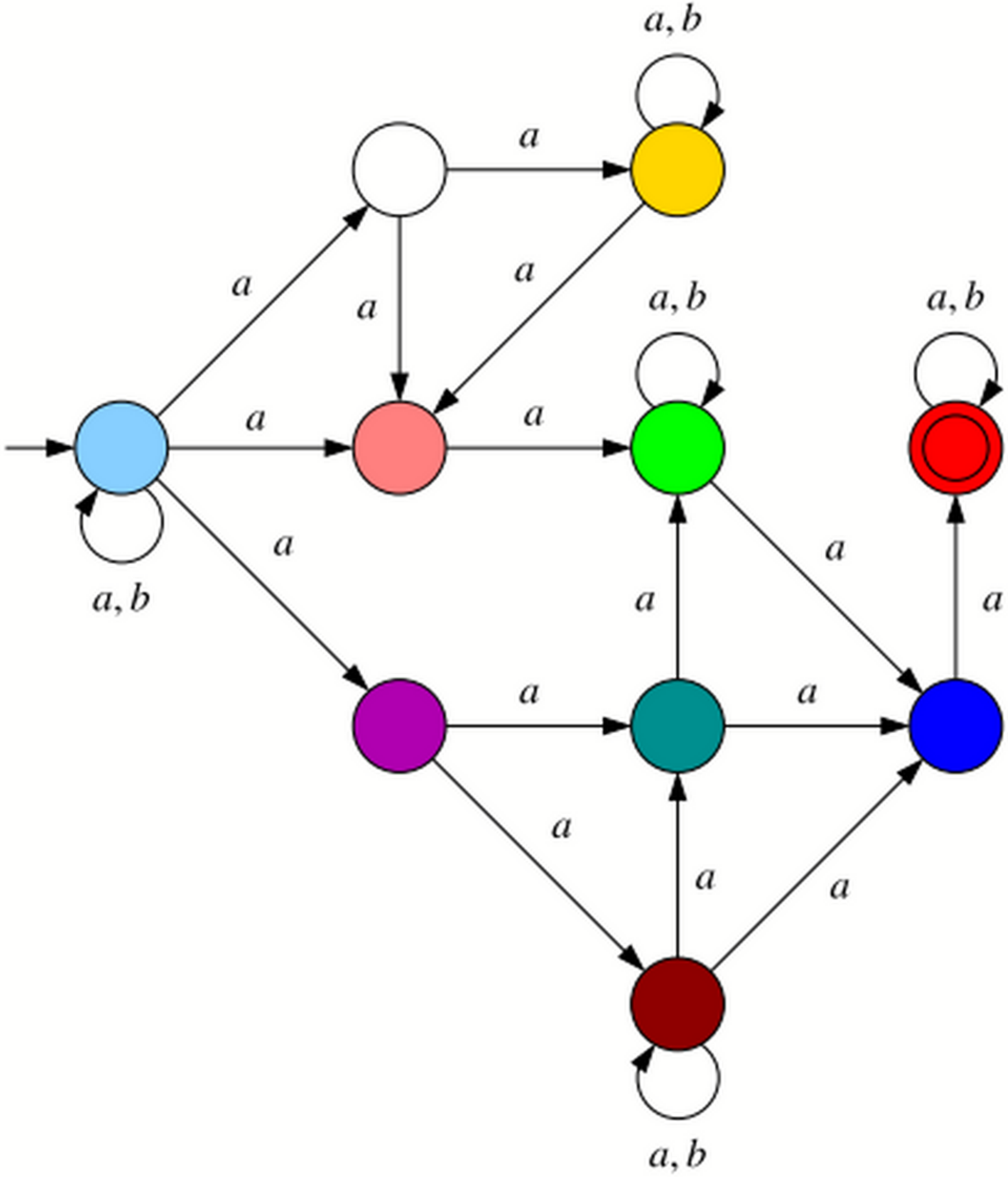
CSR(*A*)

Input: NFA $A = (Q, \Sigma, \delta, q_0, F)$

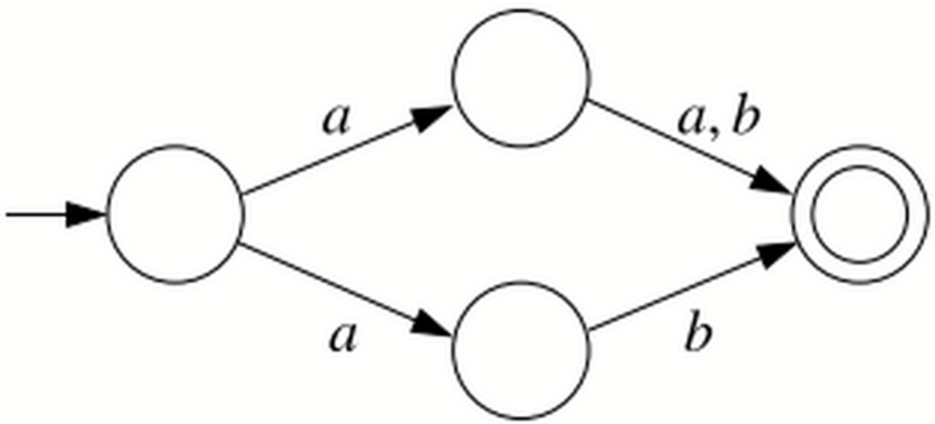
Output: The partition *CSR*.

if $F = \emptyset$ or $Q \setminus F = \emptyset$ **then return** $\{Q\}$
else $P \leftarrow \{F, Q \setminus F\}$
while P is unstable **do**
    pick $B, B' \in P$ and $a \in \Sigma$ such that (a, B') splits B
    $P \leftarrow Ref_P^{NFA}[B, a, B']$
return P

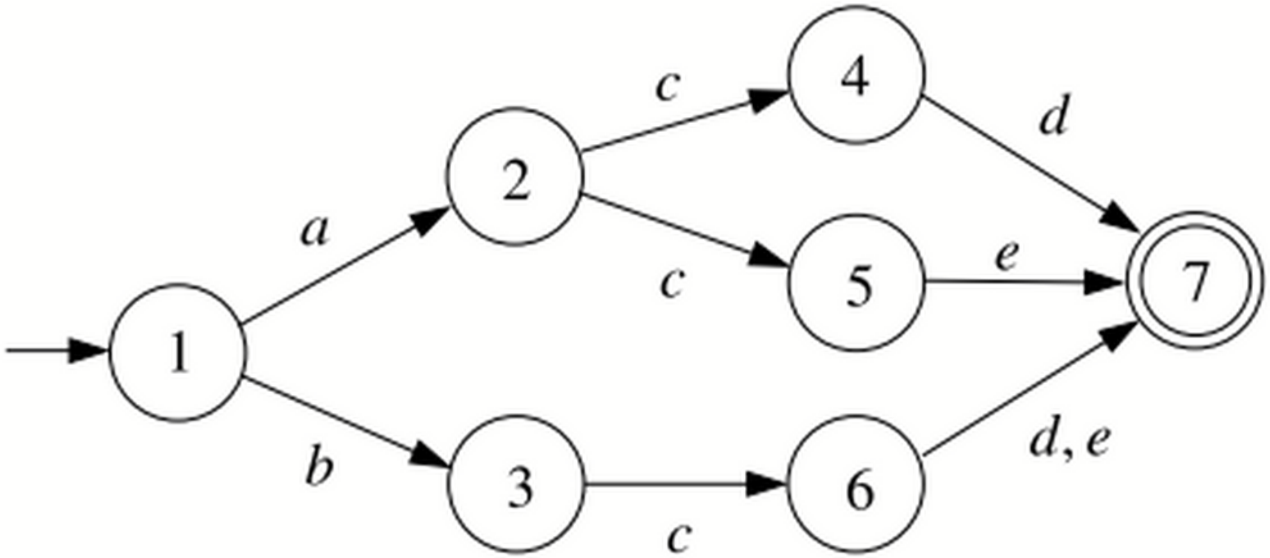

a, b
a, b
a, b
11
12
13
14
15
6
7
8
9
10
1
2
3
4
5
a
a, b
a, b
a, b


a, b
a
a
a
a
a, b
a, b
a
a
a
a, b
a
a
a
a
a
a
a
a
a, b

Reduction may not minimize



The algorithm does not compute the language partition



Non-regular languages