Non-regular languages

(Pumping Lemma)

Non-regular languages

$$\{a^n b^n : \quad n \geq 0\}$$

$$\{vv^R : \quad v \in \{a,b\}^*\}$$

Regular languages

$$a^*b$$

$$b^*c + a$$

$$b + c(a+b)^*$$

$$etc...$$

How can we prove that a language L is not regular?

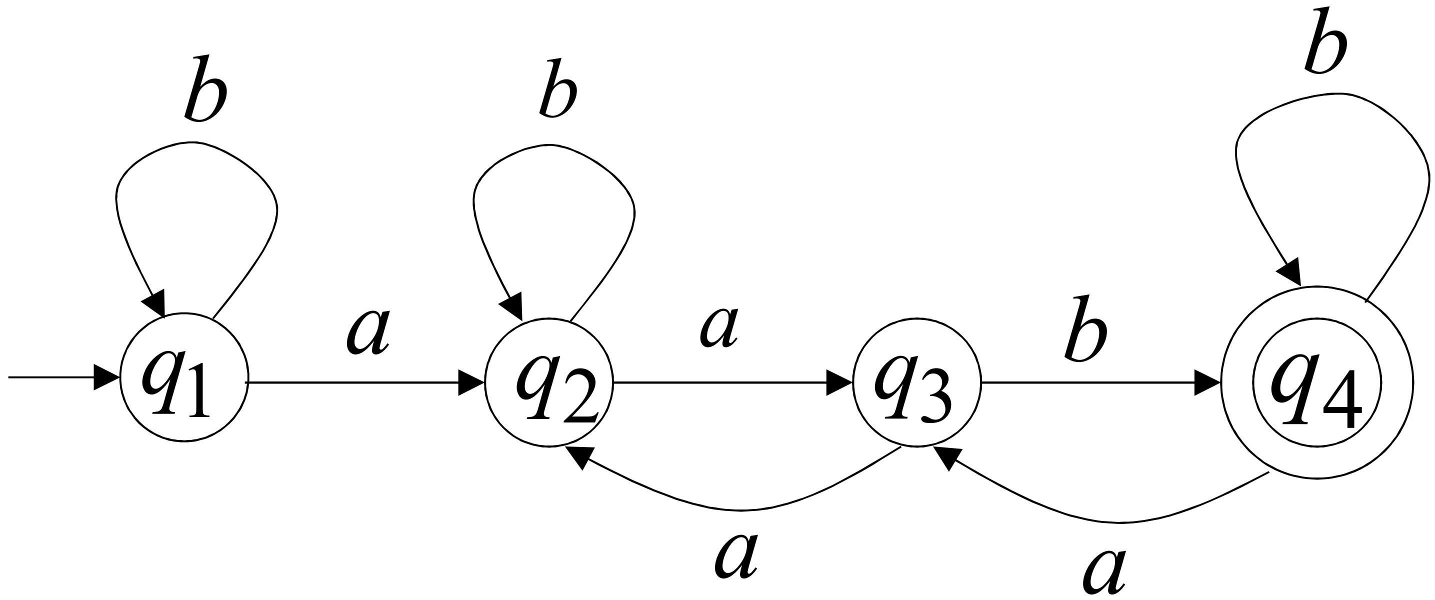
Prove that there is no DFA or NFA or RE that accepts L

Difficulty: this is not easy to prove

(since there is an infinite number of them)

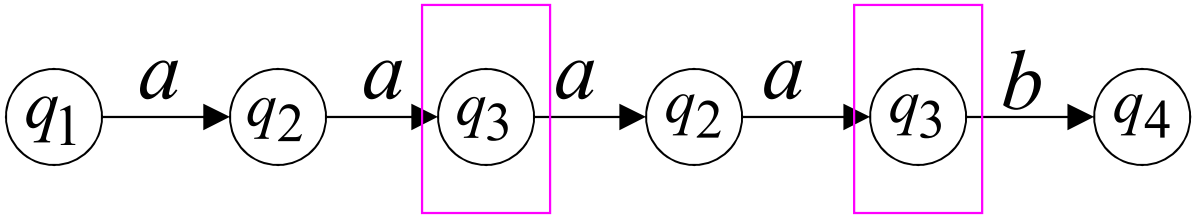
Solution: use the Pumping Lemma !!!

Consider a DFA with 4 states

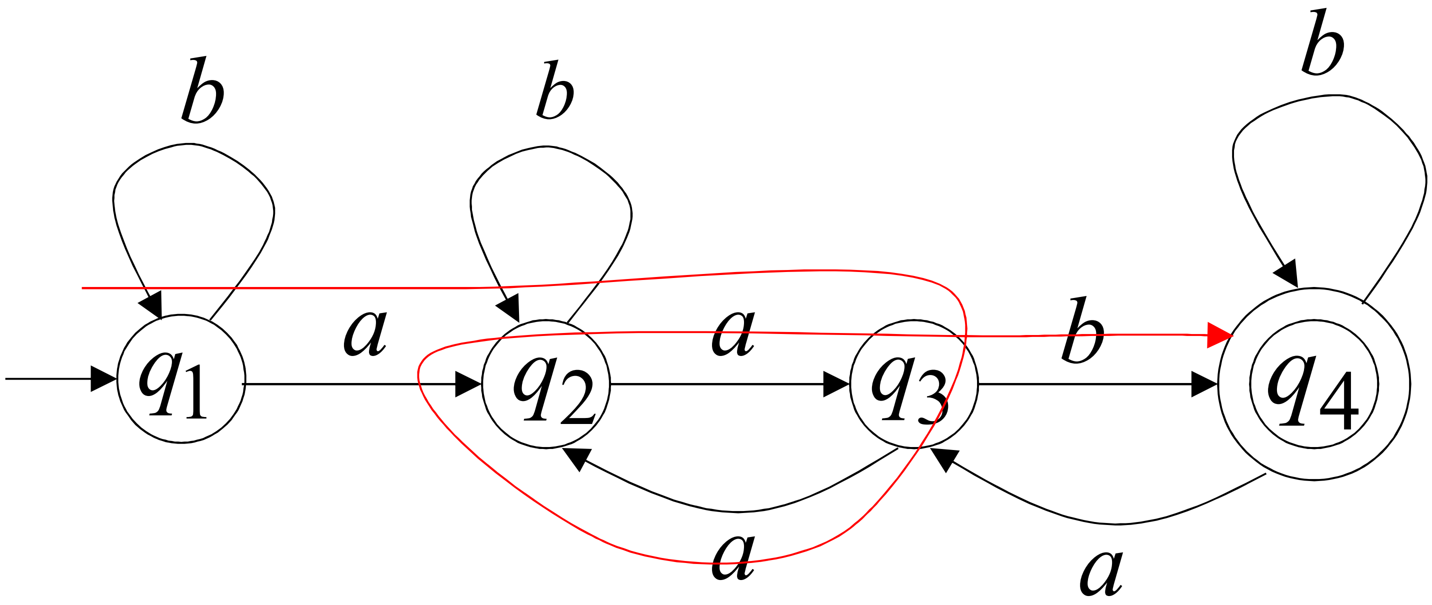


Consider the walk of a “long” string: $aaaab$
(length at least 4)

A state is repeated in the walk of $aaaab$



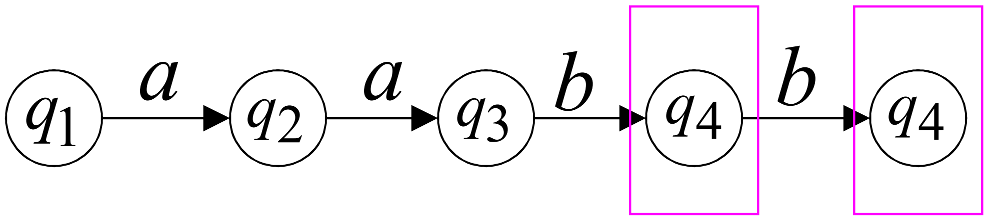
$$q_1 \xrightarrow{a} q_2 \xrightarrow{a} \boxed{q_3} \xrightarrow{a} q_2 \xrightarrow{a} \boxed{q_3} \xrightarrow{b} q_4$$

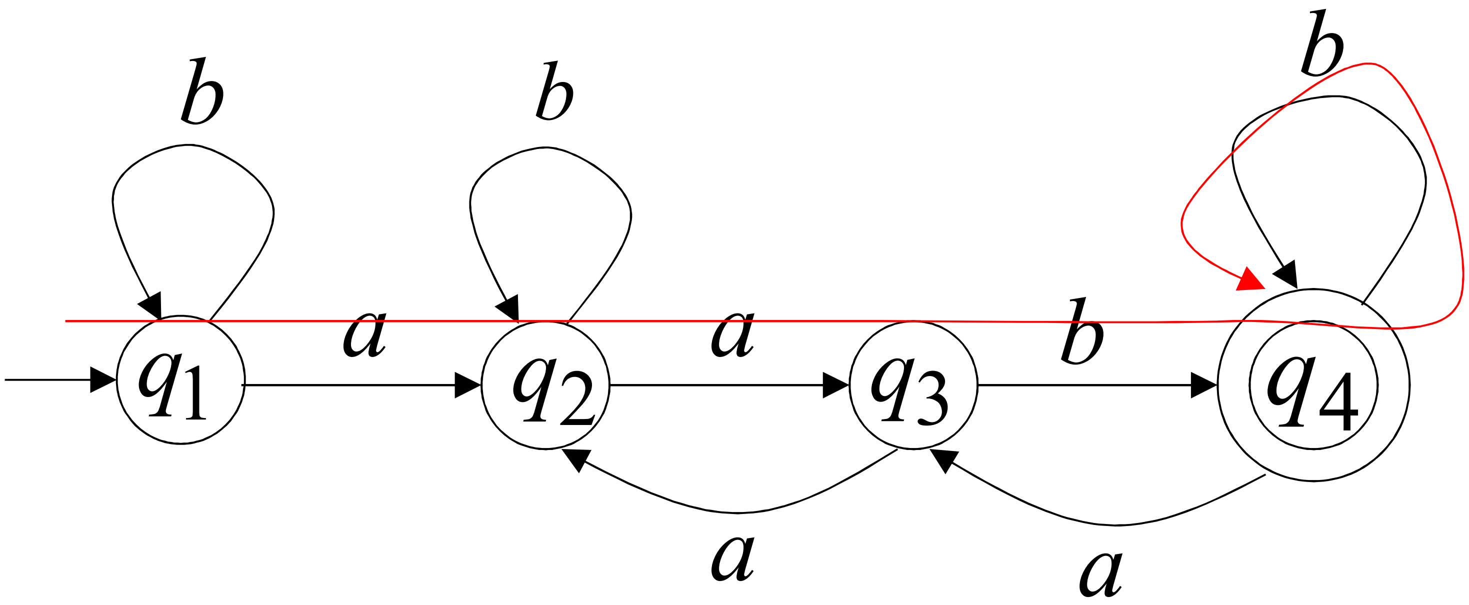


Consider the walk of a “long” string: $aabb$
(length at least 4)

Due to the pigeonhole principle:

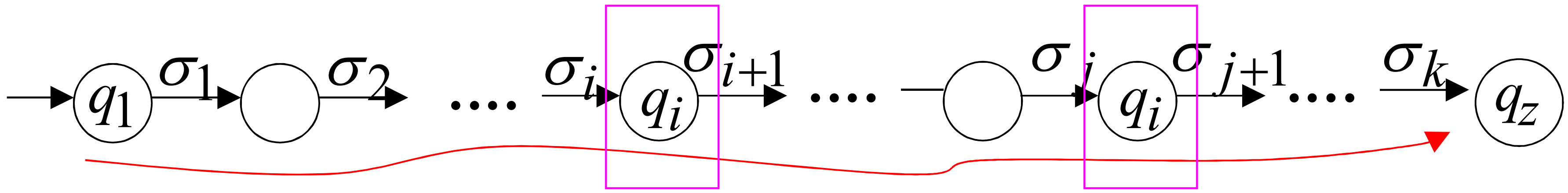
A state is repeated in the walk of $aabb$

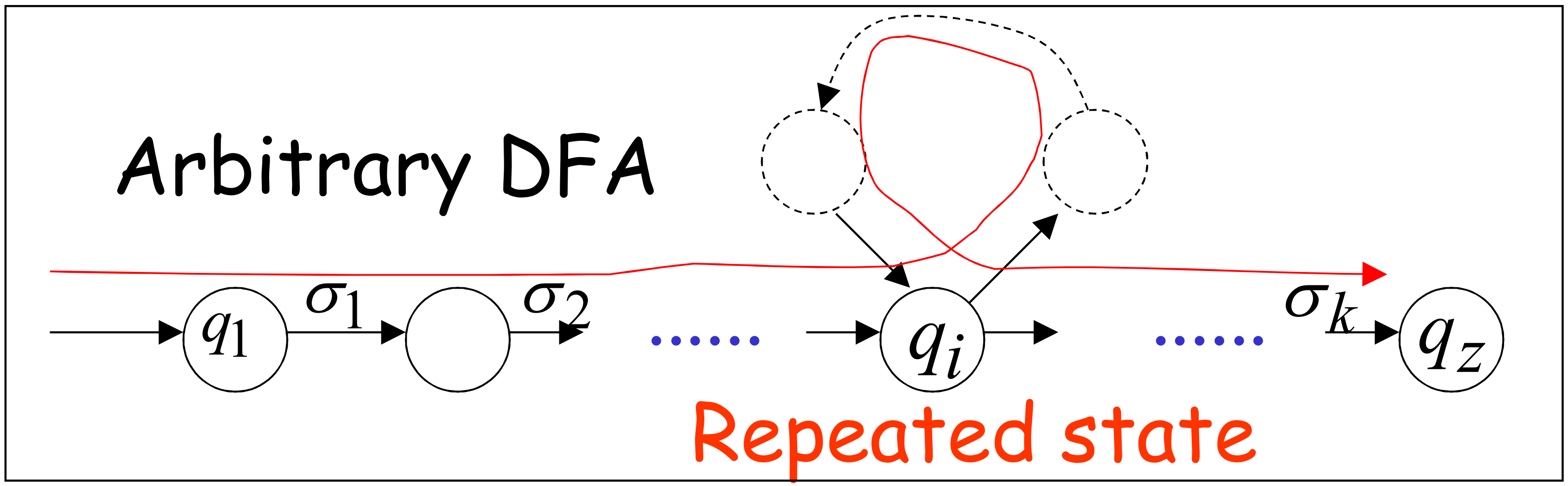




In General: If $|x| \geq \#\text{states of DFA}$,
by the pigeonhole principle,
a state is repeated in the walk

Walk of $x = \sigma_1 \sigma_2 \cdots \sigma_k$



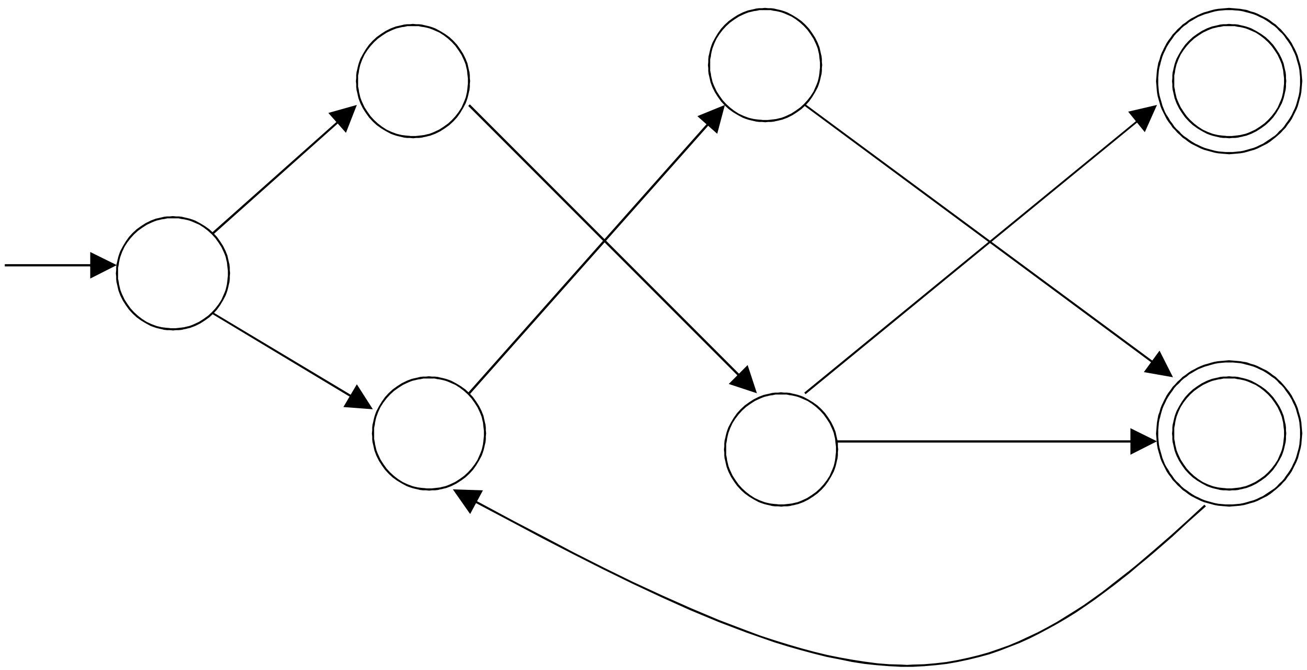


The Pumping Lemma

Take an infinite regular language L

(contains an infinite number of strings)

There exists a DFA that accepts L



n

states

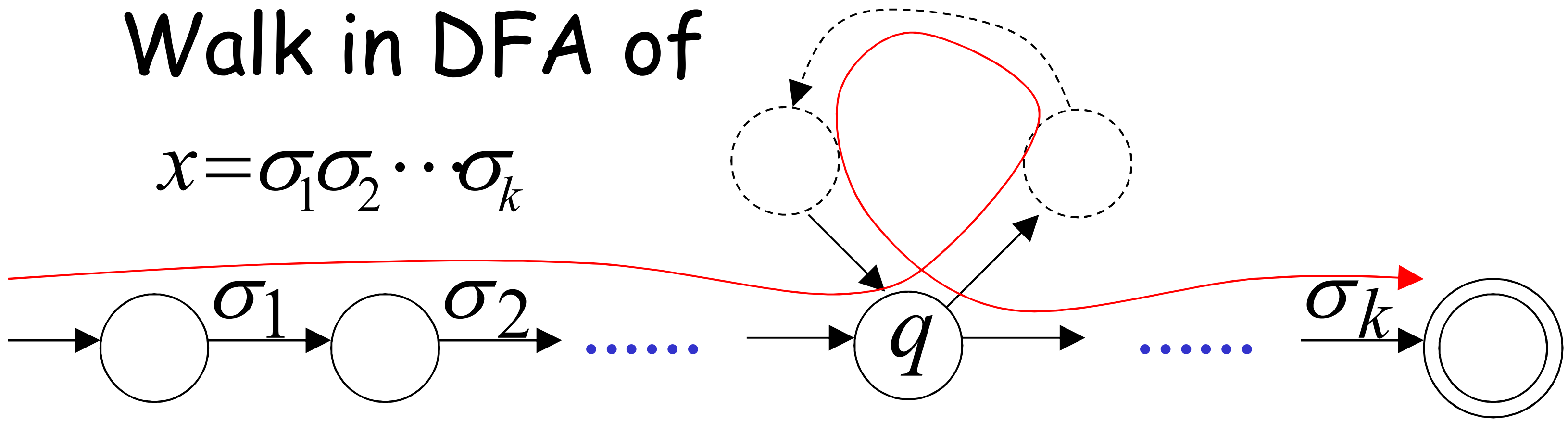
Take string $x \in L$ with $|x| \geq n$

(number of states of DFA)

then, at least one state is repeated in the walk of x

Walk in DFA of

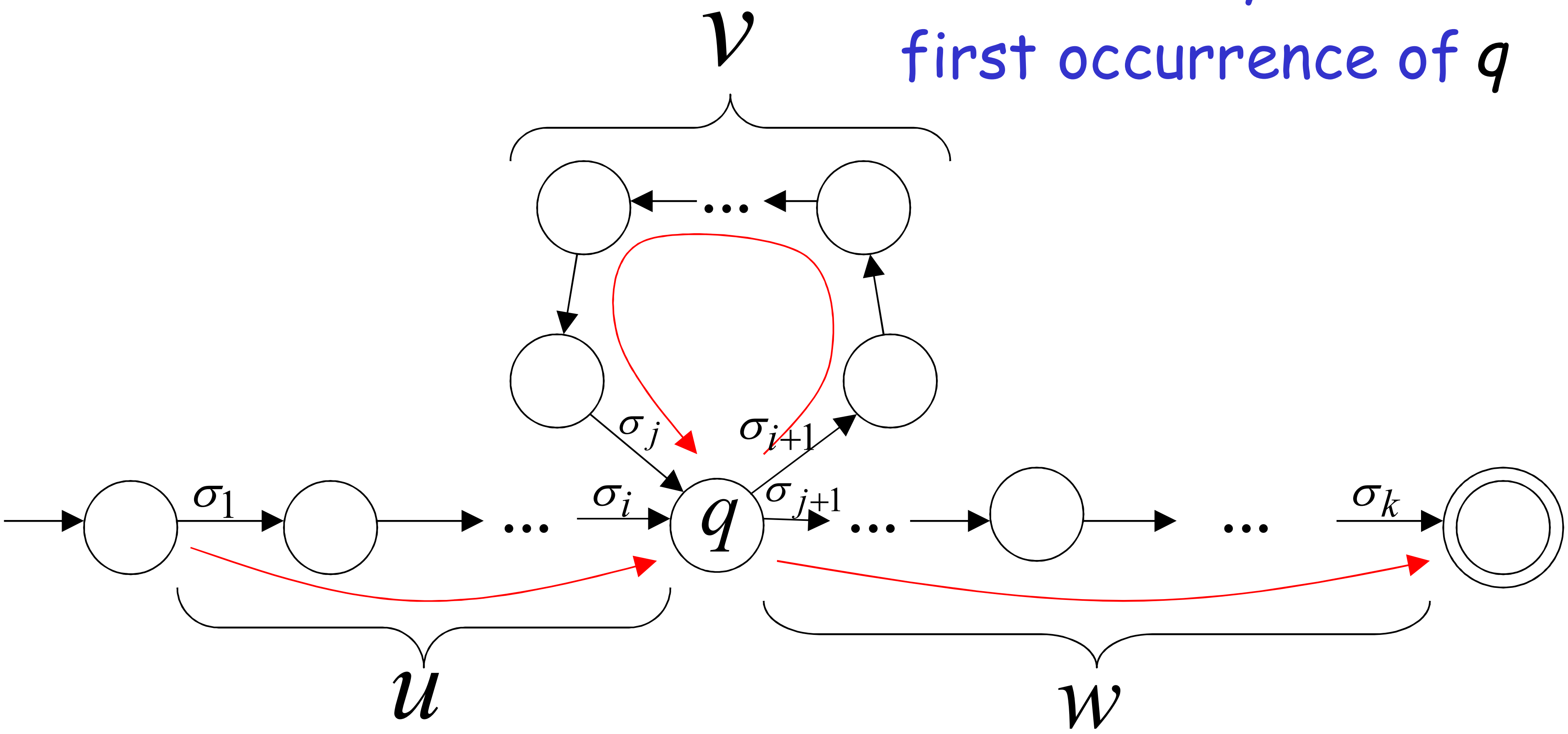
$x = \sigma_1 \sigma_2 \cdots \sigma_k$



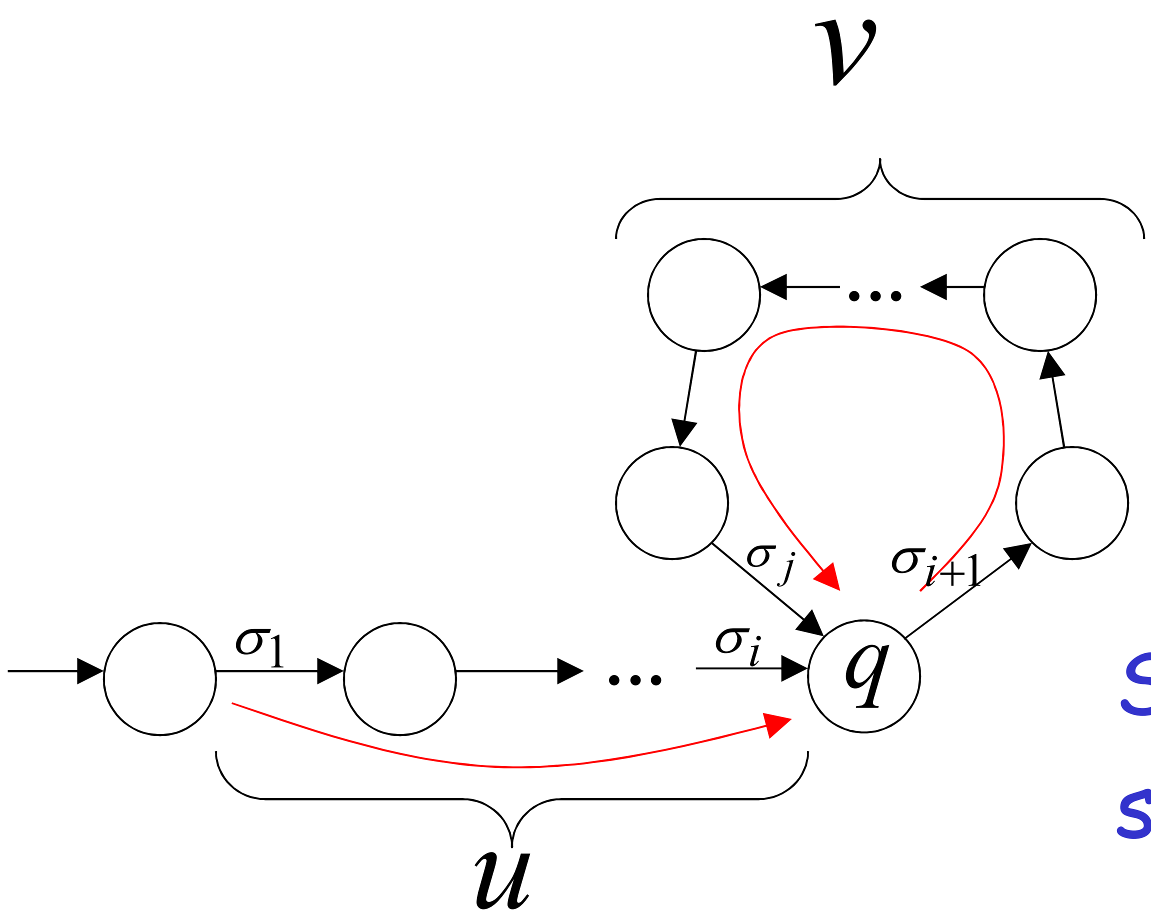
Repeated state in DFA

In DFA: $x = uvw$

contains only first occurrence of q



Observation: length $|u\,v| \leq n$ number of states of DFA



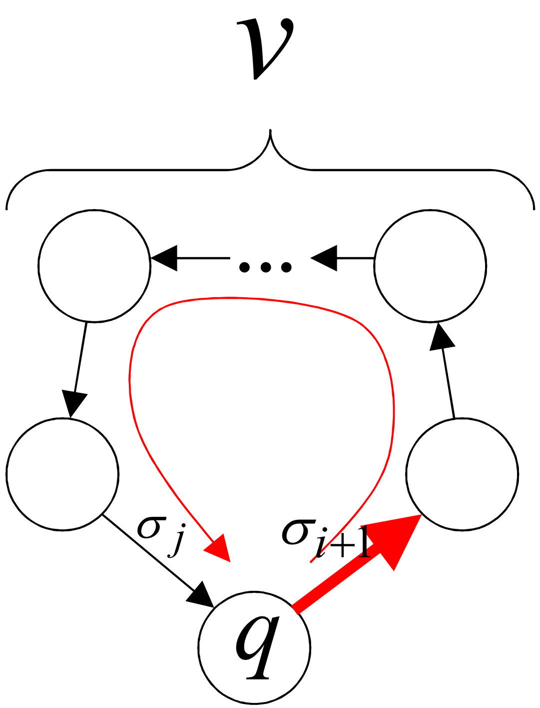
Unique States

Since, in uv no state is repeated

(except q)

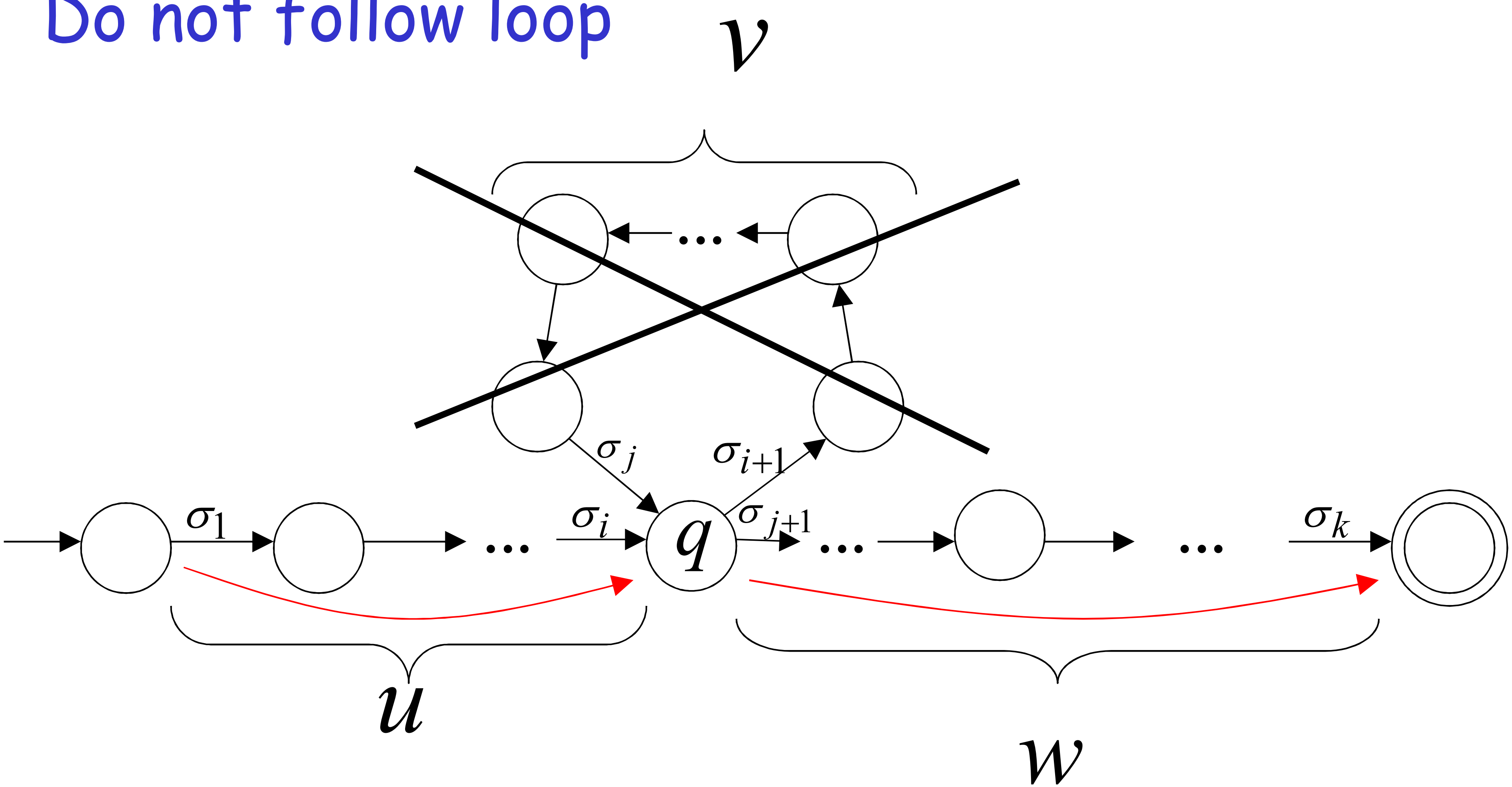
Observation: length $|v| \geq 1$

Since there is at least one transition in loop



Additional string: The string uw is accepted

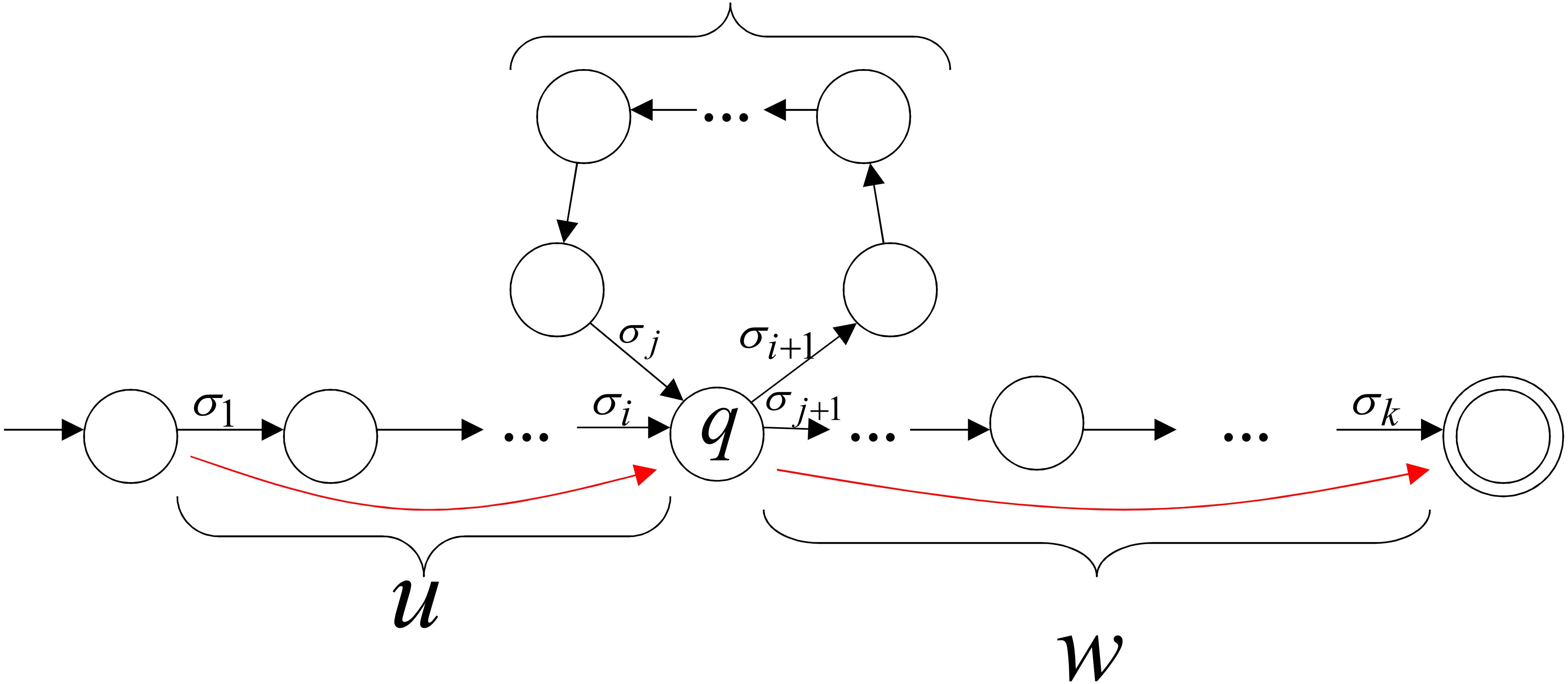
Do not follow loop



Additional string:

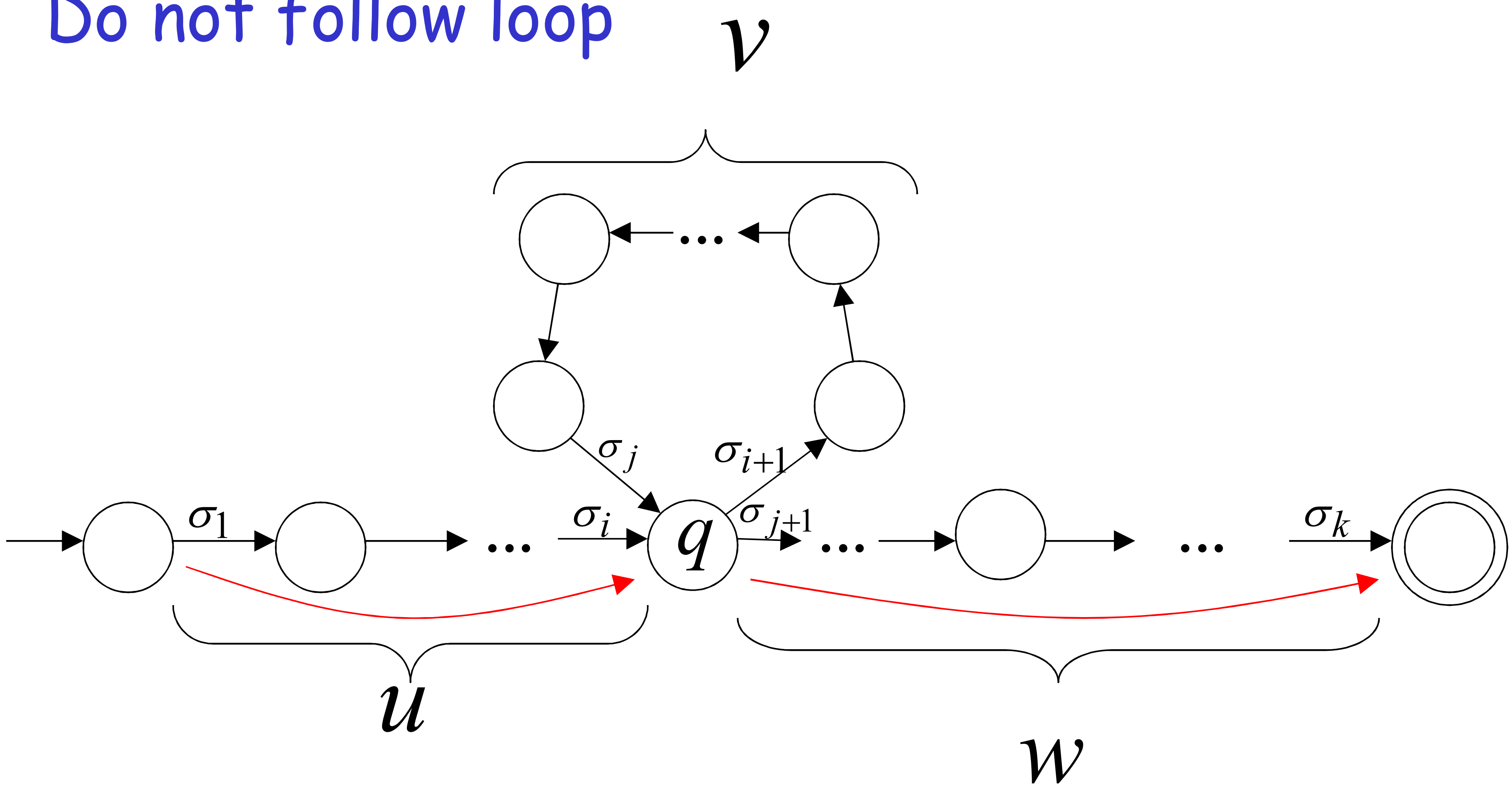
The string $uvvw$ is accepted

Do not follow loop v



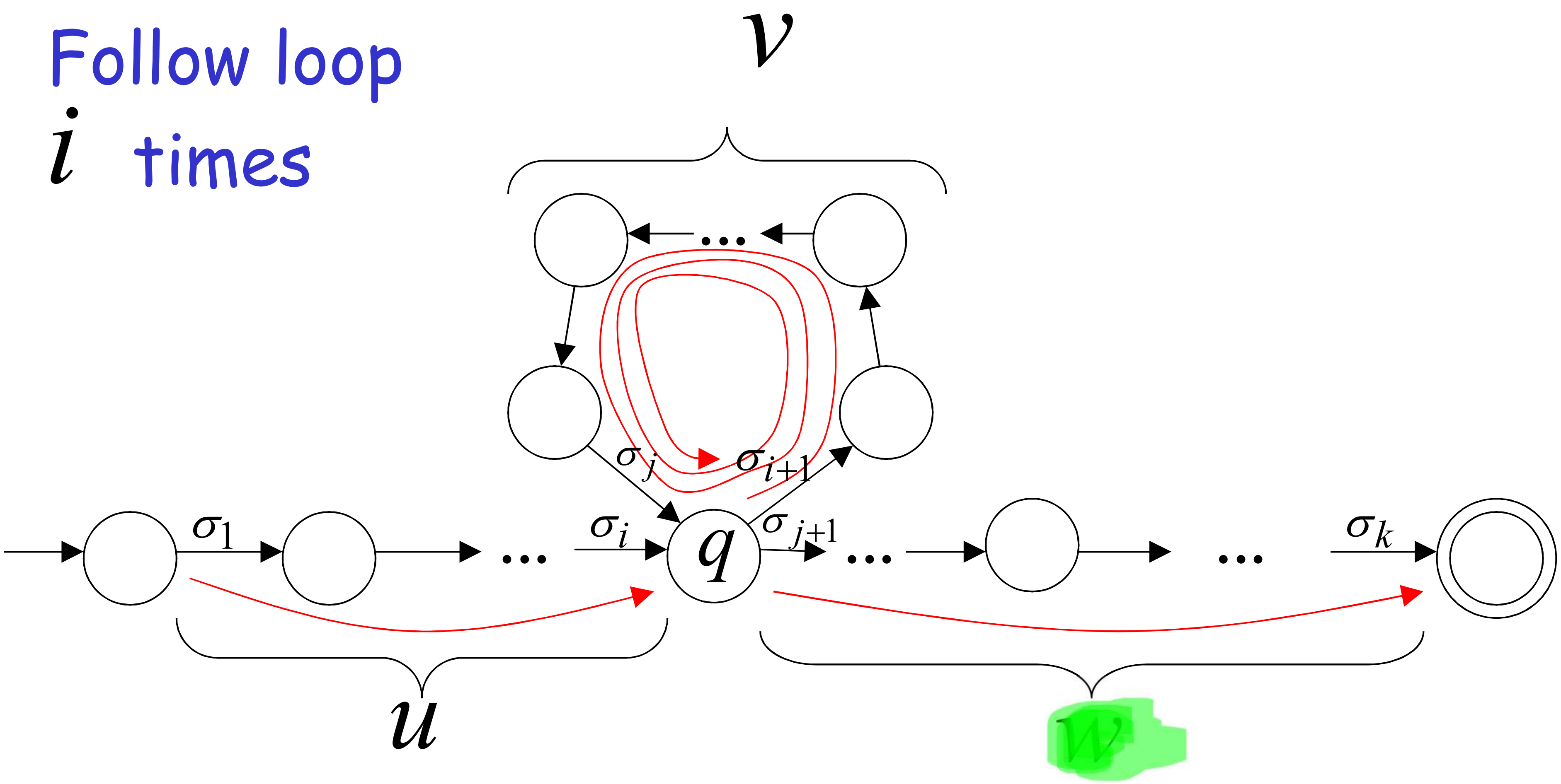
Additional string: The string $uvvvw$ is accepted

Do not follow loop v



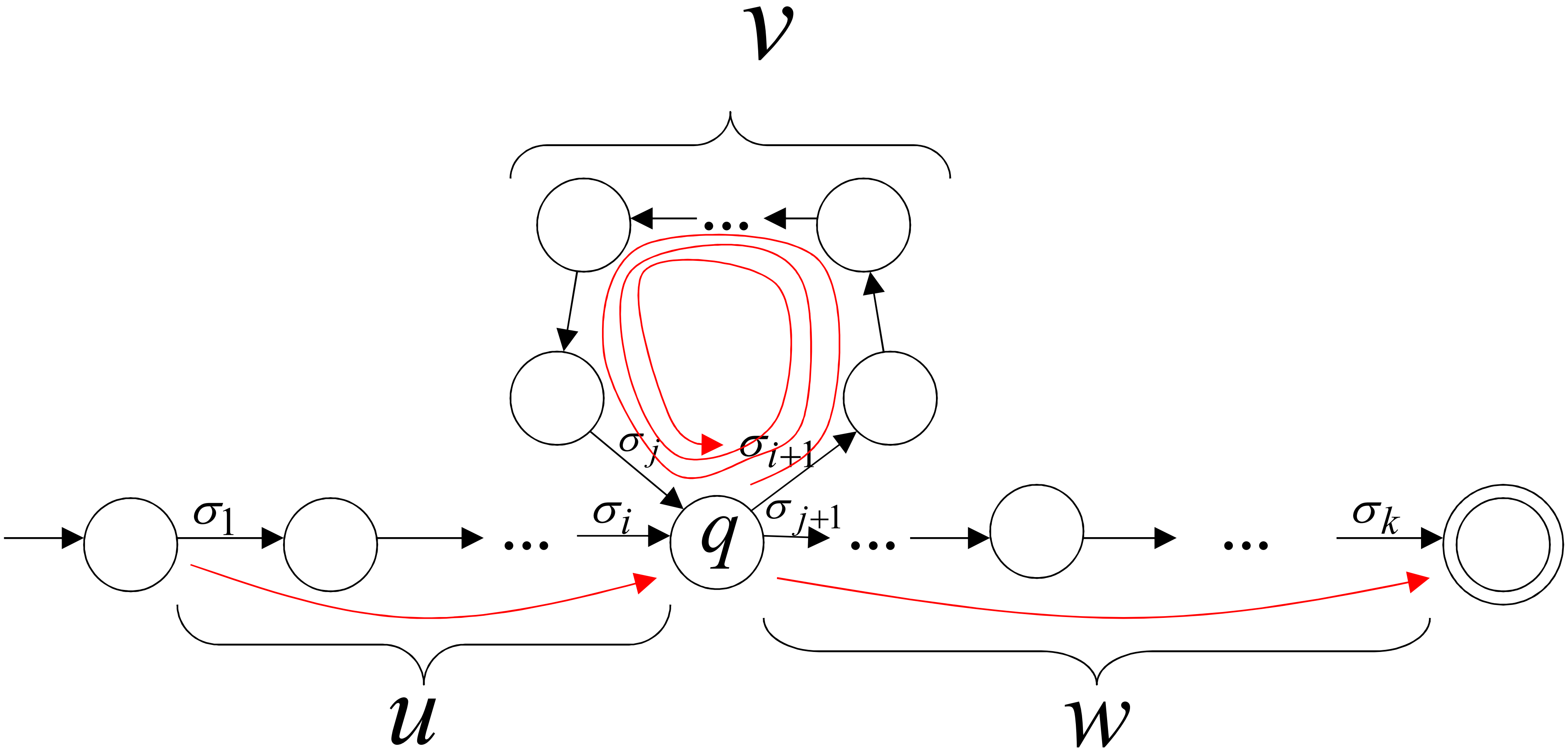
In General: The string $u\,v^i\,w$ is accepted $i = 0, 1, 2, \ldots$

Follow loop i times



Therefore: $uv^i\ w \in L \qquad i = 0, 1, 2, \ldots$

Language accepted by the DFA



The Pumping Lemma:

- Given a infinite regular language L
- there exists an integer n (critical length)
- for any string $x \in L$ with length $|x| \geq n$
- we can write $x = u\,v\,w$
- with $|u\,v| \leq n$ and $|v| \geq 1$
- such that: $u v^i w \in L$ $i = 0, 1, 2, \ldots$

Observation:

Every language of finite size has to be regular

(we can easily construct an NFA that accepts every string in the language)

Therefore, every non-regular language has to be of infinite size

(contains an infinite number of strings)

Applications

of

the Pumping Lemma

Non-Regular Language: Example

Theorem: The language:

$$L = \{0^k 1^k \mid k \geq 0\} \qquad (1)$$

is not regular.

Proof: *(by contradiction)* Suppose that L is regular. Then there exists a DFA M such that:

$$L = L(M) \qquad (2)$$

We will show that M accepts some strings not in L, contradicting (2).

Suppose that M has n states, and <u>choose</u> a string $x = 0^m 1^m$, where the constant $m >> n$.

By (1), x is in L.

By (2), x is also in L(M), *note that the machine accepts a language not just a string*

Since $|x| = m >> n$, it follows from the pumping lemma that:

$x = uvw$
$1 \leq |uv| \leq n$
$1 \leq |v|$, and
$uv^i w$ is in L(M), for all $i \geq 0$

Since $1 \leq |uv| \leq n$ and $n << m$, it follows that $1 \leq |uv| < m$.

Also, since $x = 0^m 1^m$ it follows that uv is a substring of 0^m.

In other words $v=0^j$, for some $j \geq 1$.

Since $\mathbf{uv^i w}$ is in L(M), for all $i \geq 0$, it follows that $0^{m+cj}1^m$ is in L(M), for all $c \geq 1$ (no. of loops), and $j \geq 1$ (length of the loop)

But by (1) and (2), $0^{m+cj}1^m$ is not in L(M), for any $c \geq 1$, i.e., $m+cj > m$, a contradiction.

Non-Regularity Example

Theorem: The language:

$$L = \{0^k 1^k 2^k \mid k \geq 0\} \qquad (1)$$

is not regular.

Proof: (by contradiction) Suppose that L is regular. Then there exists a DFA M such that:

$$L = L(M) \qquad (2)$$

We will show that M accepts some strings not in L, contradicting (2).

Suppose that M has n states, and consider a string $x=0^m1^m2^m$, where the constant *m>>n*.

By (1), x is in L.

By (2), x is also in L(M), *note that the machine accepts a language not just a string*

Since $|x| = m >> n$, it follows from the pumping lemma that:

$x = uvw$

$1 \leq |uv| \leq n$

$1 \leq |v|$, and

uv^iw is in L(M), for all $i \geq 0$

Since $1 \leq |uv| \leq n$ and $n<<m$, it follows that $1 \leq |uv| \leq m$.

Also, since $x = 0^m1^m2^m$ it follows that uv is a substring of 0^m.

In other words $v=0^j$, for some $j \geq 1$.

Since $\mathbf{uv^iw}$ is in L(M), for all $i \geq 0$, it follows that $0^{m+cj}1^m2^m$ is in L(M), for all $c \geq 1$ and $j \geq 1$.

But by (1) and (2), $0^{m+cj}1^m2^m$ is not in L(M), for any integer $c \geq 1$, a contradiction.

Note that the above proof is almost identical to the previous proof.

Non-Regularity Example

Theorem: The language:

$$L = \{0^m 1^n 2^{m+n} \mid m,n \geq 0\} \qquad (1)$$

is not regular.

Proof: (by contradiction) Suppose that L is regular. Then there exists a DFA M such that:

$$L = L(M) \qquad (2)$$

We will show that M accepts some strings not in L, contradicting (2).

Suppose that M has n states, and consider a string $x=0^m 1^n 2^{m+n}$, where m>>n.

By (1), x is in L.

By (2), x is also in L(M).

Since $|x| = m >> n$, it follows from the pumping lemma that:

$x = uvw$

$1 \leq |uv| \leq n$

$1 \leq |v|$, and

uv^iw is in L(M), for all $i \geq 0$

Since $1 \leq |uv| \leq n$ and $n << m$, it follows that $1 \leq |uv| \leq m$.

Also, since $x = 0^m 1^n 2^{m+n}$ it follows that uv is a substring of 0^m.

In other words $v=0^j$, for some $j \geq 1$.

Since uv^iw is in L(M), for all $i \geq 0$, it follows that $0^{m+cj}1^m2^{m+n}$ is in L(M), for all $c \geq 1$. In other words v can be "pumped" as many times as we like, and we still get a string in L(M).

But by (1) and (2), $0^{m+cj}1^n2^{m+n}$ is not in L(M), for any $c \geq 1$, *because the acceptable expression should be* $0^{m+cj}1^n2^{m+\mathbf{cj}+n}$, a contradiction.

What about $\{0^m 1^n \mid m, n \geq 0\}$?

$\{0^m 1^n \mid m, n \geq 0, \text{ and } m<n\}$?

$\{0^m 1^n \mid m, n \geq 0, \text{ and } m = n\text{^}2\}$?

$\{0^m 1^n \mid m, n \geq 0, \text{ and } m>n\}$?

Are these regular languages, or not?

Non-regular language $\{a^n b^n : \; n \geq 0\}$

Regular languages

$L(a^* b^*)$