# CS 6041
# Theory of Computation

## Regular language

### Kun Suo

Computer Science, Kennesaw State University

https://kevinsuo.github.io/

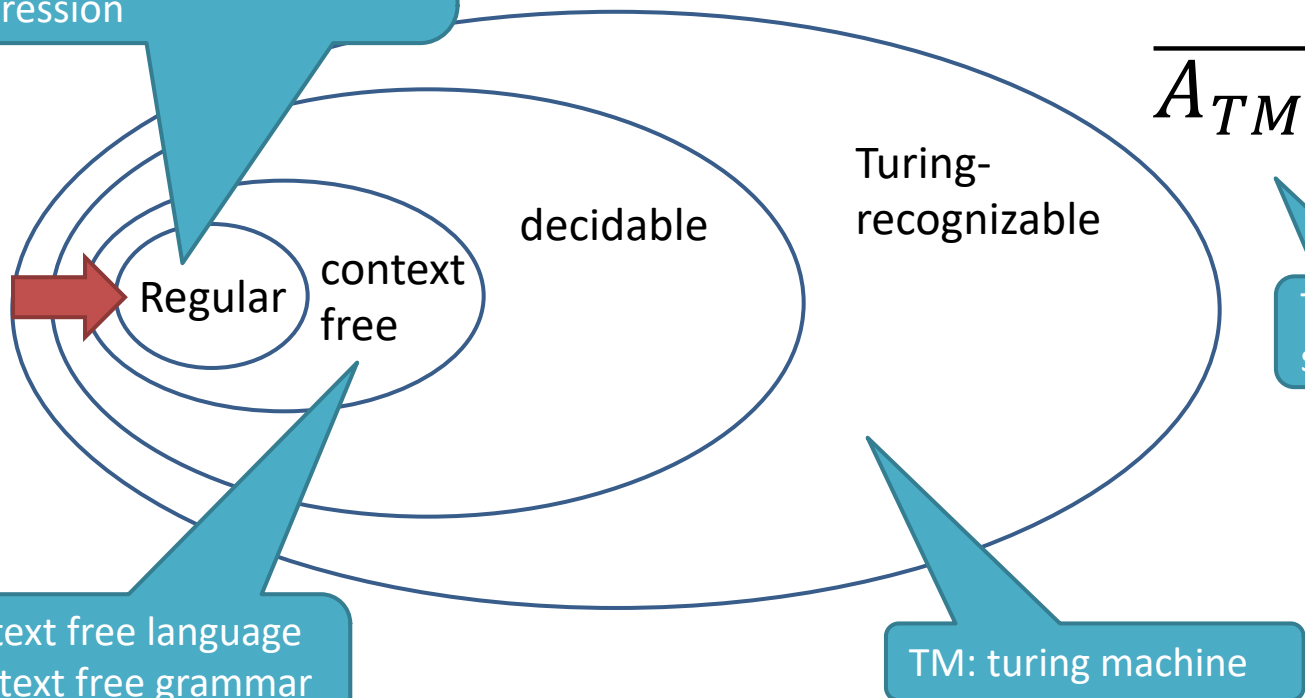# Where are we now?

Regular language
DFA: deterministic finite automata
NFA: non-deterministic finite automata
RE: regular expression

$$\overline{A_{TM}}$$

Turing-recognizable

decidable

context free

Regular

TM cannot solve

CFL: context free language
CFG: context free grammar
PDA: push down automata

TM: turing machine

# Regular language

- A language is called a regular language if some finite automaton (DFA) recognizes it

- A language is called a regular language if some nondeterministic finite automaton (NFA) recognizes it

- Regular language:
  - L=L(M)
  - M is DFA or NFA

# Regular operations

| | DFA/NFA/RL | PDA | TM |
|---|---|---|---|
| **Union** | close | ? | ? |
| **Concatenation** | close | ? | ? |
| **Star** | close | ? | ? |
| **Complement** | close | ? | ? |
| **Boolean operation** | close | ? | ? |

# Closure under the union

- Theorem: regular language is closed under the union operation.

- Proof idea:

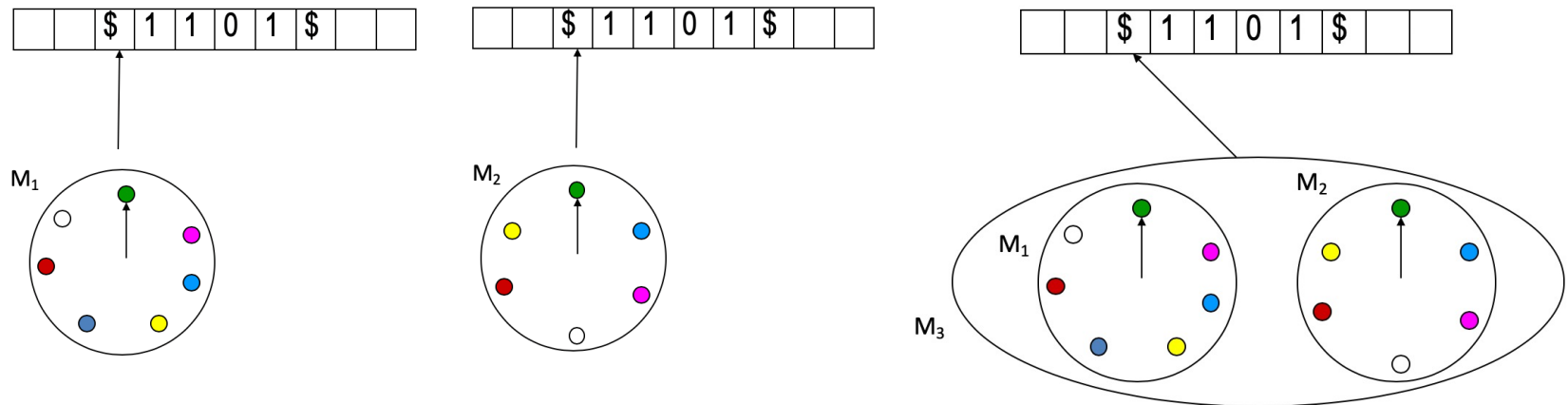  In the past, we prove this by creating a DFA. This time we prove it by creating a NFA.

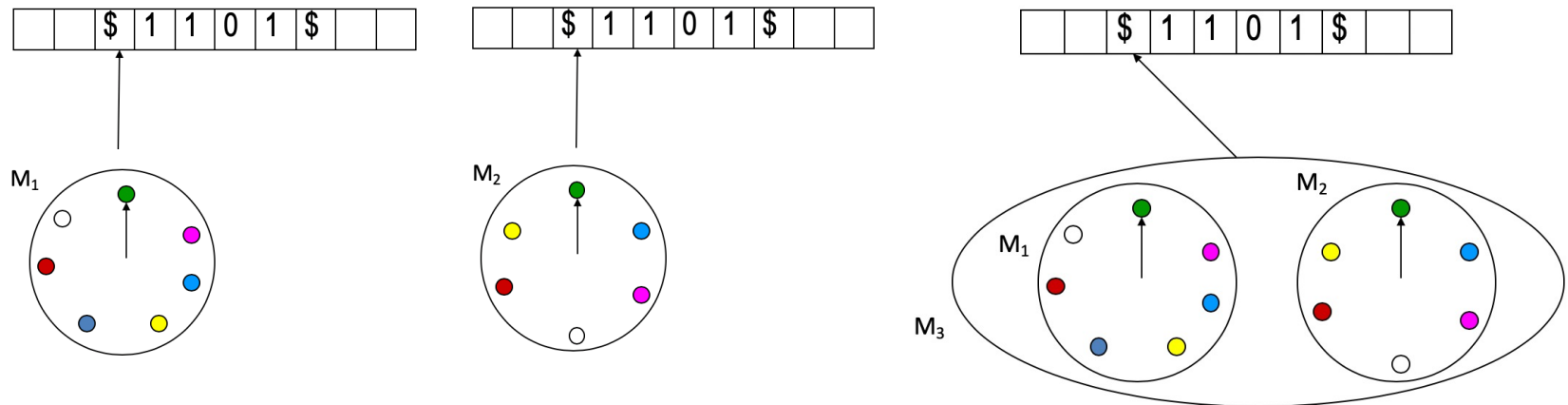# Closure under the union proved by DFA

- Proof:

  Let $L_i=L(M_i)$ is a regular language, $M_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$, i=1,2. We need to build a finite automata to recognize $L_1\cup L_2$

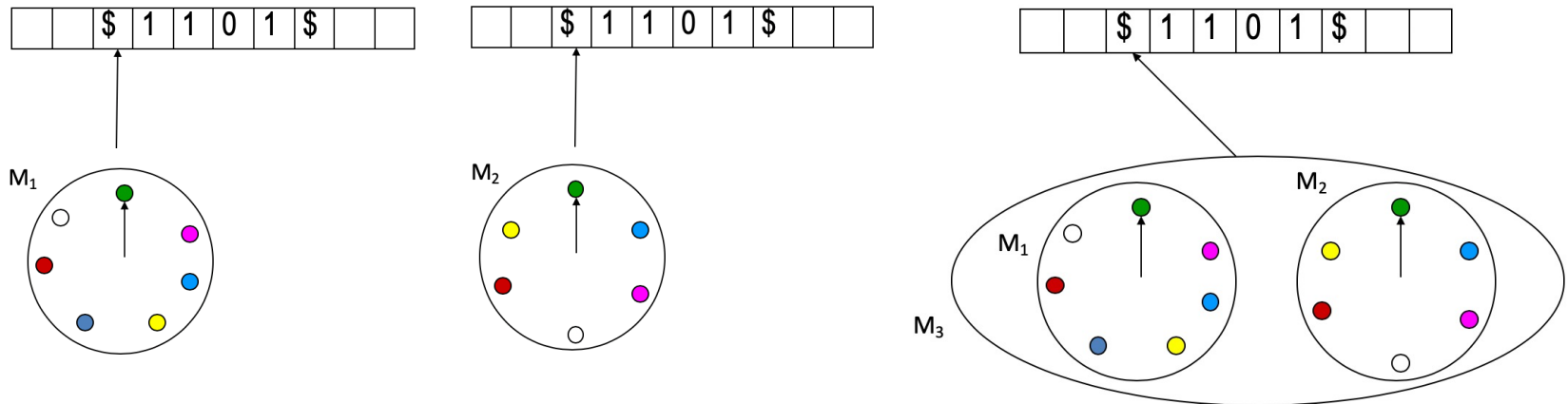  Build $M_3=(Q_3,\Sigma,\delta_3,q_3,F_3)$.
  $Q_3=Q_1\times Q_2$;

# Closure under the union proved by DFA

- Proof:

  Let $L_i$=$L(M_i)$ is a regular language, $M_i$=$(Q_i,\Sigma,\delta_i,q_i,F_i)$, i=1,2. We need to build a finite automata to recognize $L_1 \cup L_2$

  Build $M_3$=$(Q_3,\Sigma,\delta_3,q_3,F_3)$.
  $\delta_3((r_1,r_2),a)$=$(\delta_1(r_1,a),\delta_2(r_2,a))$;
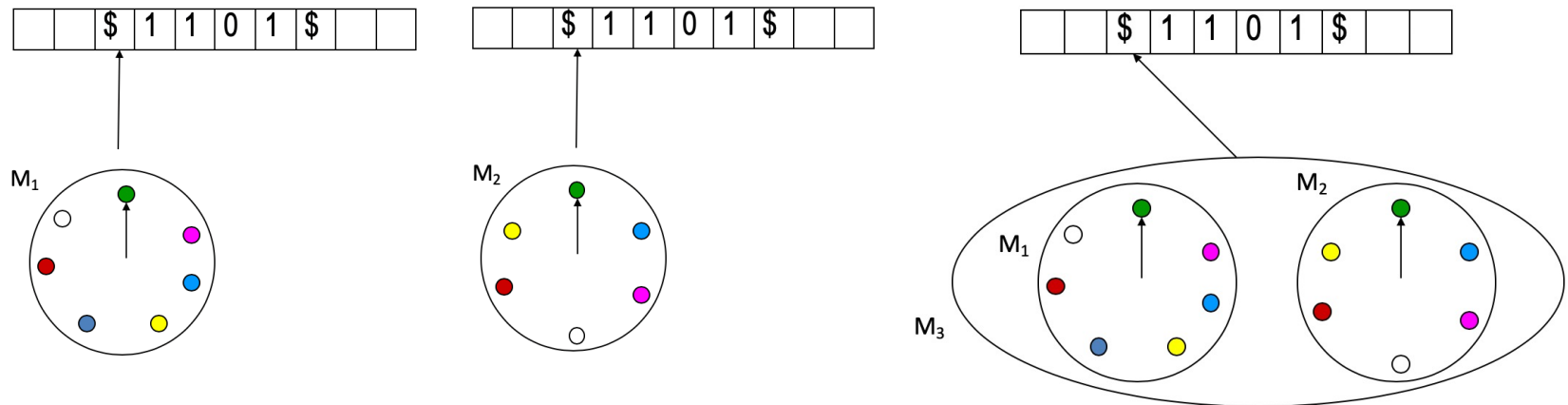
# Closure under the union proved by DFA

- Proof:

  Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, $i=1,2$. We need to build a finite automata to recognize $L_1 \cup L_2$

  Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.
  $q_3 = (q_1, q_2)$;

# Closure under the union proved by DFA

- Proof:

  Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, $i = 1, 2$. We need to build a finite automata to recognize $L_1 \cup L_2$

  Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.
  $F_3 = (F_1 \times Q_2) \cup (Q_1 \times F_2)$.

  > One accept state in M1 with all states in M2
  > One accept state in M2 with all states in M1

# Closure under the union proved by DFA

- Proof:

Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, $i=1,2$. We need to build a finite automata to recognize $L_1 \cup L_2$

Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.
  $Q_3 = Q_1 \times Q_2$;
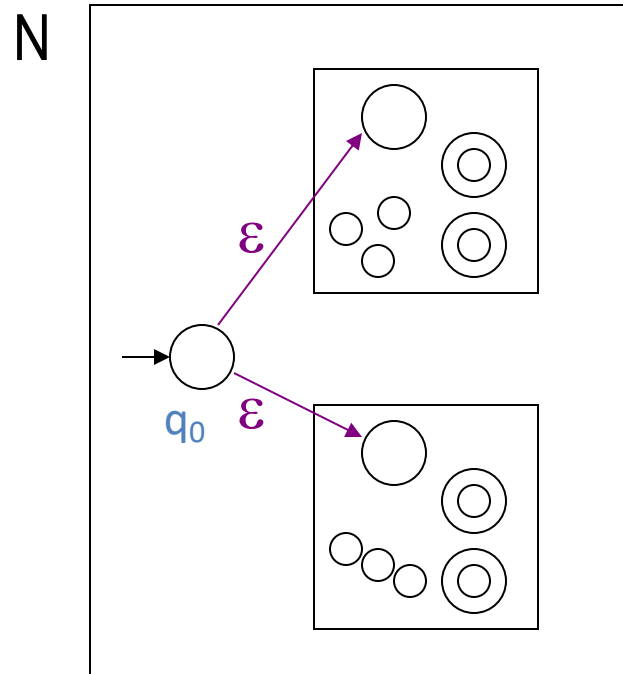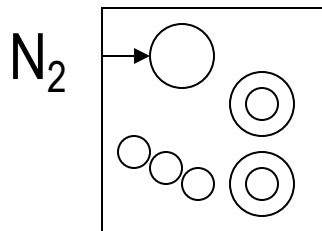  $\delta_3((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$;
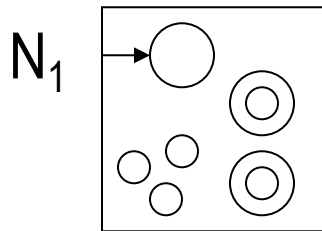  $q_3 = (q_1, q_2)$;
  $F_3 = (F_1 \times Q_2) \cup (Q_1 \times F_2)$.

$L(M_3) = L_1 \cup L_2$, so $L_1 \cup L_2$ is still regular language

# Closure under the union

- Theorem: regular language ($L_1$, $L_2$) is closed under the union operation ($L_1 \cup L_2$).

- Proof idea:

# Closure under the union

- Theorem: regular language is closed under the union operation.

- Proof:

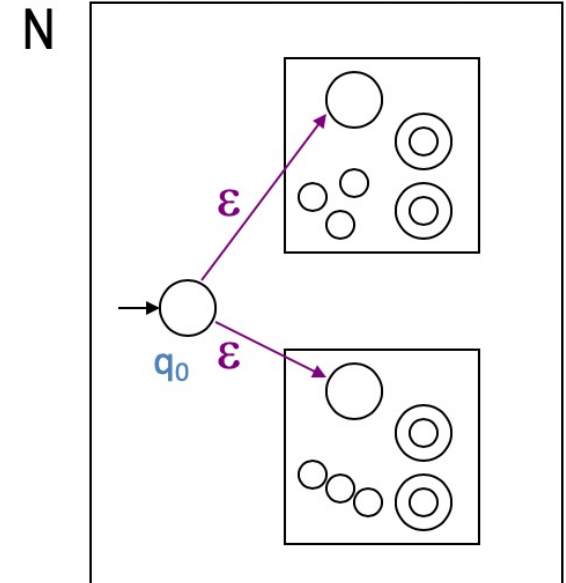  Let NFA $N_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$ recognize $A_i$, i=1,2.

  Create NFA

  $N=(Q,\Sigma,\delta,q_0,F)$ to recognize $A_1\cup A_2$

  Let $Q=Q_1\cup Q_2\cup\{q_0\}$

  $F=F_1\cup F_2$

$$\delta(q,a) = \begin{cases} \delta_1(q,a), & if\ q \in Q_1 \\ \delta_2(q,a), & if\ q \in Q_2 \\ \{q_1,q_2\}, & if\ q = q_0\ and\ a = \ \varepsilon \\ \emptyset, & if\ q = q_0\ and\ a \neq \ \varepsilon \end{cases}$$



N

# Closure under the concatenation

- Theorem: regular language is closed under the concatenation operation.

- Proof idea:

In the past, we prove this by creating a DFA. This time we prove it by creating a NFA.

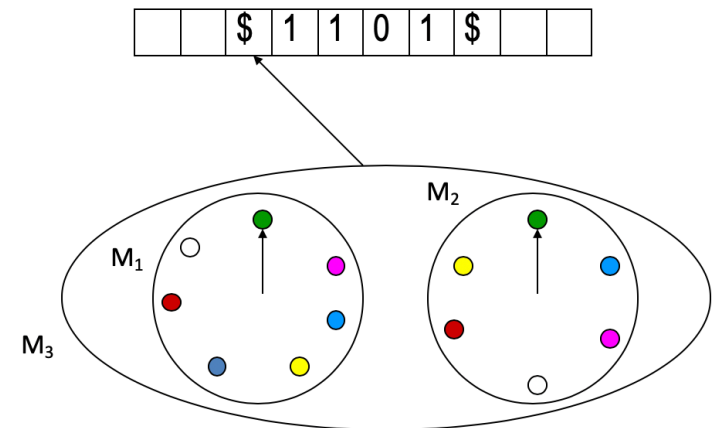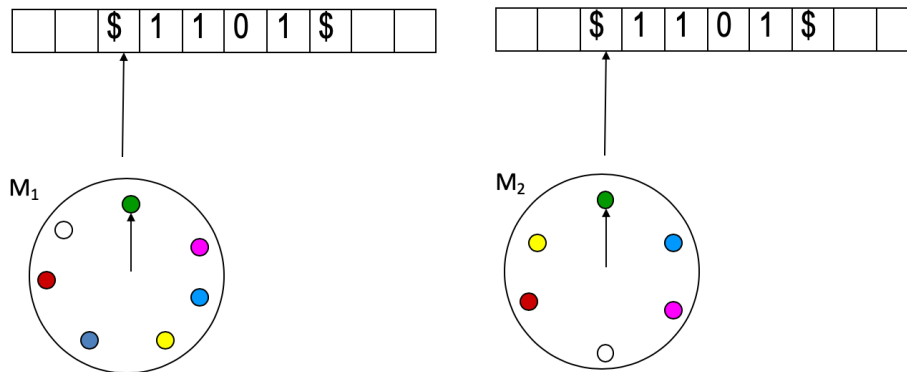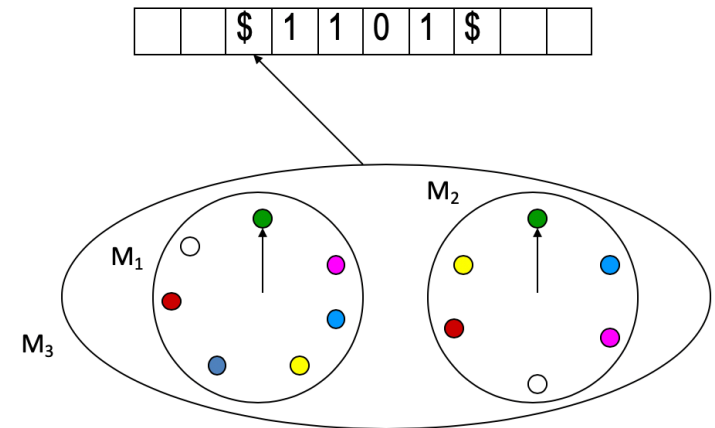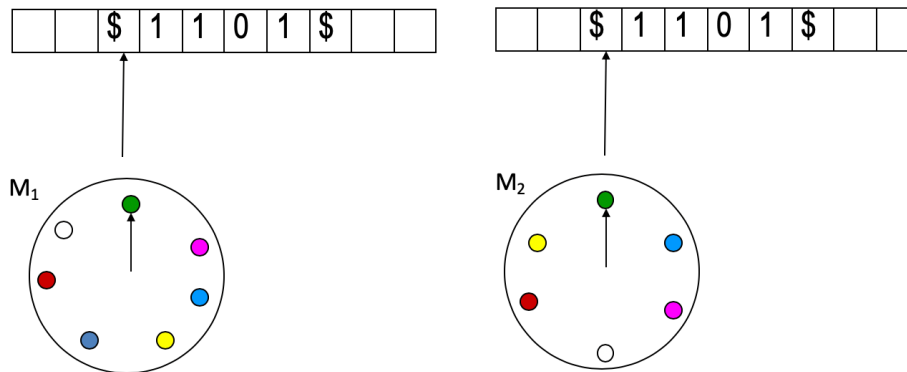# Closure under the concatenation proved by DFA

- Proof:

  Let $L_i=L(M_i)$ is a regular language, $M_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$, i=1,2. We need to build a finite automata to recognize $L_1 \cap L_2$

  Build $M_3=(Q_3,\Sigma,\delta_3,q_3,F_3)$.
  $Q_3=Q_1 \times Q_2$;

# Closure under the concatenation proved by DFA

- Proof:

Let $L_i=L(M_i)$ is a regular language, $M_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$, i=1,2. We need to build a finite automata to recognize $L_1 \cap L_2$

Build $M_3=(Q_3,\Sigma,\delta_3,q_3,F_3)$.
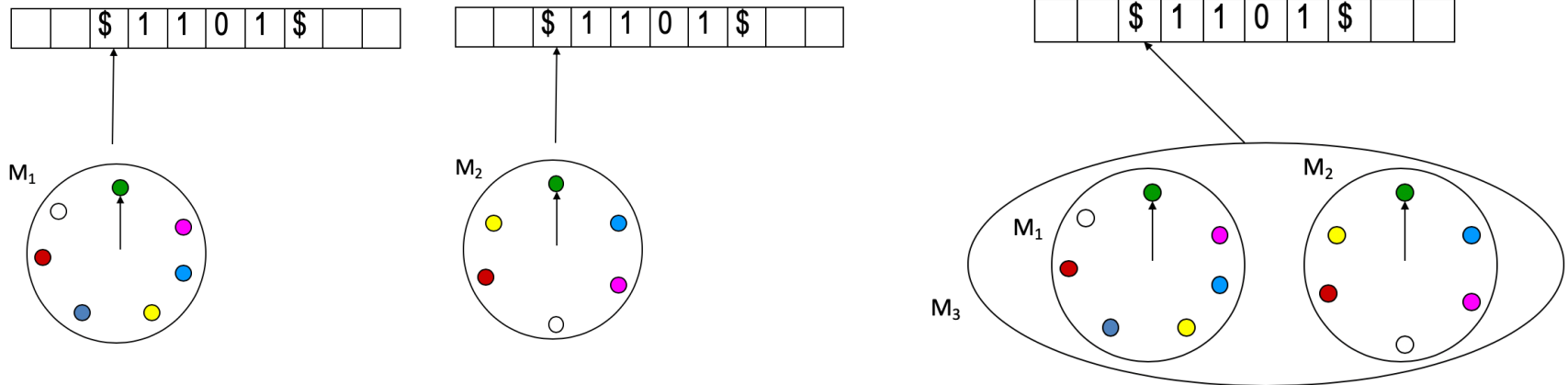$\delta_3((r_1,r_2),a)=(\delta_1(r_1,a),\delta_2(r_2,a))$;

# Closure under the concatenation proved by DFA

- ## Proof:

  Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, i=1,2. We need to build a finite automata to recognize $L_1 \cap L_2$

  Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.
  $q_3 = (q_1, q_2)$;
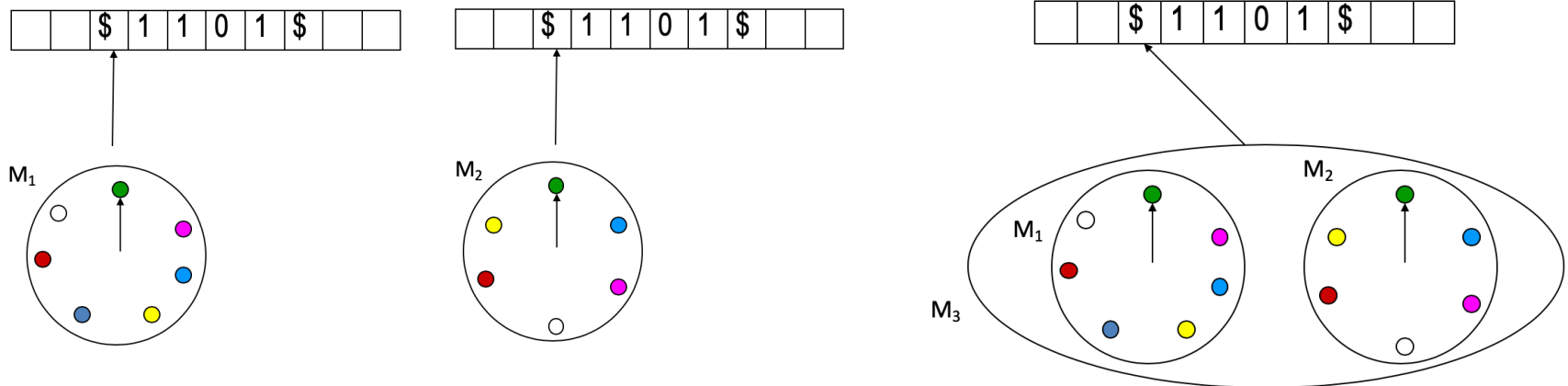
# Closure under the concatenation proved by DFA

- Proof:

  Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, i=1,2. We need to build a finite automata to recognize $L_1 \cap L_2$

  Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.

  $F_3 = F_1 \times F_2$.

One accept state in M1 with One accept state in M2

# Closure under the concatenation proved by DFA

- Proof:

Let $L_i = L(M_i)$ is a regular language, $M_i = (Q_i, \Sigma, \delta_i, q_i, F_i)$, $i=1,2$. We need to build a finite automata to recognize $L_1 \cap L_2$

Build $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$.

$\quad Q_3 = Q_1 \times Q_2$;

$\quad \delta_3((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$;
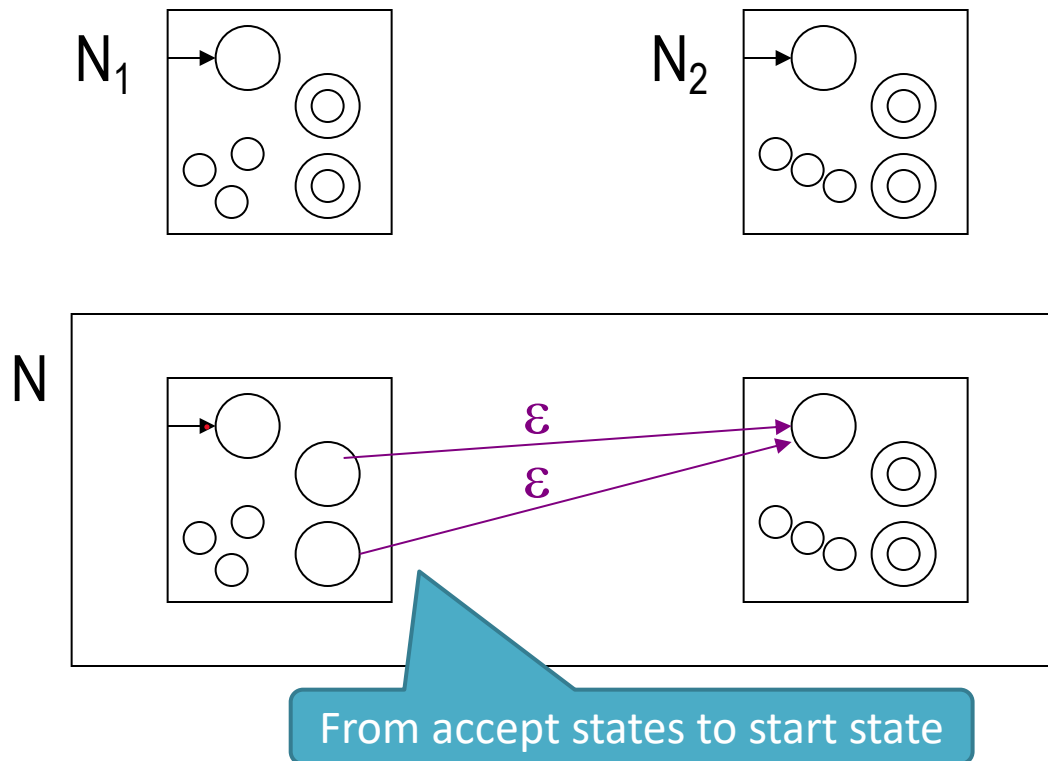
$\quad q_3 = (q_1, q_2)$;

$\quad F_3 = F_1 \times F_2$.

$L(M_3) = L_1 \cap L_2$, so $L_1 \cap L_2$ is still regular language

# Closure under the concatenation

- Theorem: regular language ($L_1$, $L_2$) is closed under the concatenation operation ($L_1 \cap L_2$).



From accept states to start state

# Closure under the concatenation

- Theorem: regular language is closed under the concatenation operation.

- Proof:

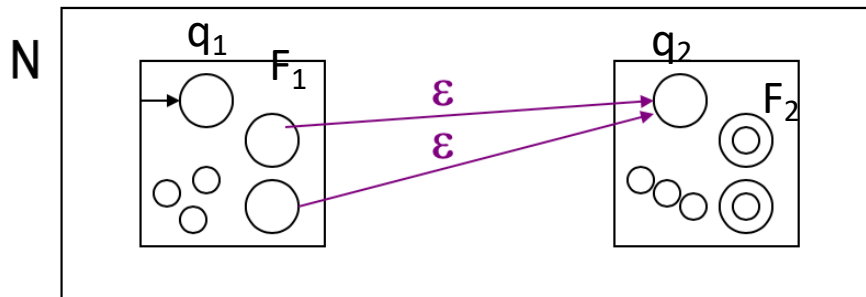    Let NFA $N_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$ recognize $A_i$, i=1,2.

    Create NFA

    $N=(Q,\Sigma,\delta,q_1,F)$ to recognize $A_1 \cap A_2$

    Let $Q=Q_1 \cup Q_2$

    $F=F_2$

$$\delta(q,a) = \begin{cases} \delta_1(q,a), & if\ q \in\ Q_1\ and\ q \notin\ F_1 \\ \delta_1(q,a), & if\ q \in\ F_1\ and\ a \neq \varepsilon \\ \delta_1(q,a) \cup \{q_2\} & if q \in\ F_1\ and\ a =\ \varepsilon \\ \delta_2(q,a), & if q\ \in\ Q_2 \end{cases}$$
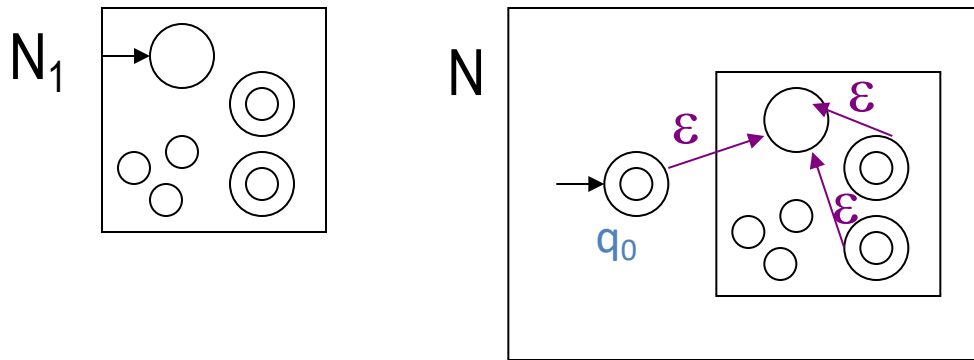
# Closure under the star

- Theorem: regular language is closed under the star operation.

$$A^* = A \times A \times \ldots \times A$$

i.e., $A = \{0,1\}$, $A^2 = \{\varepsilon, 1, 0, 00, 01, 10, 11\}$
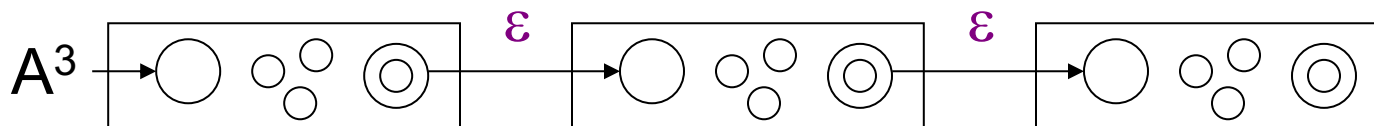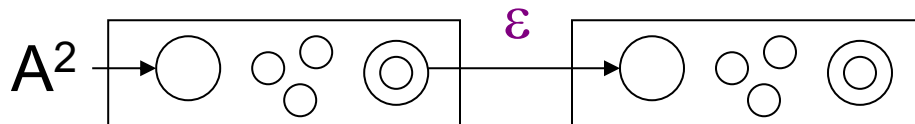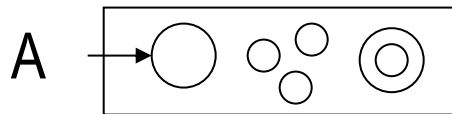


Can anyone explain this figure?

# Closure under the star

i.e., A={0,1},
$A^2$={ε,1,0, 00,01,10,11}

$A^2 = A \times A$

> 0 is computed on 1st A
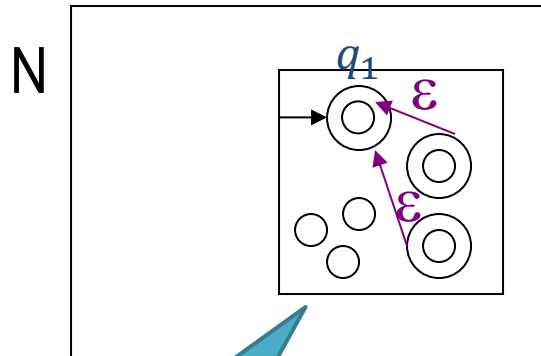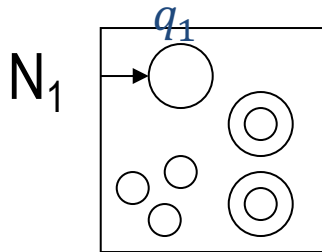> 1 is computed on 2nd A
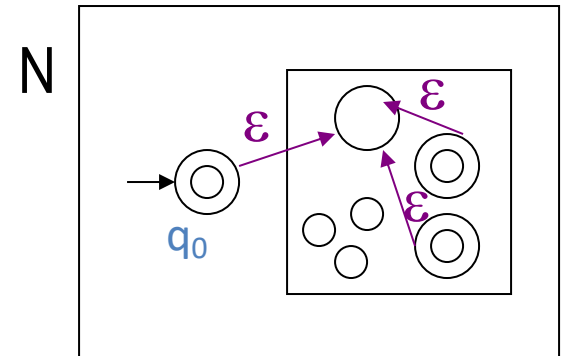
A

$A^2$ ──ε──>

$A^3$ ──ε──> ──ε──>

# Closure under the star

- Theorem: regular language is closed under the star operation.

$A^* = A \times A \times \dots \times A$

i.e., $A=\{0,1\}$, $A^2=\{\varepsilon,1,0,00,01,10,11\}$

$N_1$

$q_1$

$N$

$q_1$

$\varepsilon$

$\varepsilon$

$N$

$\varepsilon$

$\varepsilon$

$\varepsilon$

$q_0$

Is this correct? Why?

Wrong! Because the start state $q_1$ does not accept originally in $N_1$. It also should not accept in $N$

# Closure under the star

- Theorem: regular language is closed under the star operation.



- Proof:

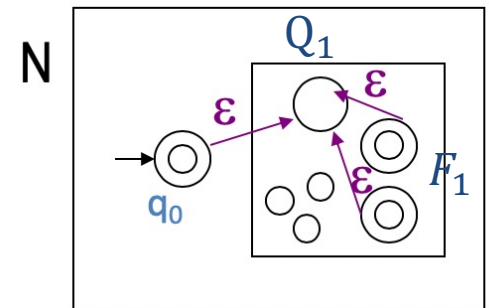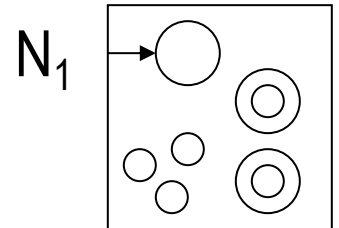  let $A_1 = L(N_1)$, $A_1^* = L(N)$, NFA $N_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$.

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



  $Q = Q_1 \cup \{q_0\}$;

  $F = F_1 \cup \{q_0\}$;

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

# Regular operations

| | DFA/NFA/RL | PDA | TM |
|---|---|---|---|
| **Union** | close | ? | ? |
| **Concatenation** | close | ? | ? |
| **Star** | close | ? | ? |
| **Complement** | close | ? | ? |
| **Boolean operation** | close | ? | ? |