第二讲 非确定型有限自动机与正则表达式

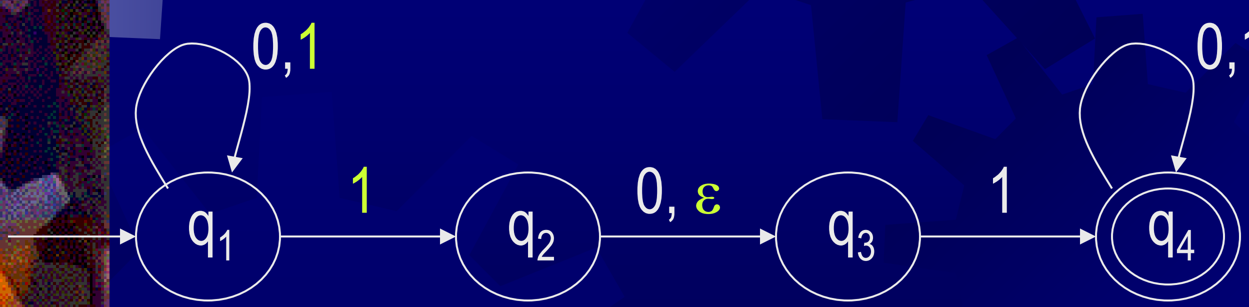
罗 欢

Email：hluo@fzu.edu.cn

非确定型有限自动机

非确定性(nondeterminism)

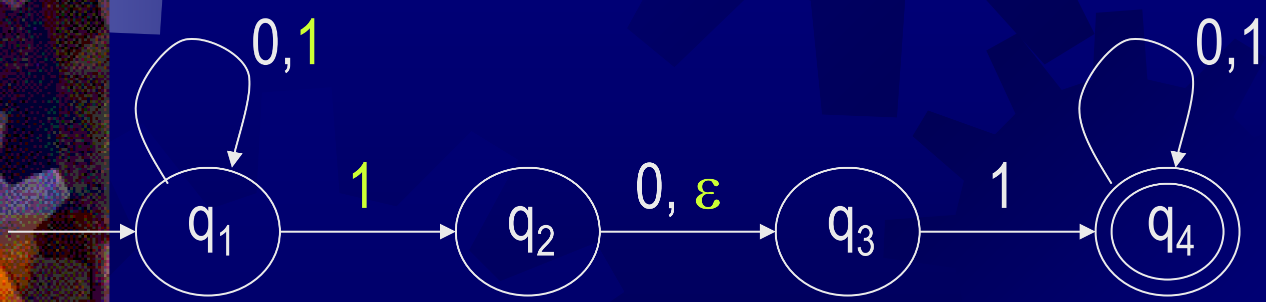
- 确定型(deterministic)计算
  - 下一个状态是唯一确定的
- 非确定型(nondeterministic)计算
  - 下一个状态可以不唯一确定
  - ε移动，多种选择(含0种选择)



N_1

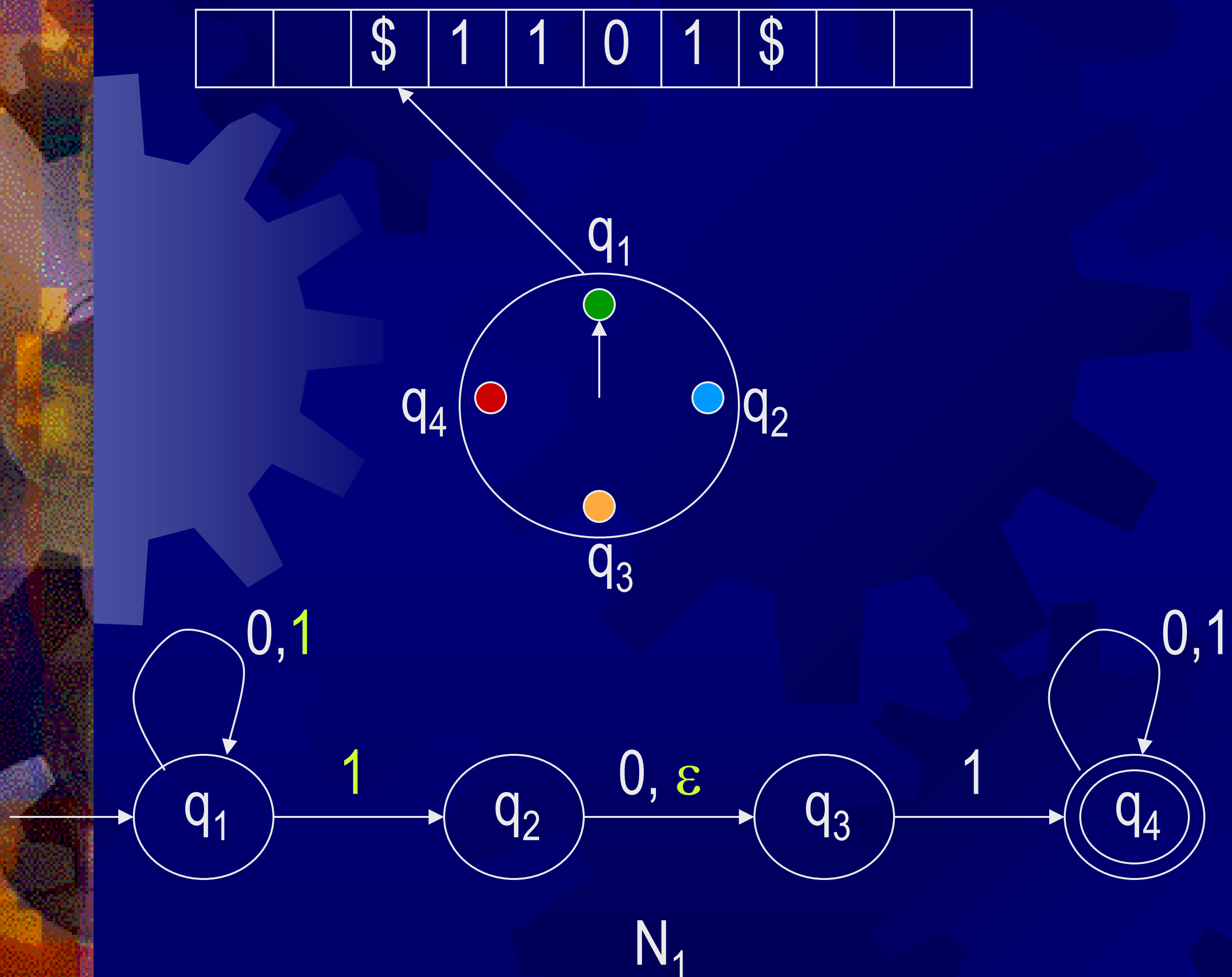
非确定型计算

- 非确定型计算
  - ε移动和多种选择产生不同备份
  - 无法移动时，该备份就消失
  - 有一个备份接受，整个计算就接受



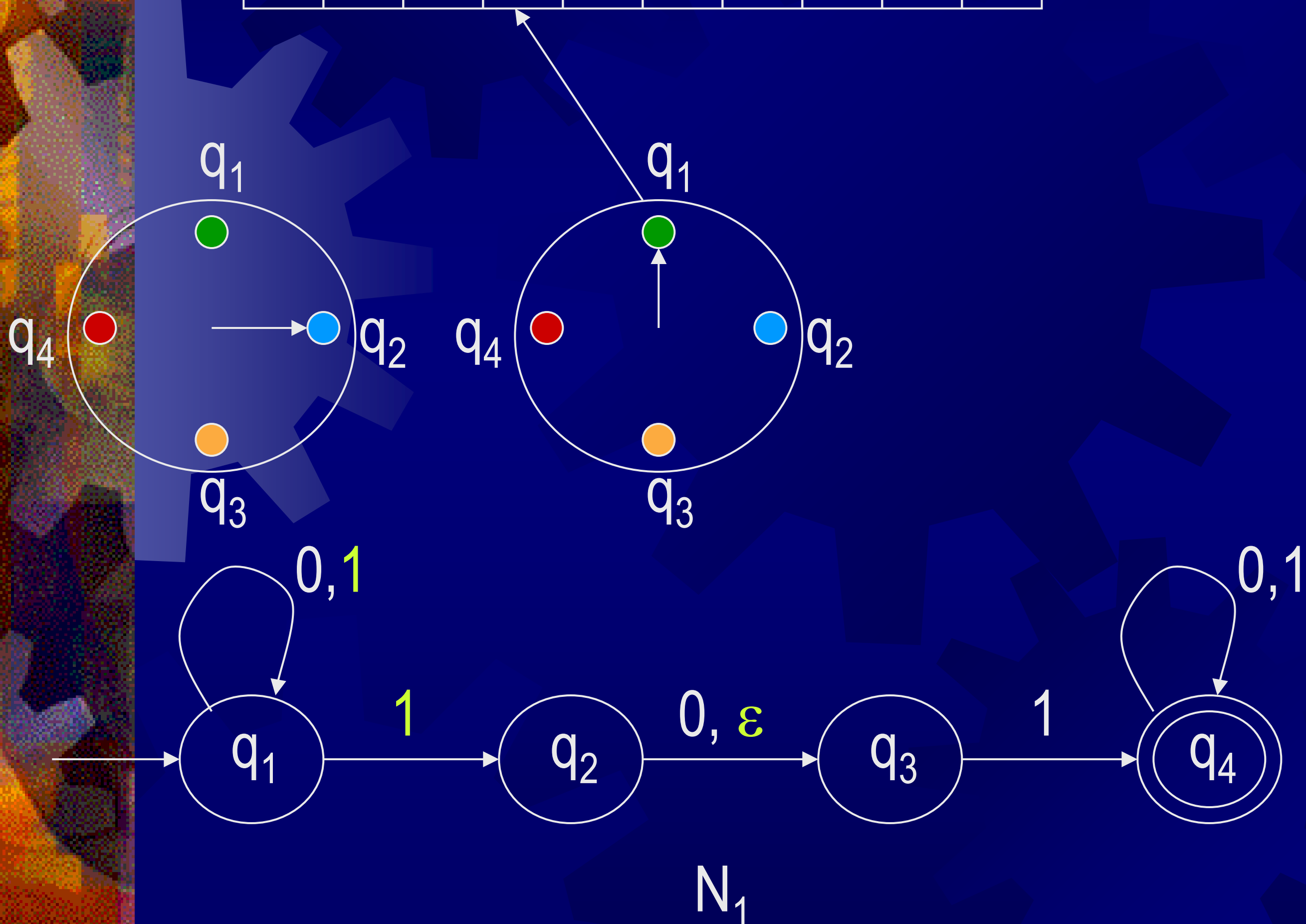
N_1

N_1在1101上计算(0)



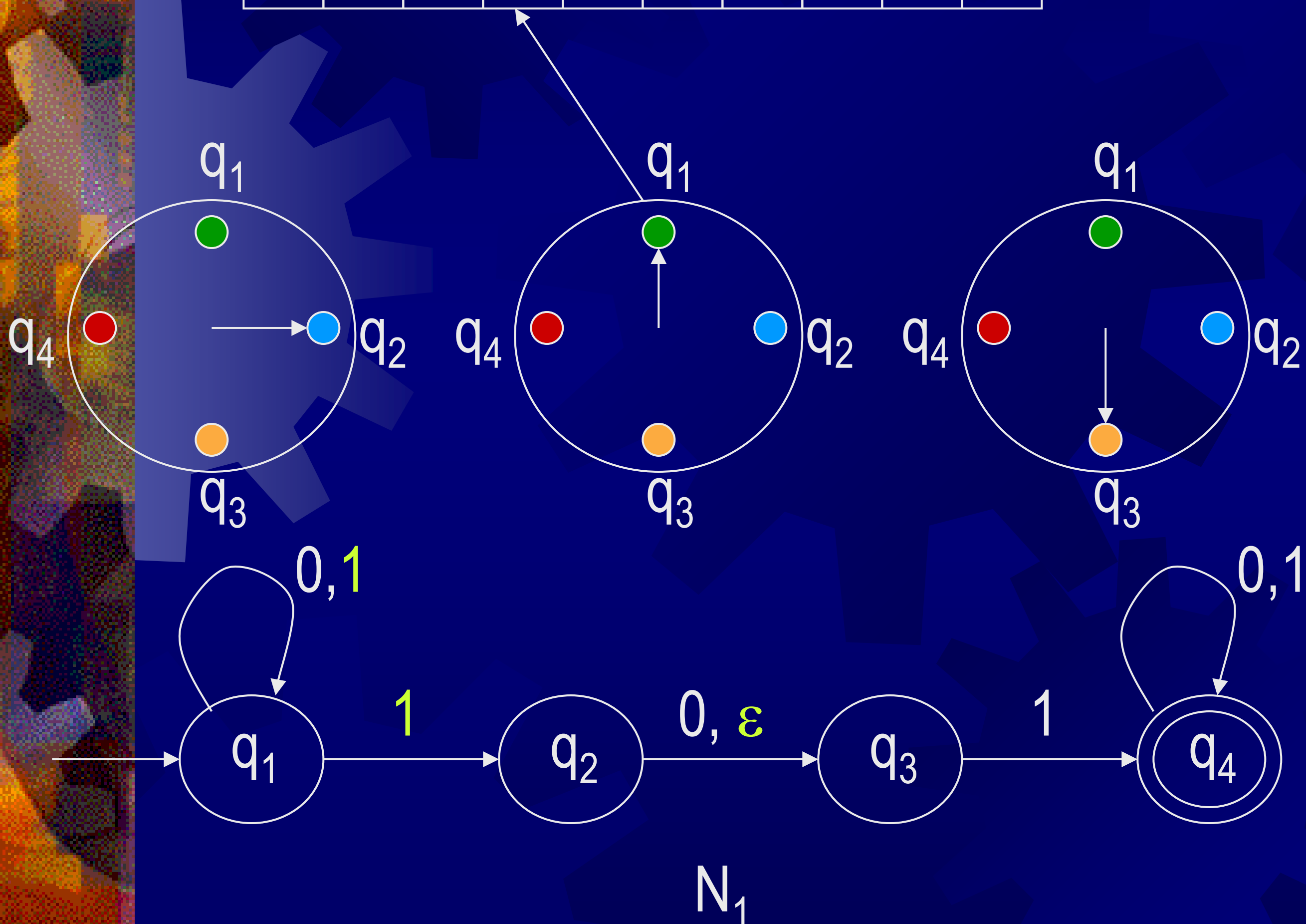
N_1在1101上计算(1)

		$	1	1	0	1	$		

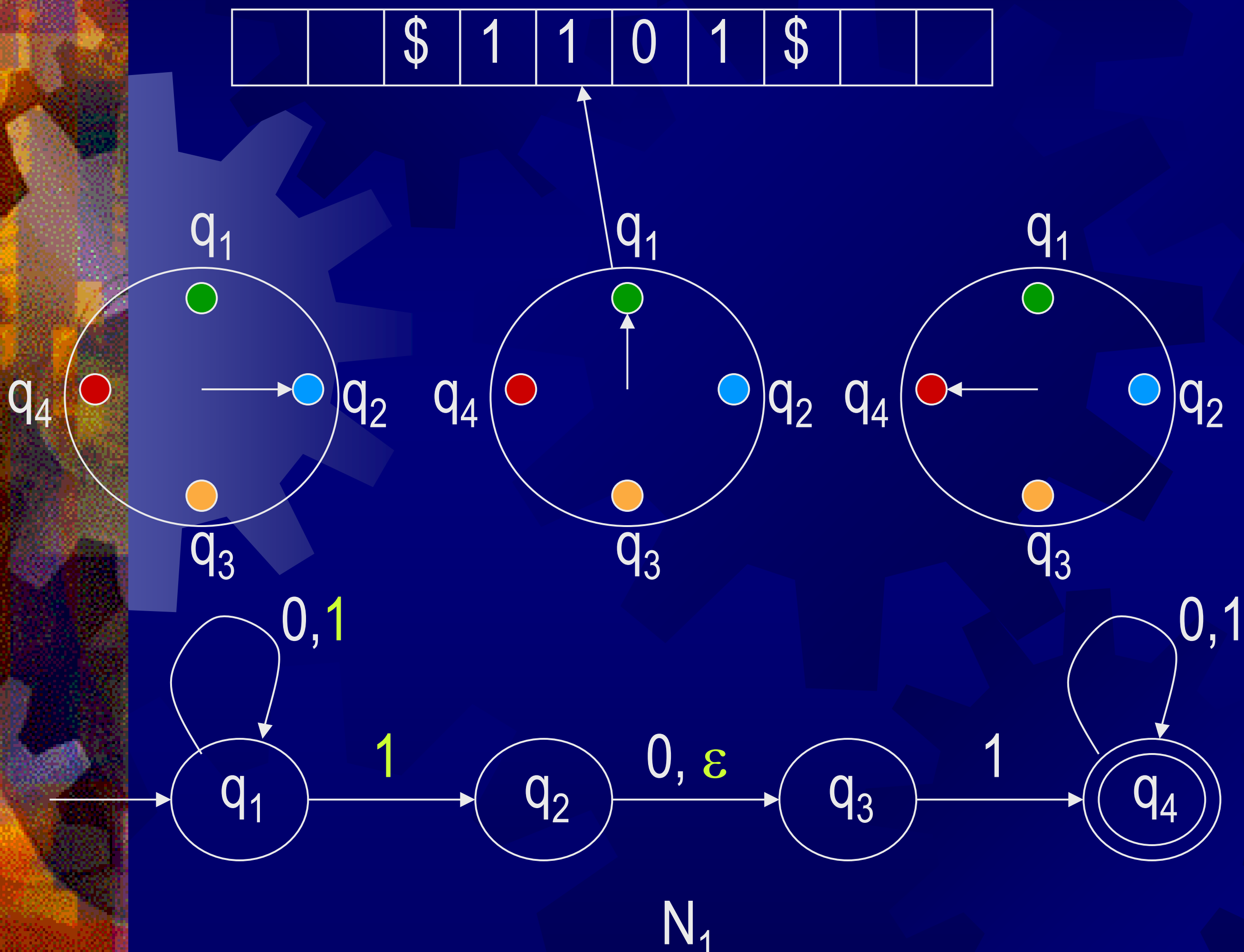


N_1在1101上计算(2)

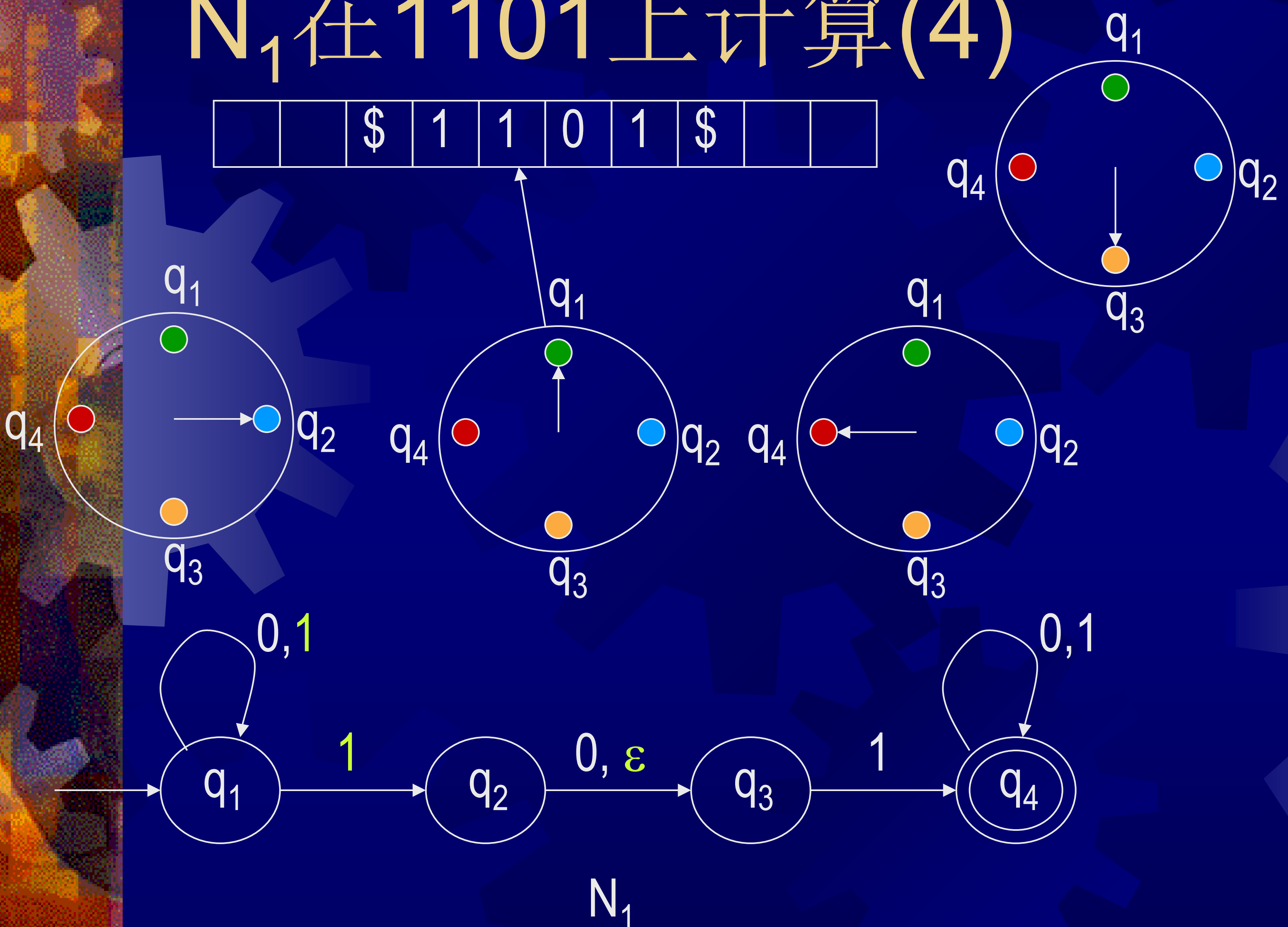
		$	1	1	0	1	$		



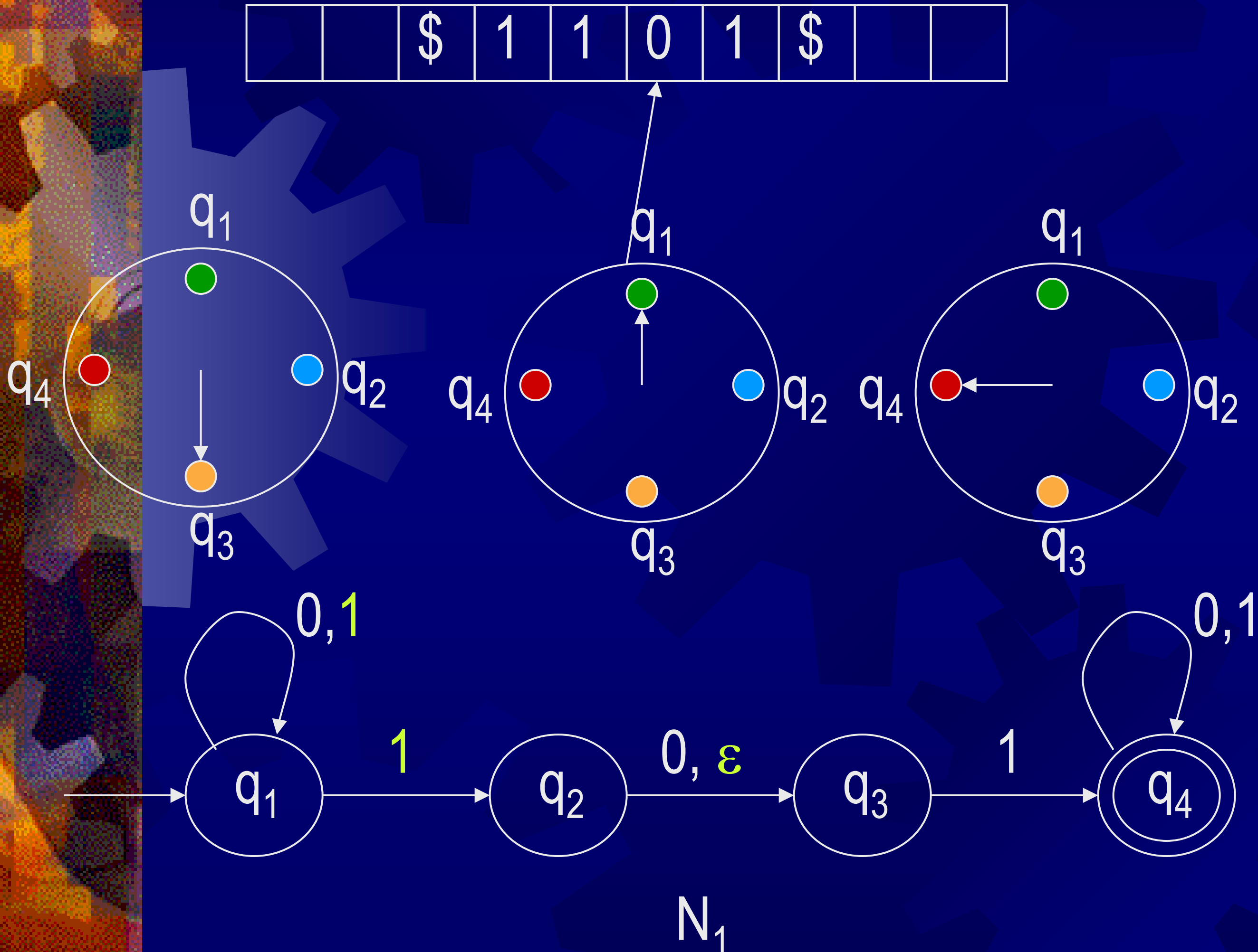
N_1在1101上计算(3)



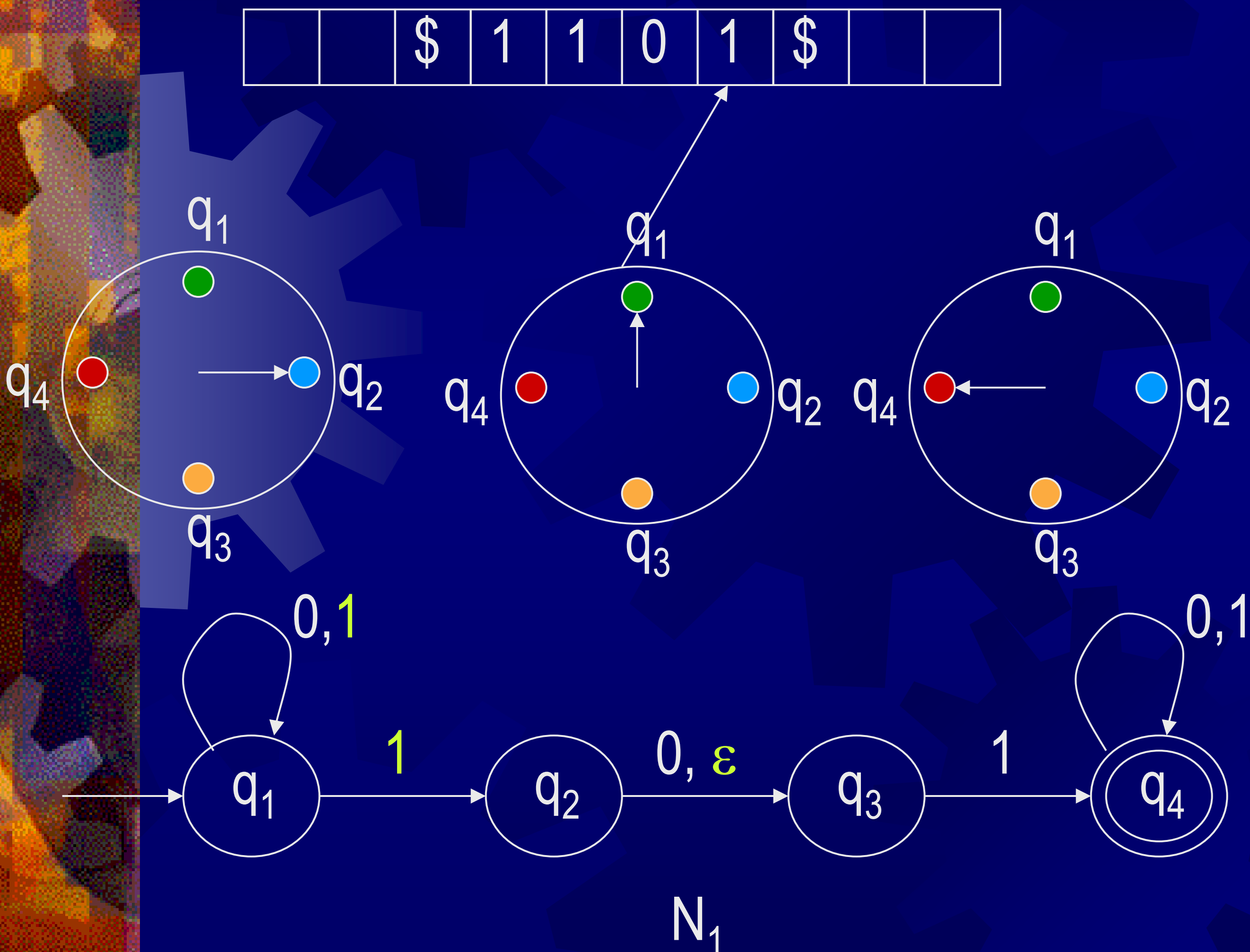
N_1在1101上计算(4)



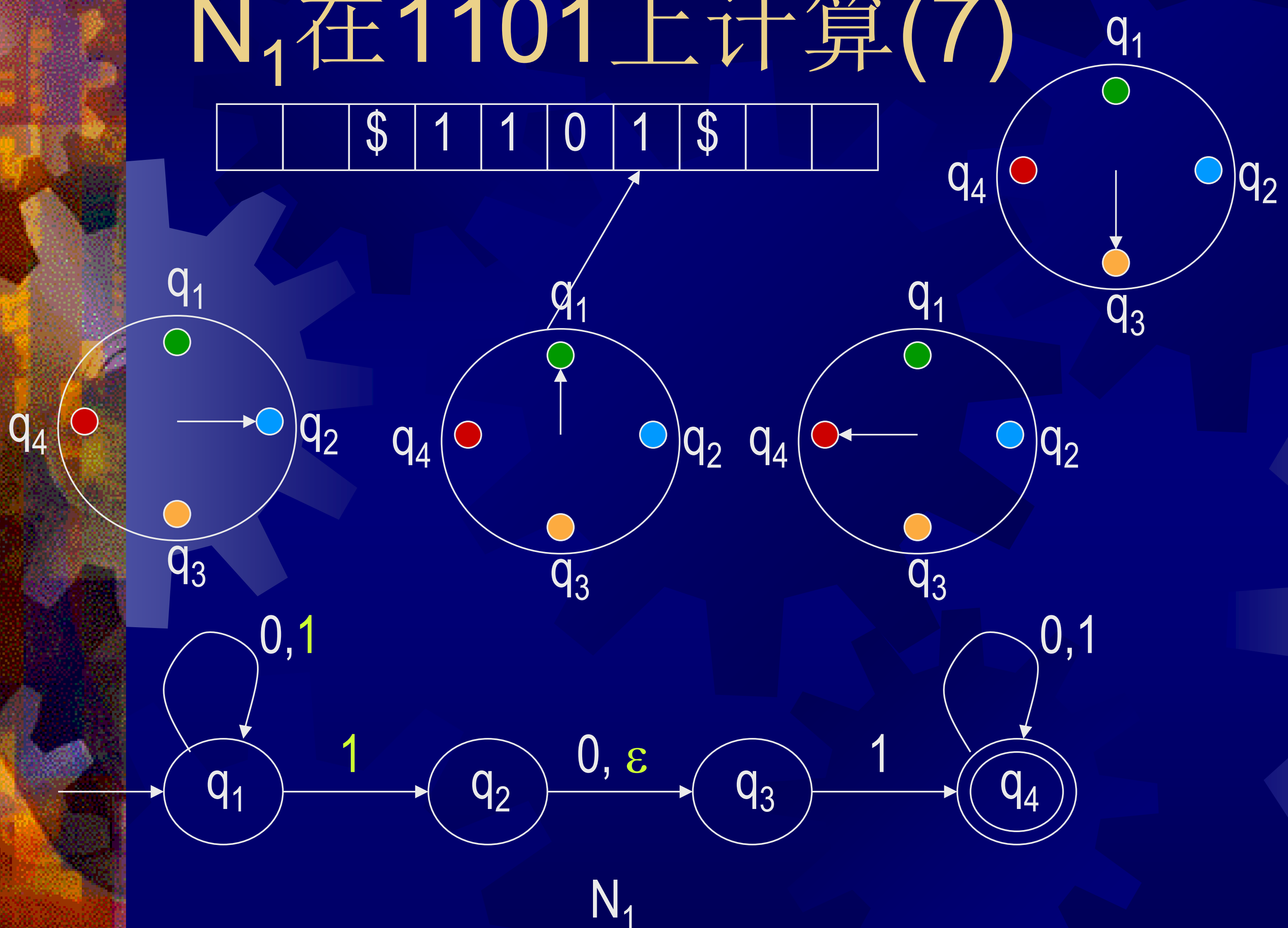
N_1在1101上计算(5)



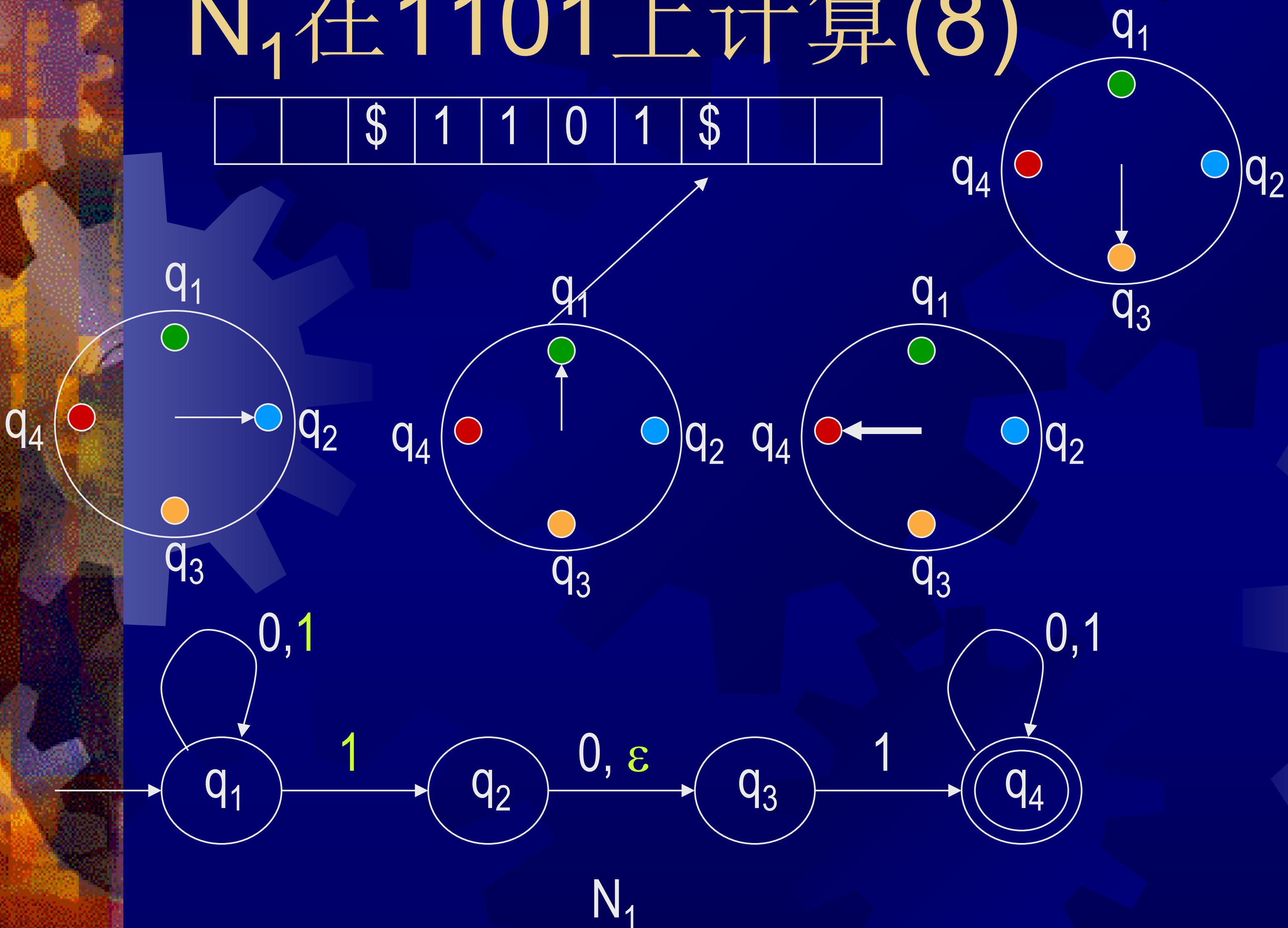
N_1在1101上计算(6)



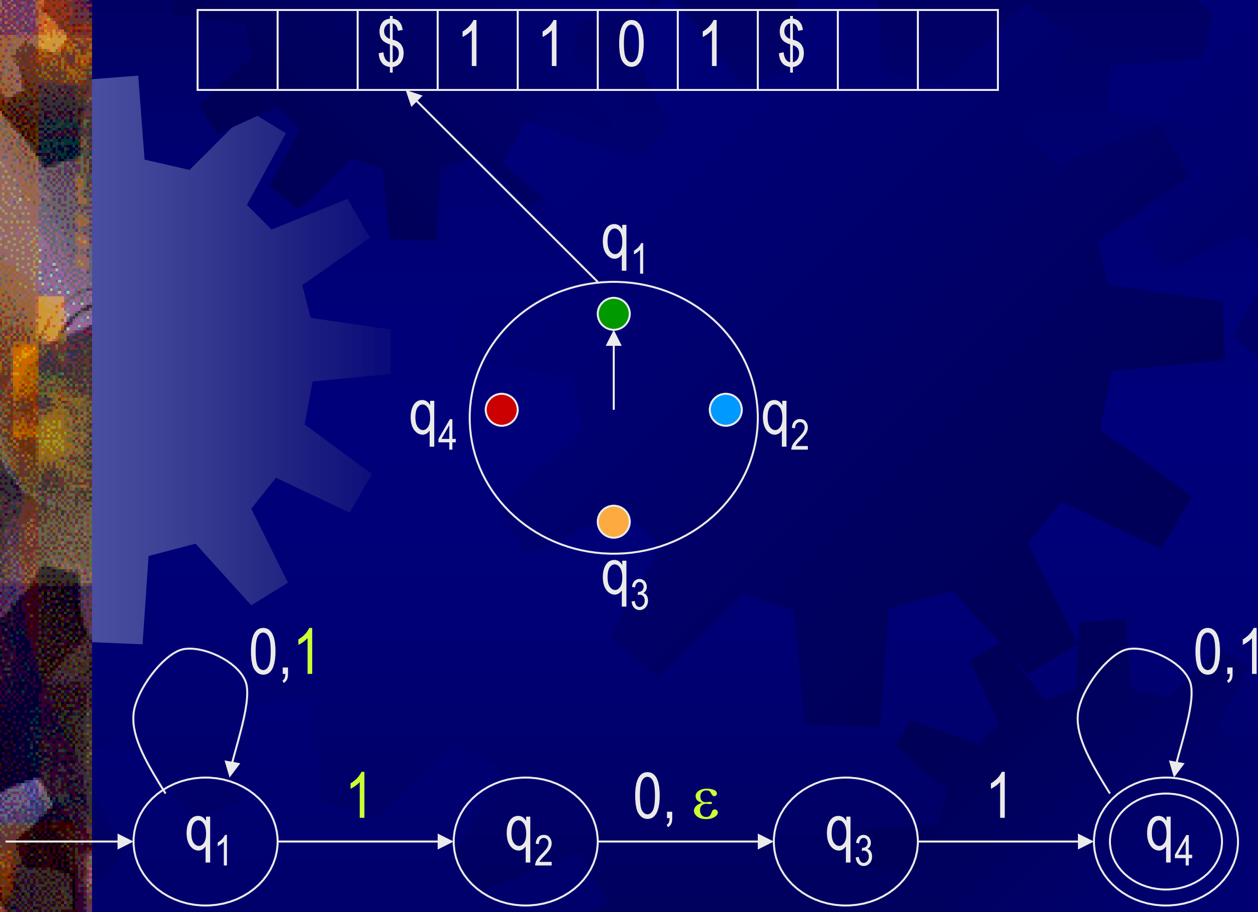
N_1在1101上计算(7)



N_1在1101上计算(8)

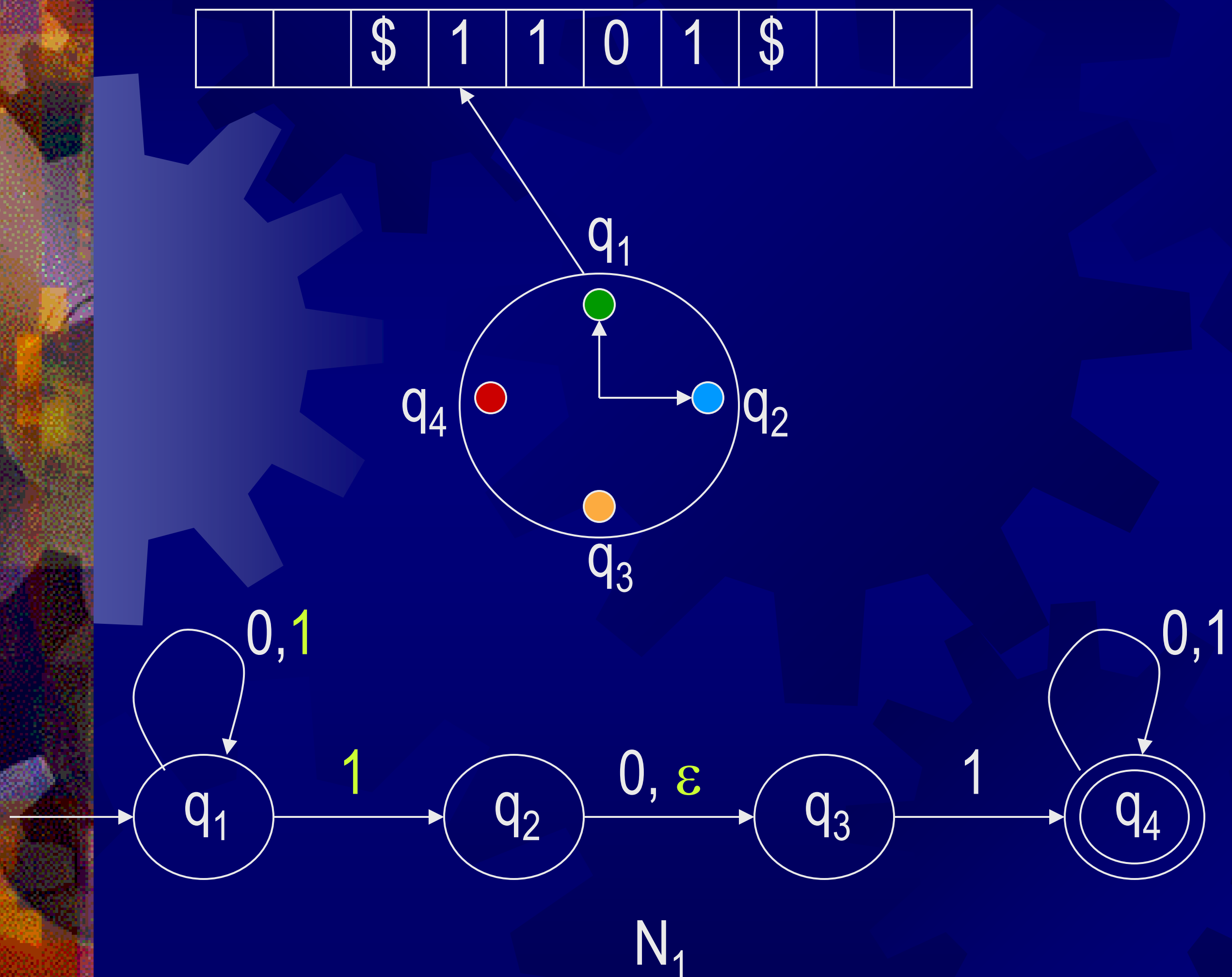


N_1在1101上计算(0)

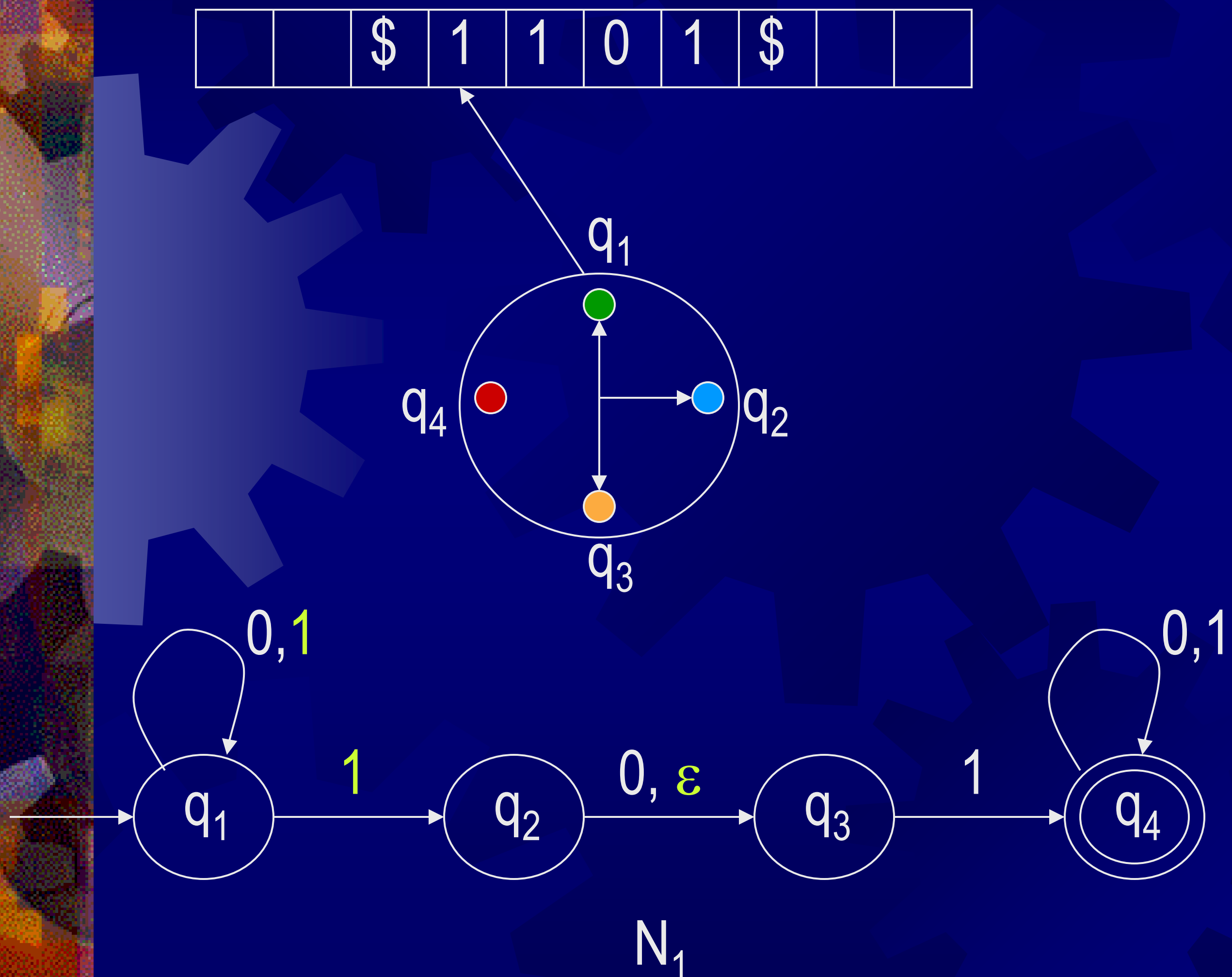


N_1

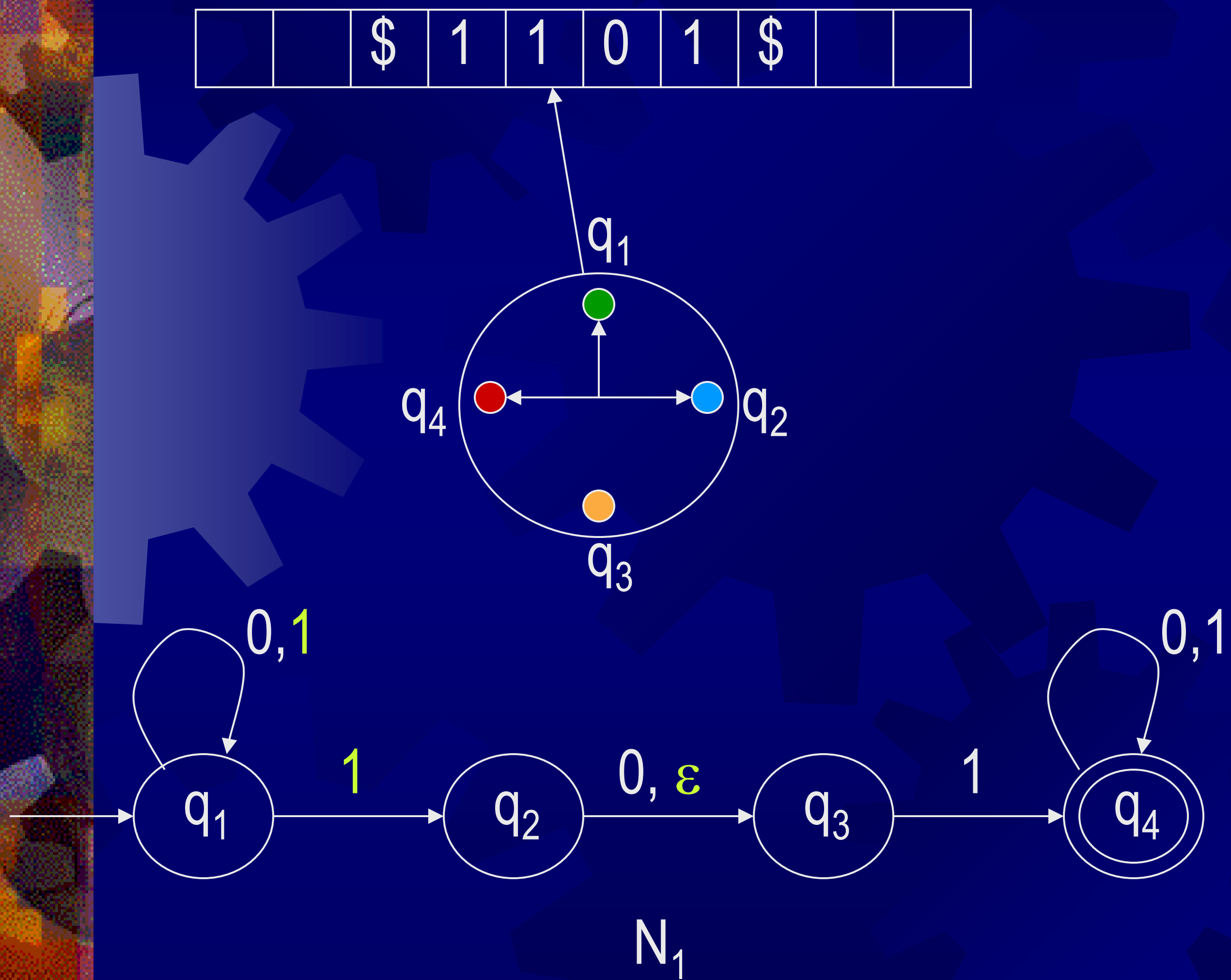
N_1在1101上计算(1)



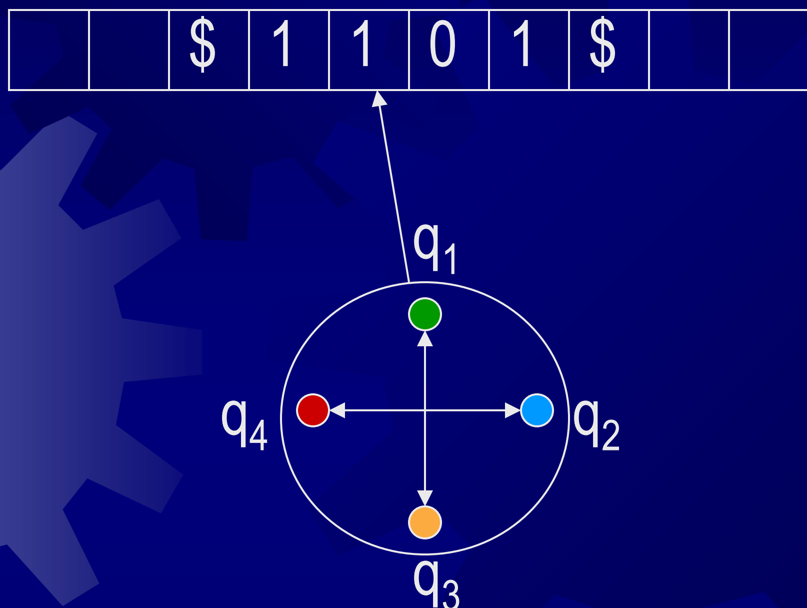
N_1在1101上计算(2)

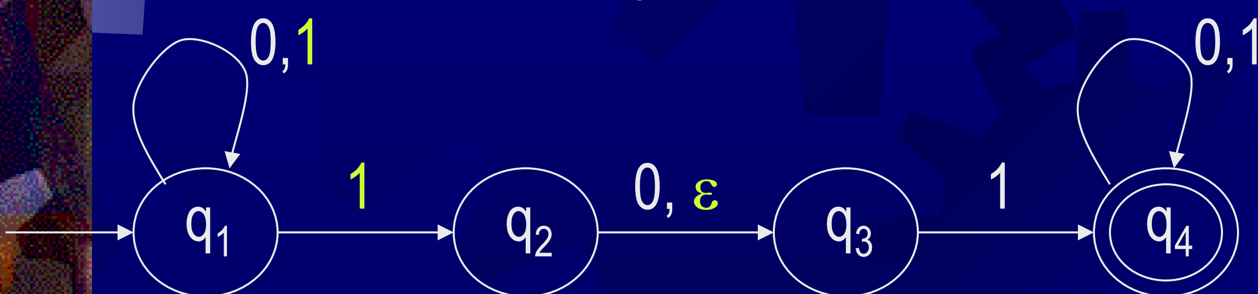


N_1在1101上计算(3)



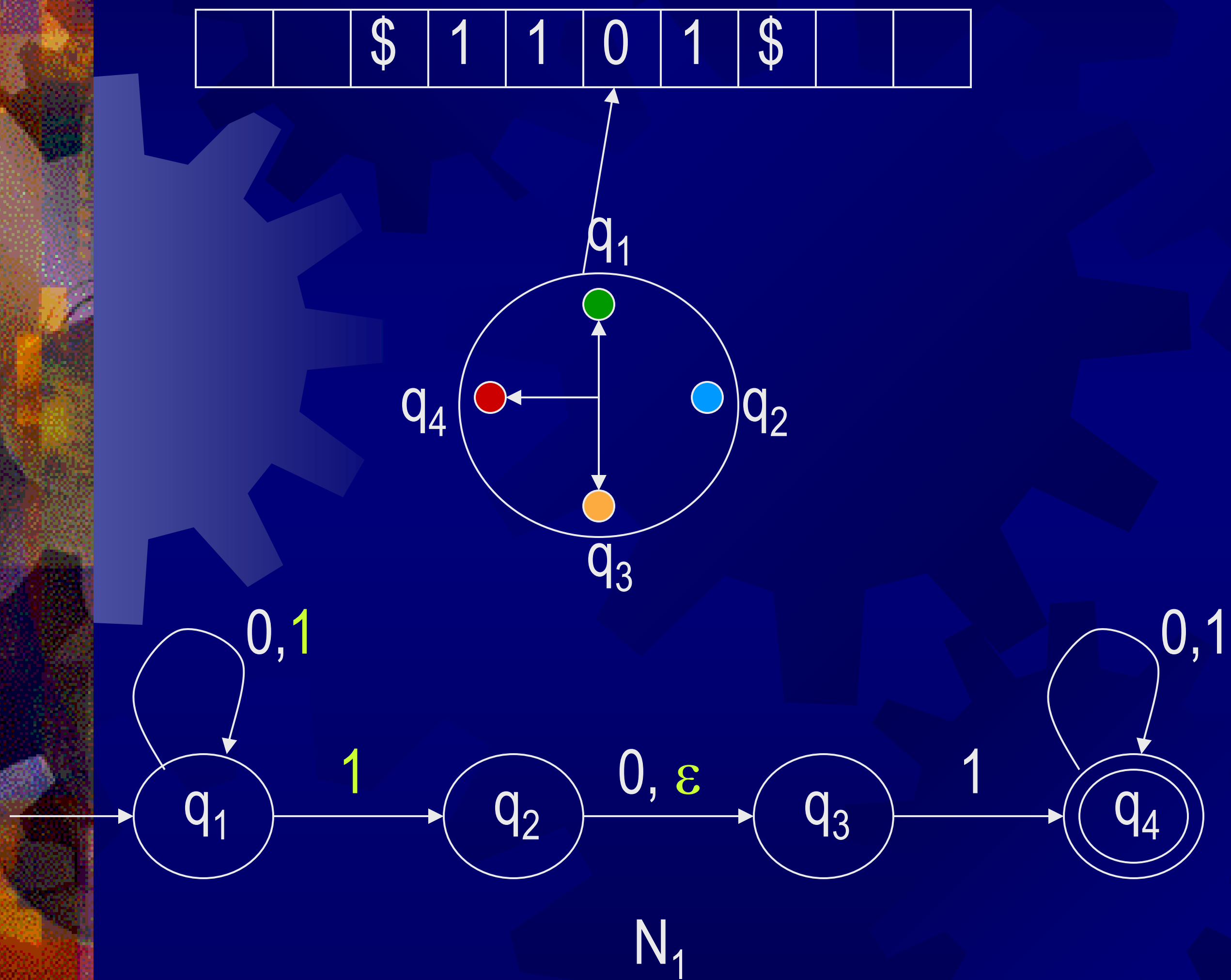
N_1在1101上计算(4)



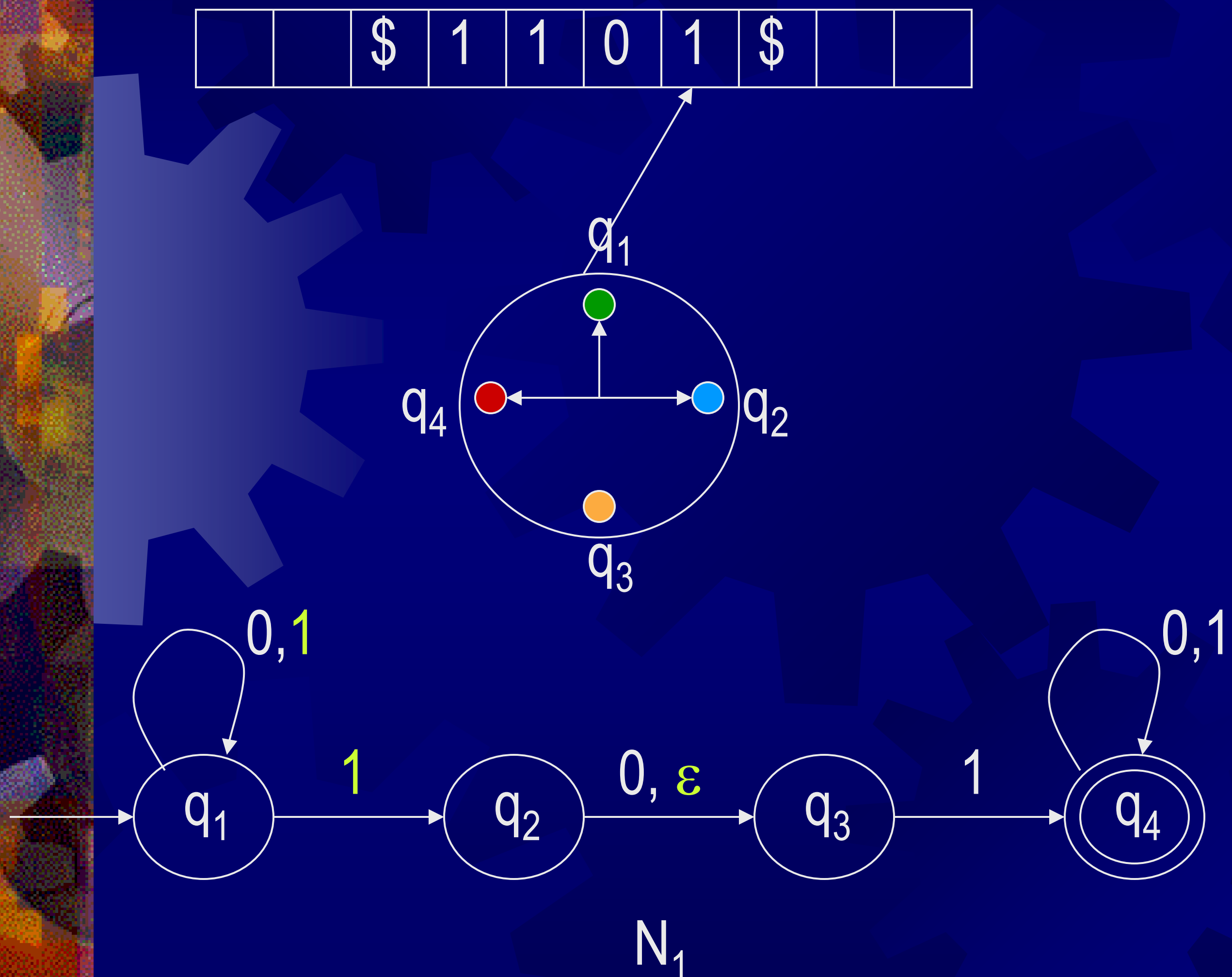


N_1

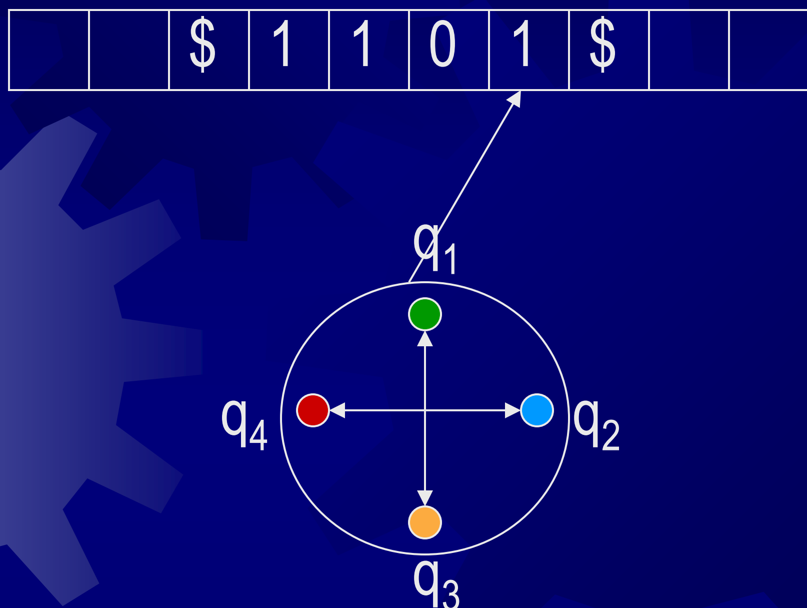
N_1在1101上计算(5)

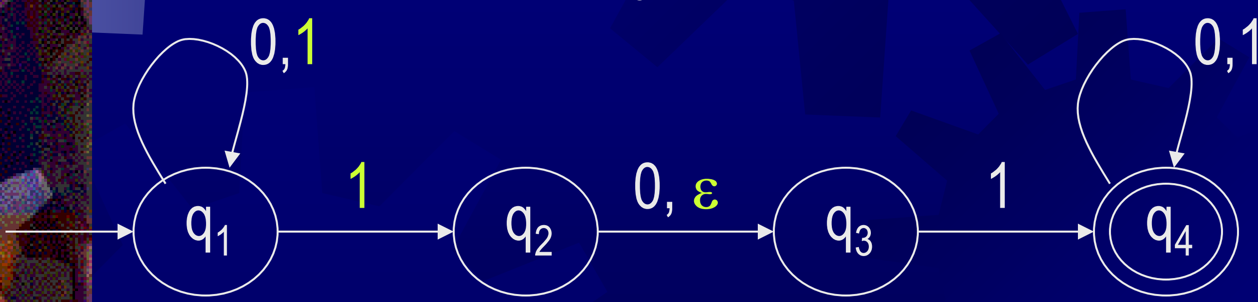


N_1在1101上计算(6)



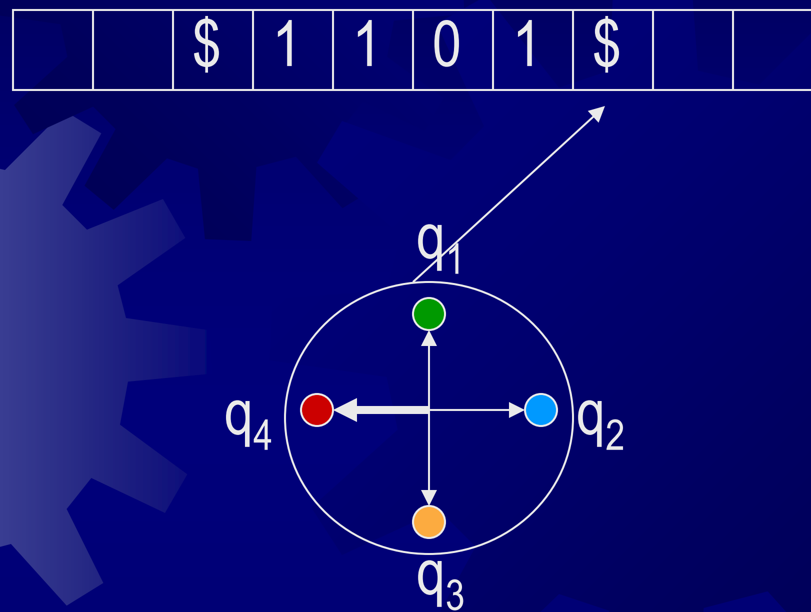
N_1在1101上计算(7)

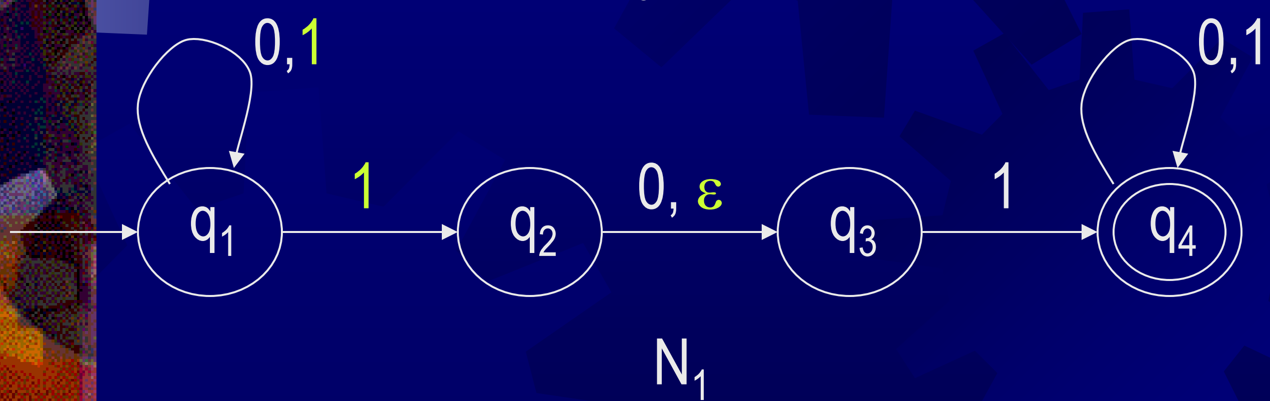




N_1

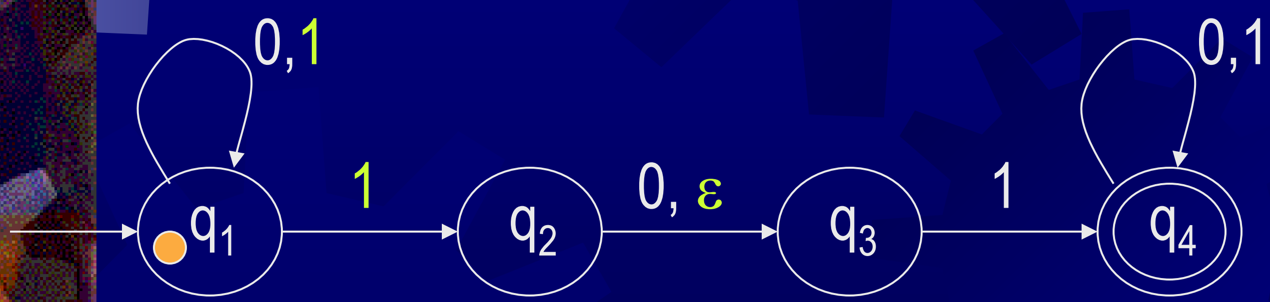
N_1在1101上计算(8)





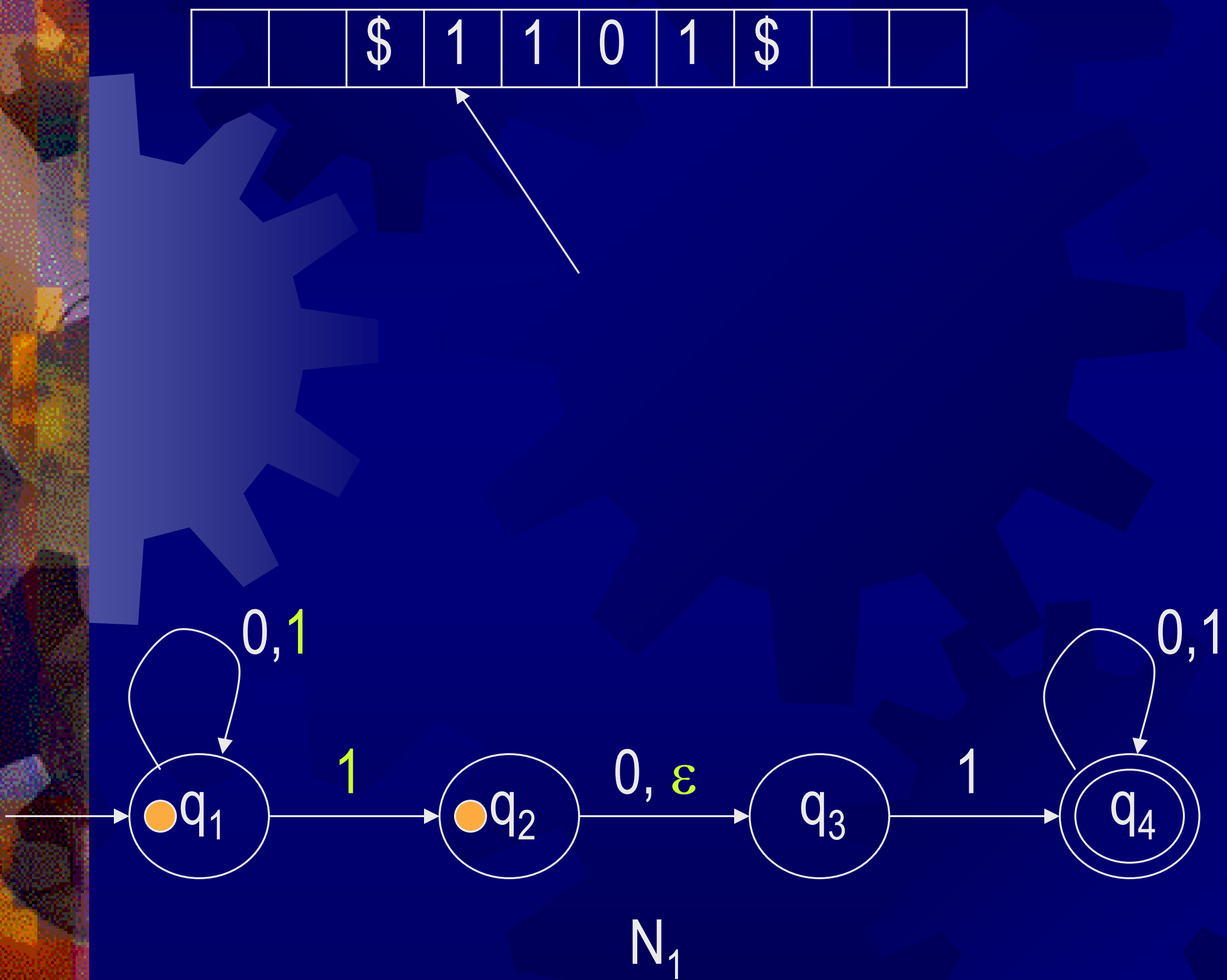
N_1在1101上计算(0)

		$	1	1	0	1	$		

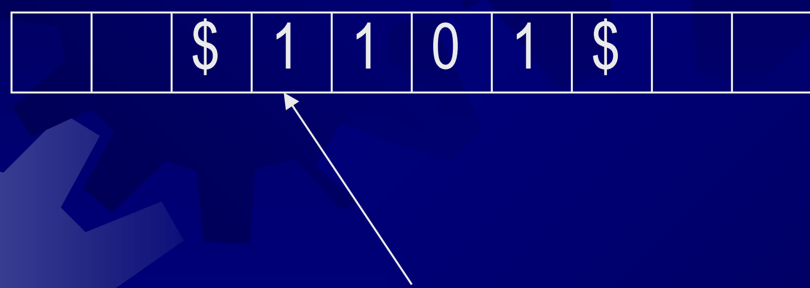


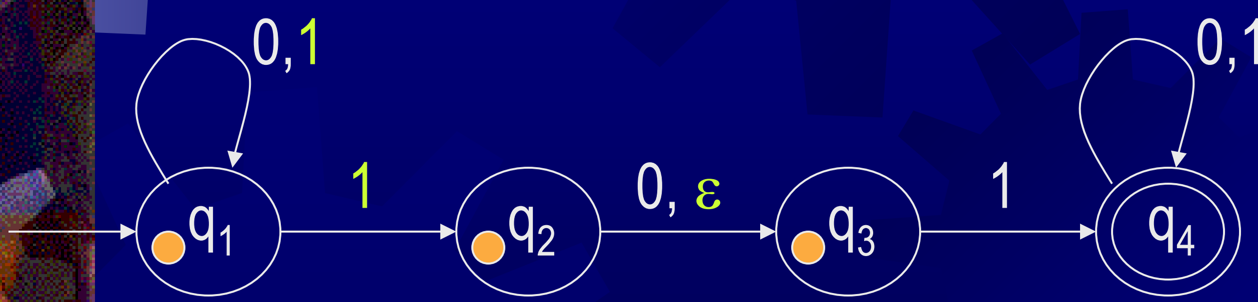
N_1

N_1在1101上计算(1)



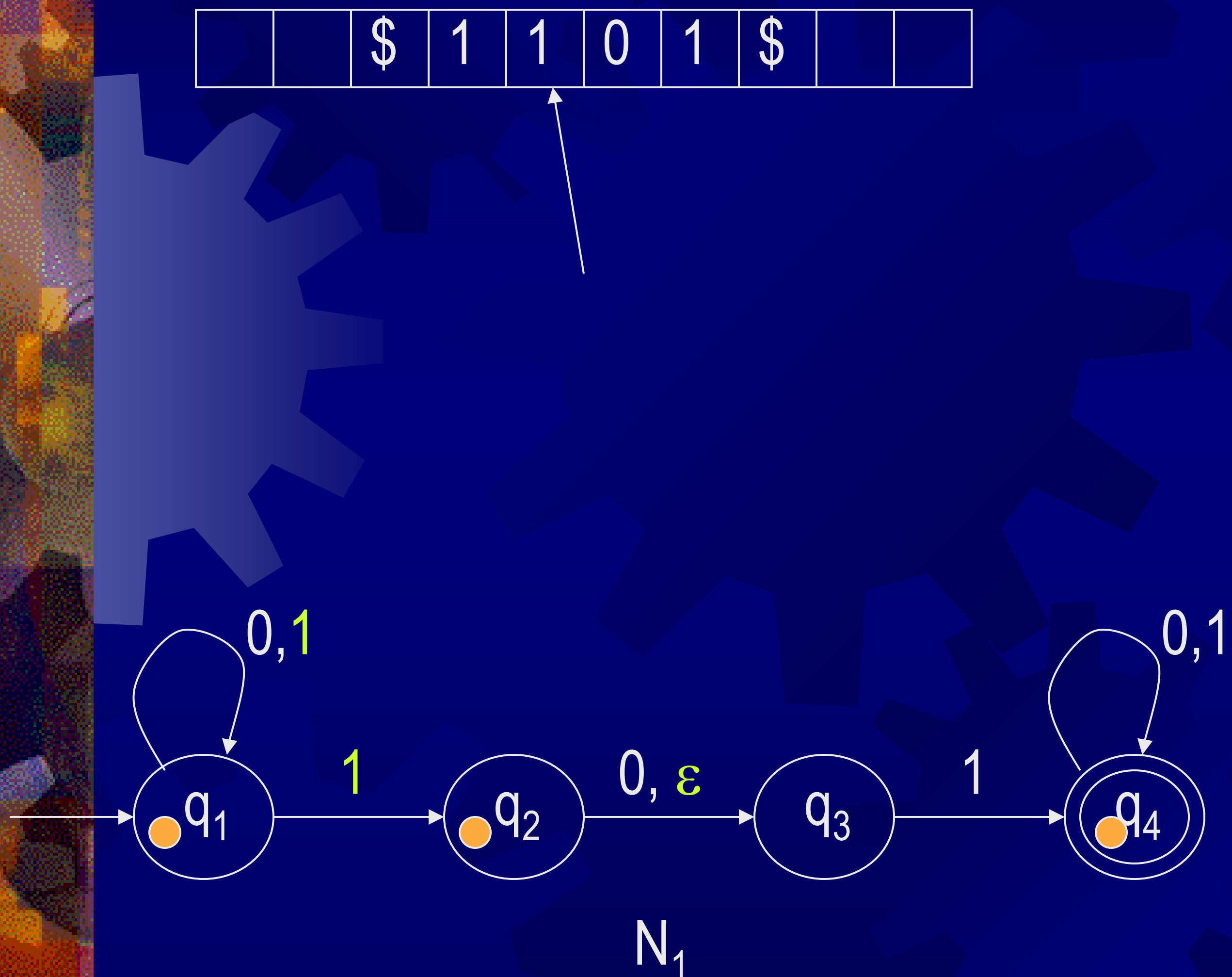
N_1在1101上计算(2)





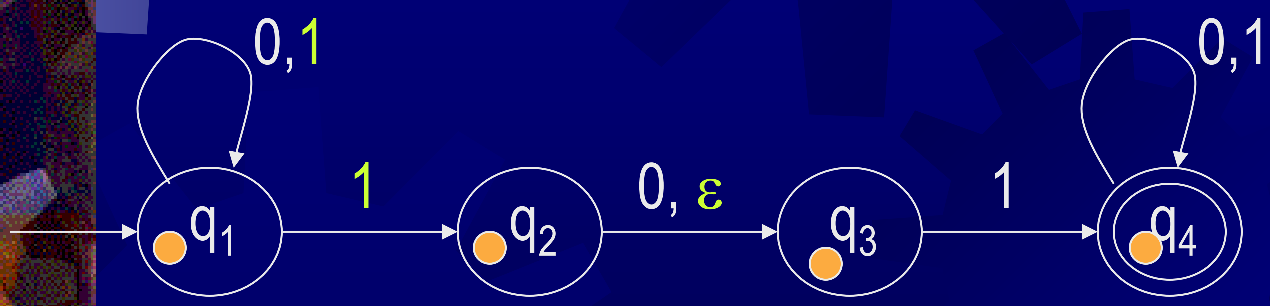
N_1

N_1在1101上计算(3)



N_1在1101上计算(4)

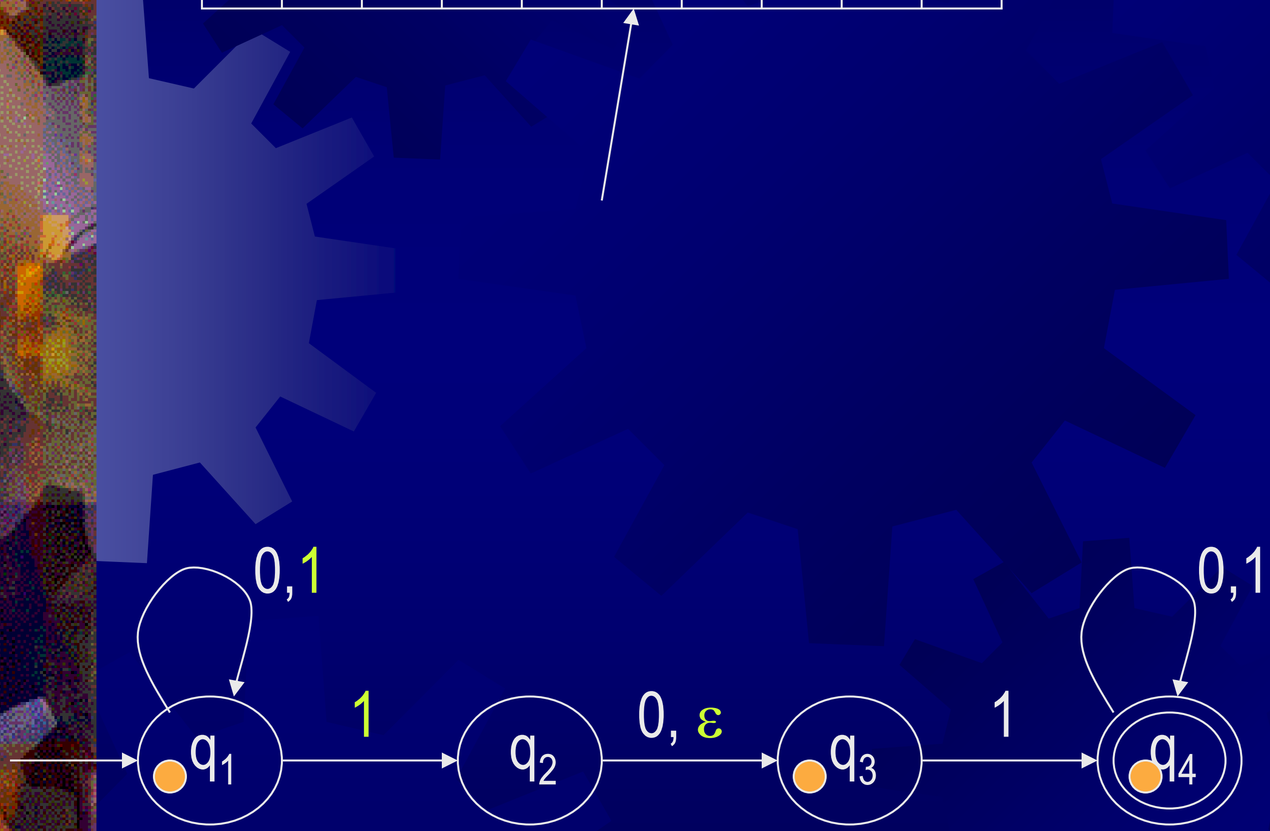
		$	1	1	0	1	$		



N_1

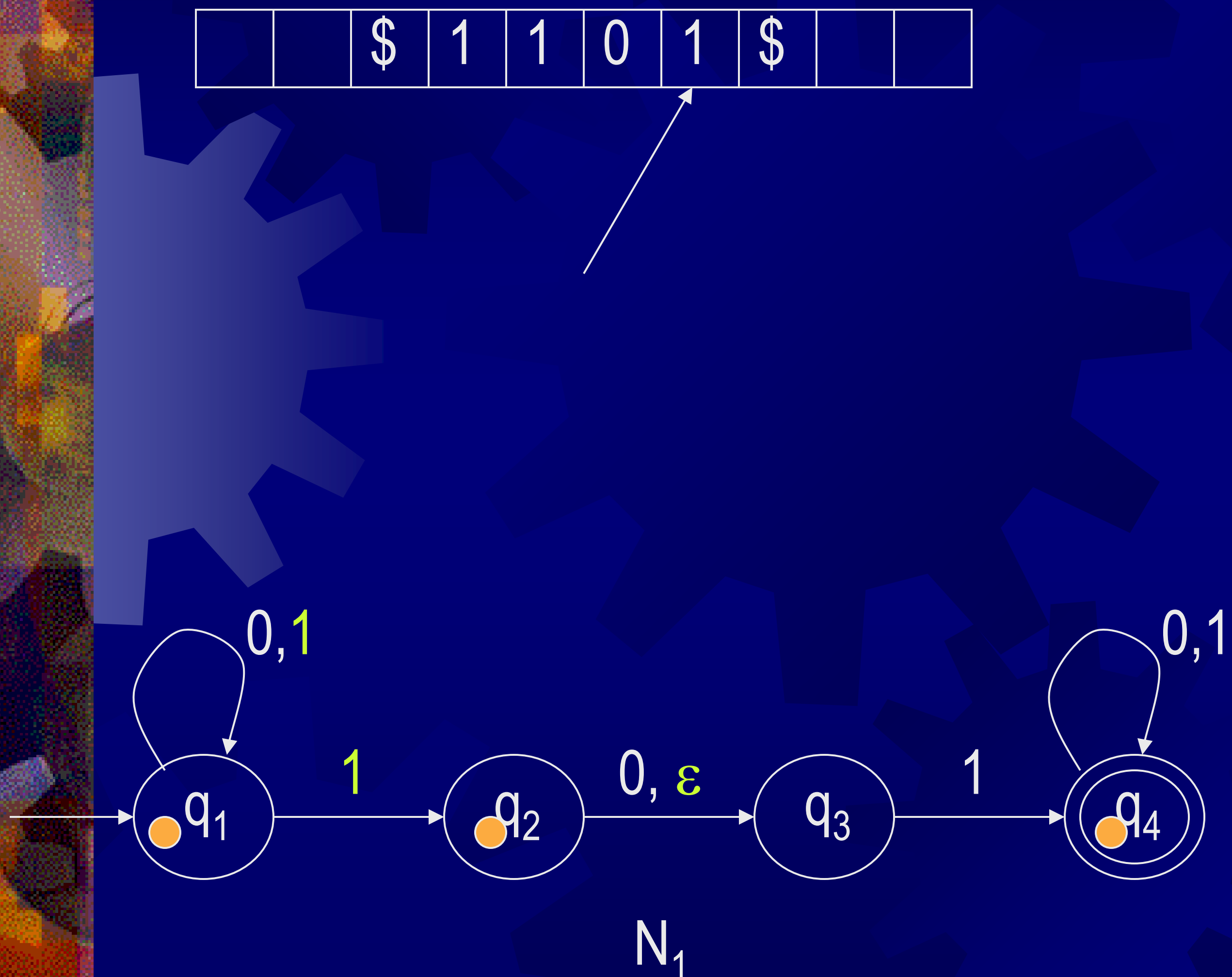
N_1在1101上计算(5)

		$	1	1	0	1	$		

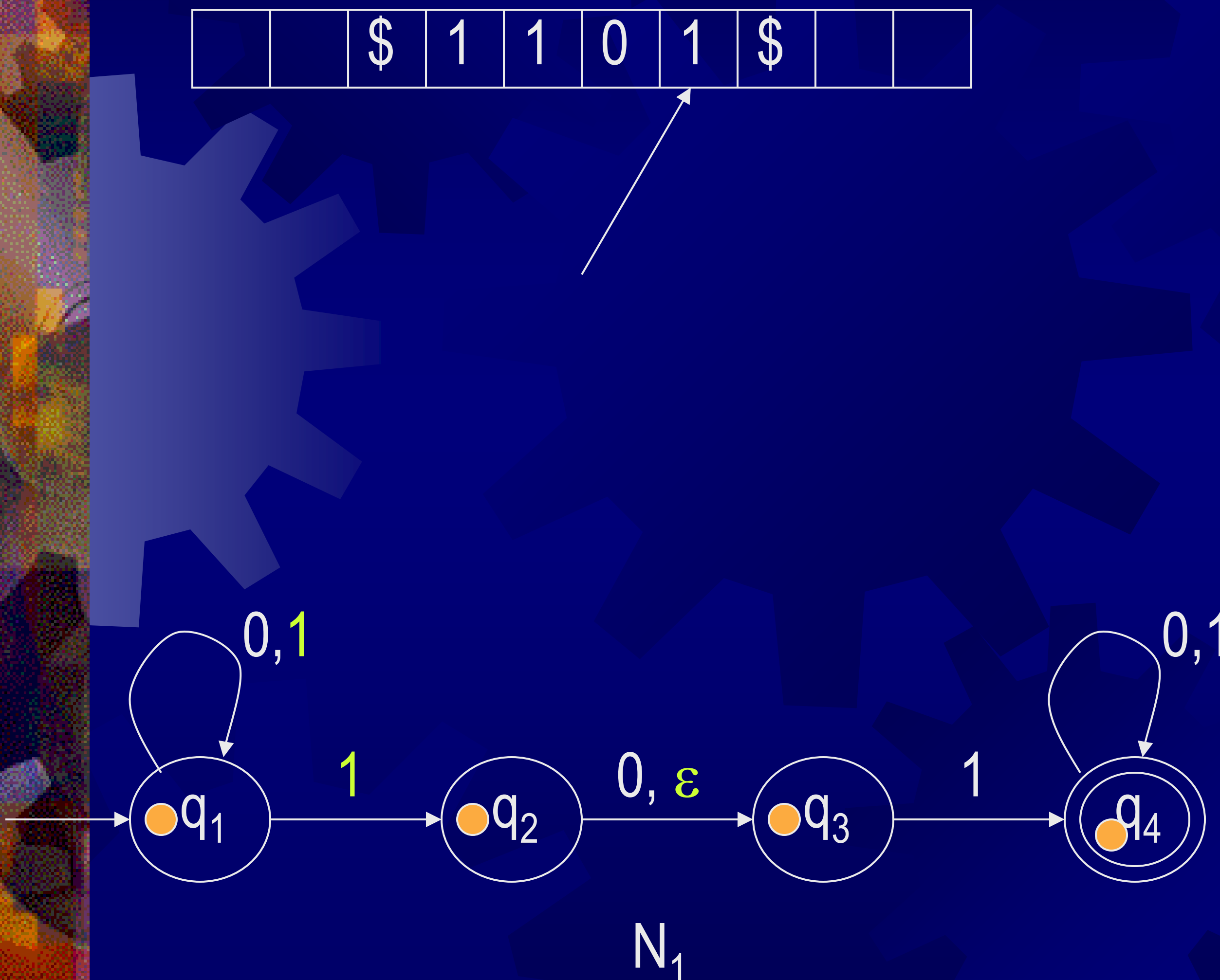


N1

N_1在1101上计算(6)

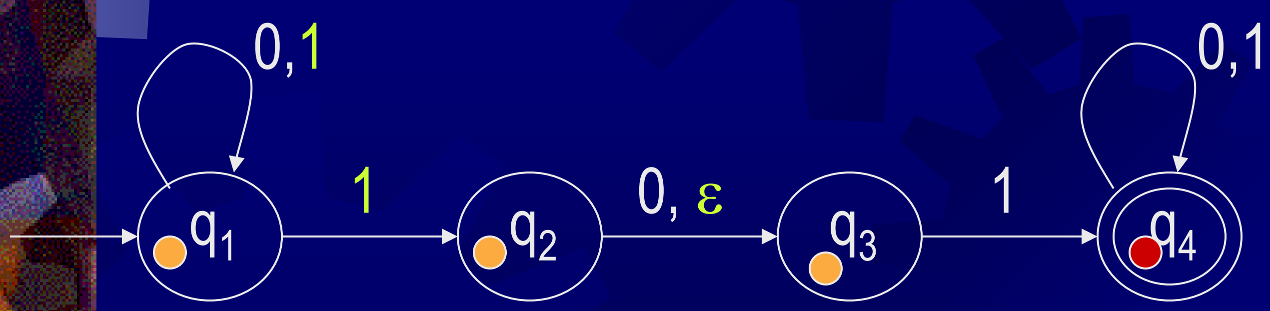


N_1在1101上计算(7)



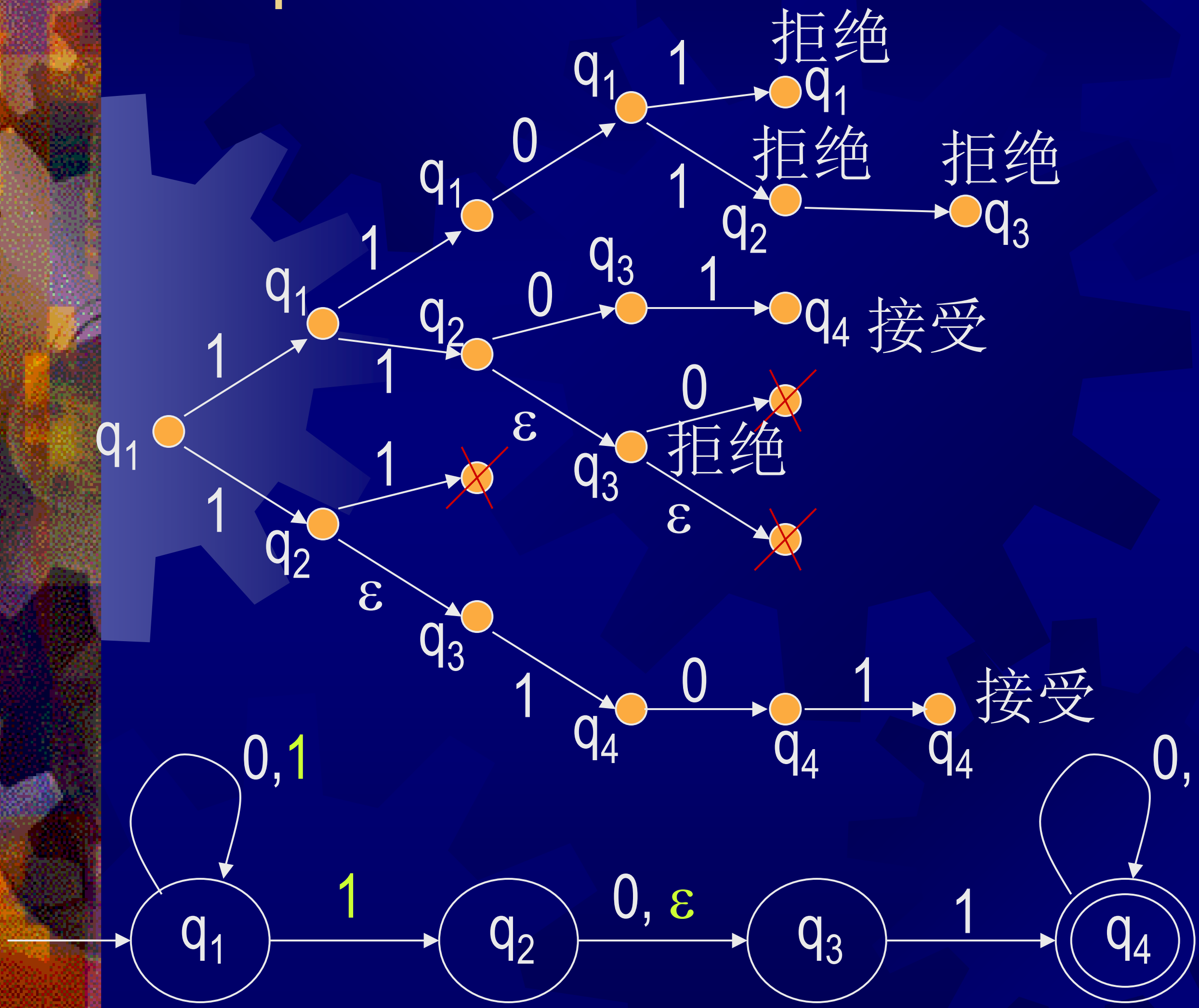
N_1在1101上计算(8)

		$	1	1	0	1	$		



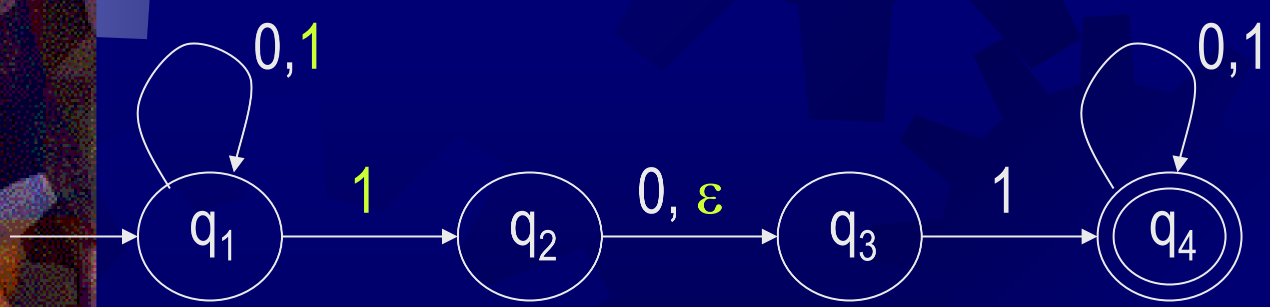
N_1

N_1在1101上的计算树



N_1接受的语言

$L(N_1)$ = { w | w包含子串101或11 }



N_1

例2.14

- $L(N_2)=\{w|w$倒数第3个字母为$1\}$

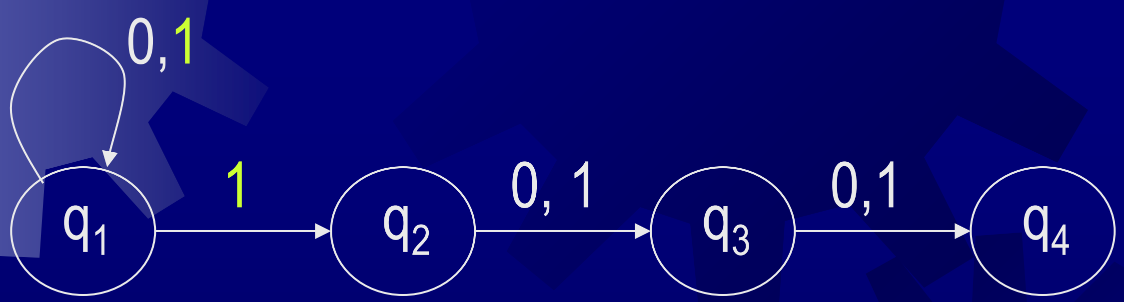
$\Sigma=\{0,1\}$

例2.14

- $L(N_2)$={w|w倒数第3个字母为1}
  $\Sigma=\{0,1\}$
- 非确定型: 猜测倒数第3个字母

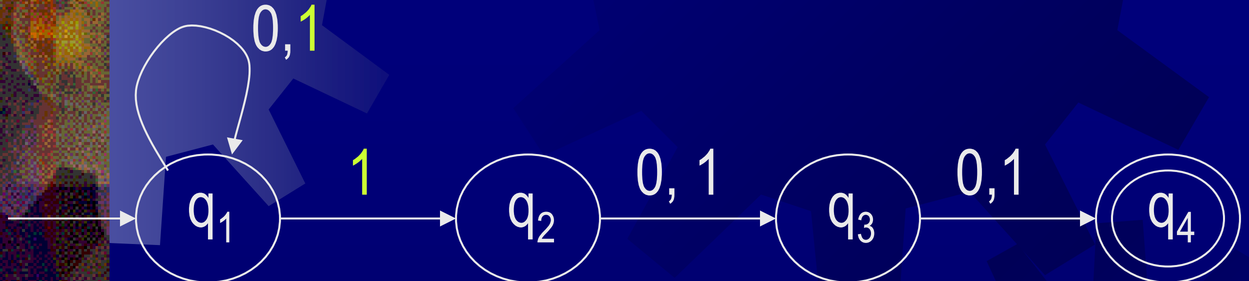
例2.14

- $L(N_2)$={w|w倒数第3个字母为1}
  Σ={0,1}
- 非确定型: 猜测倒数第3个字母



例2.14

- $L(N_2)$={w|w倒数第3个字母为1}
  Σ={0,1}
- 非确定型: 猜测倒数第3个字母



N_2

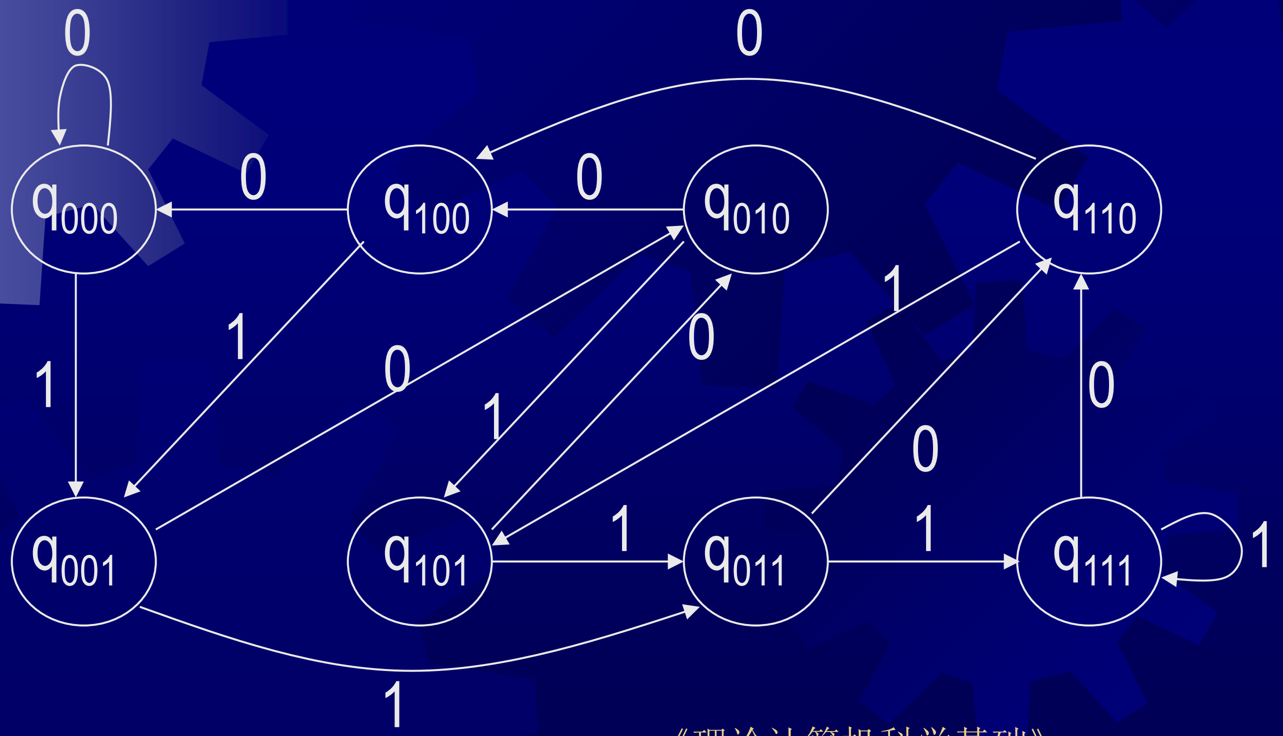
例2.14

- $L(N_2)$={w|w倒数第3个字母为1}

  Σ={0,1}
- 确定型: 记忆最后3个字母

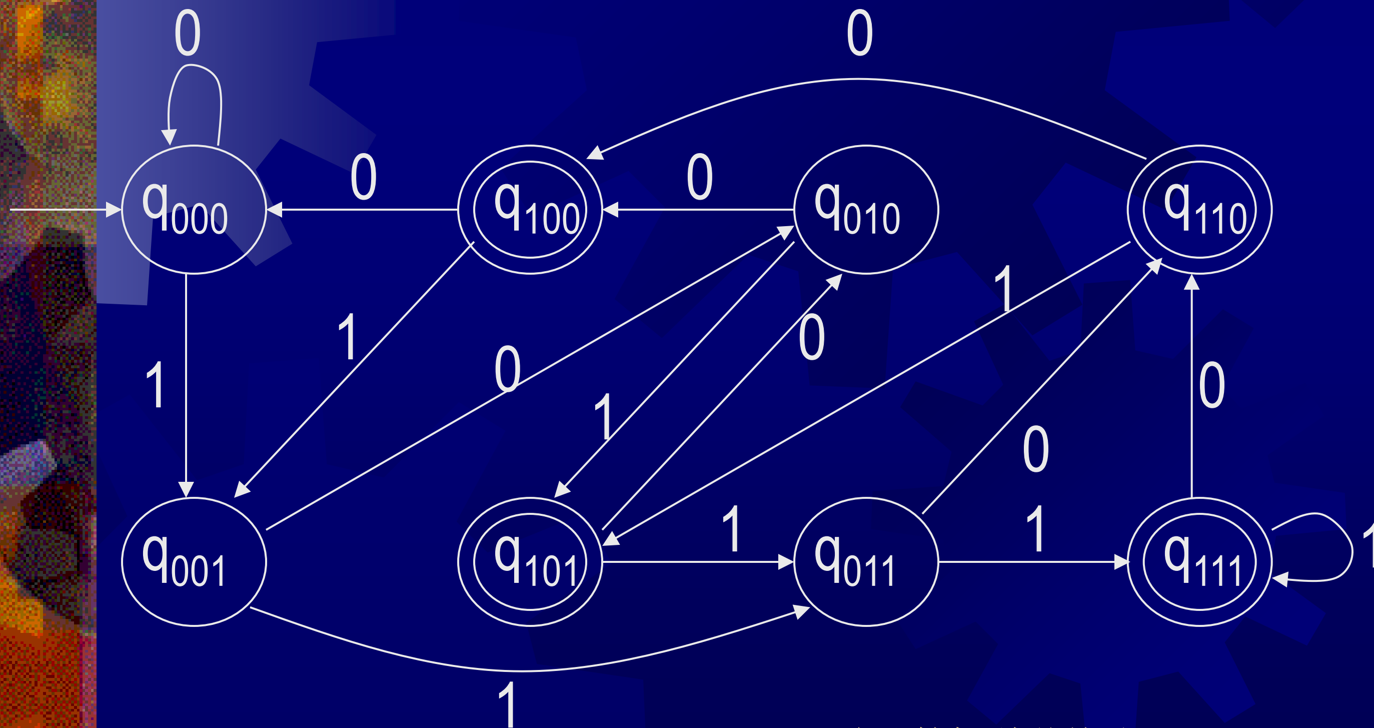
例2.14

- $L(N_2)$={w|w倒数第3个字母为1}
  Σ={0,1}
- 确定型: 记忆最后3个字母

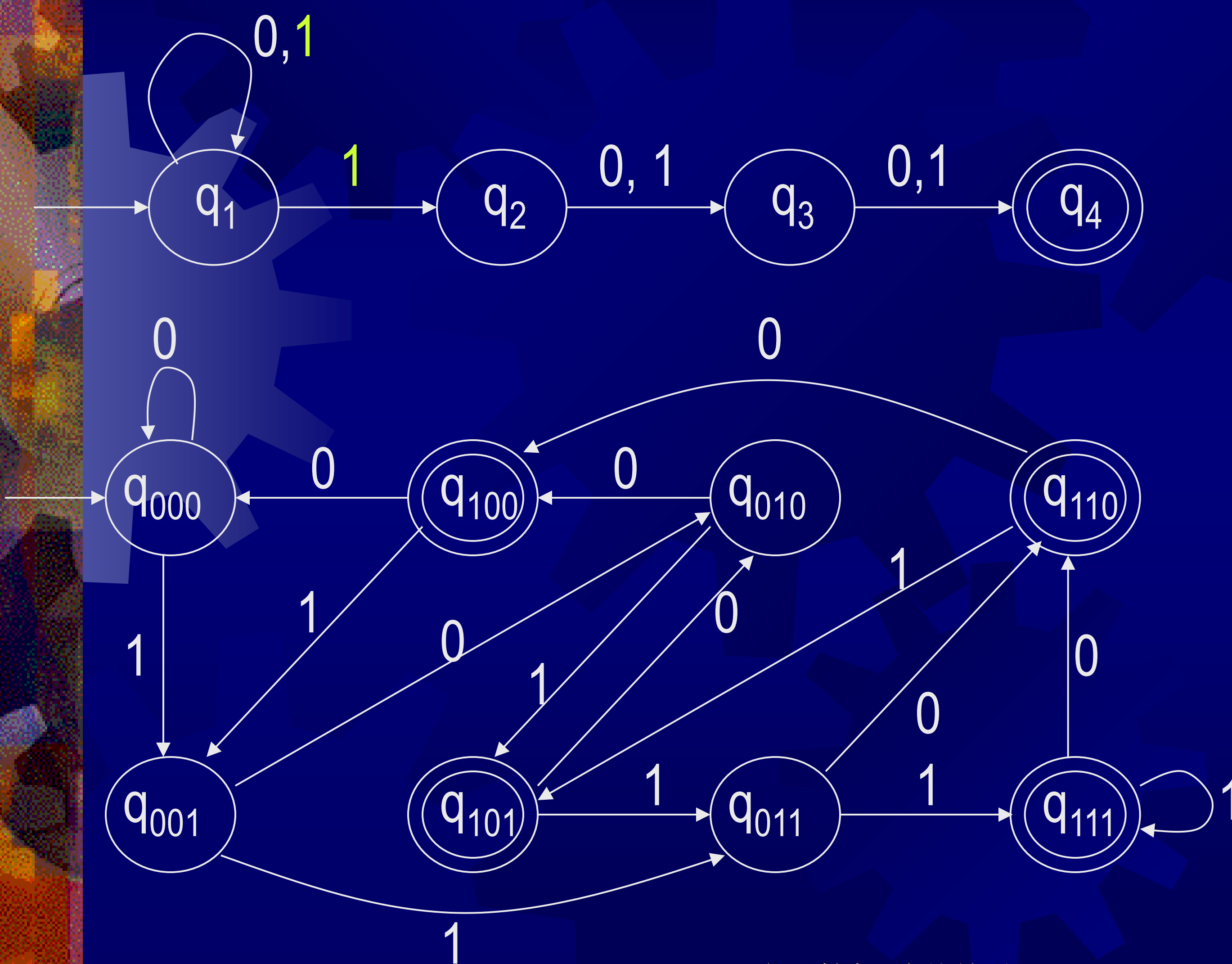


例2.14

- $L(N_2)$={w|w倒数第3个字母为1}
  Σ={0,1}
- 确定型: 记忆最后3个字母

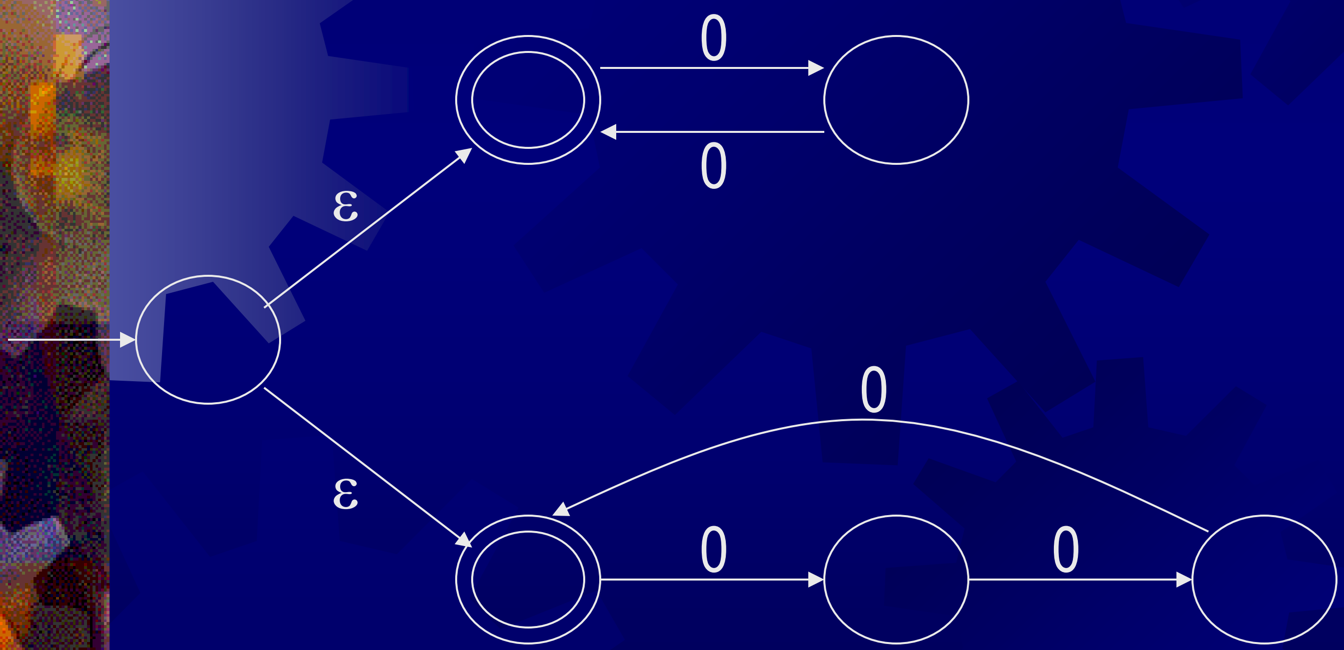


例2.14(比较)


0,1
1
0, 1
0,1
q1
q2
q3
q4
0
0
q000
0
q100
0
q010
q110
1
1
0
1
0
0
1
0
q001
q101
1
q011
1
q111
1
1

例2.15

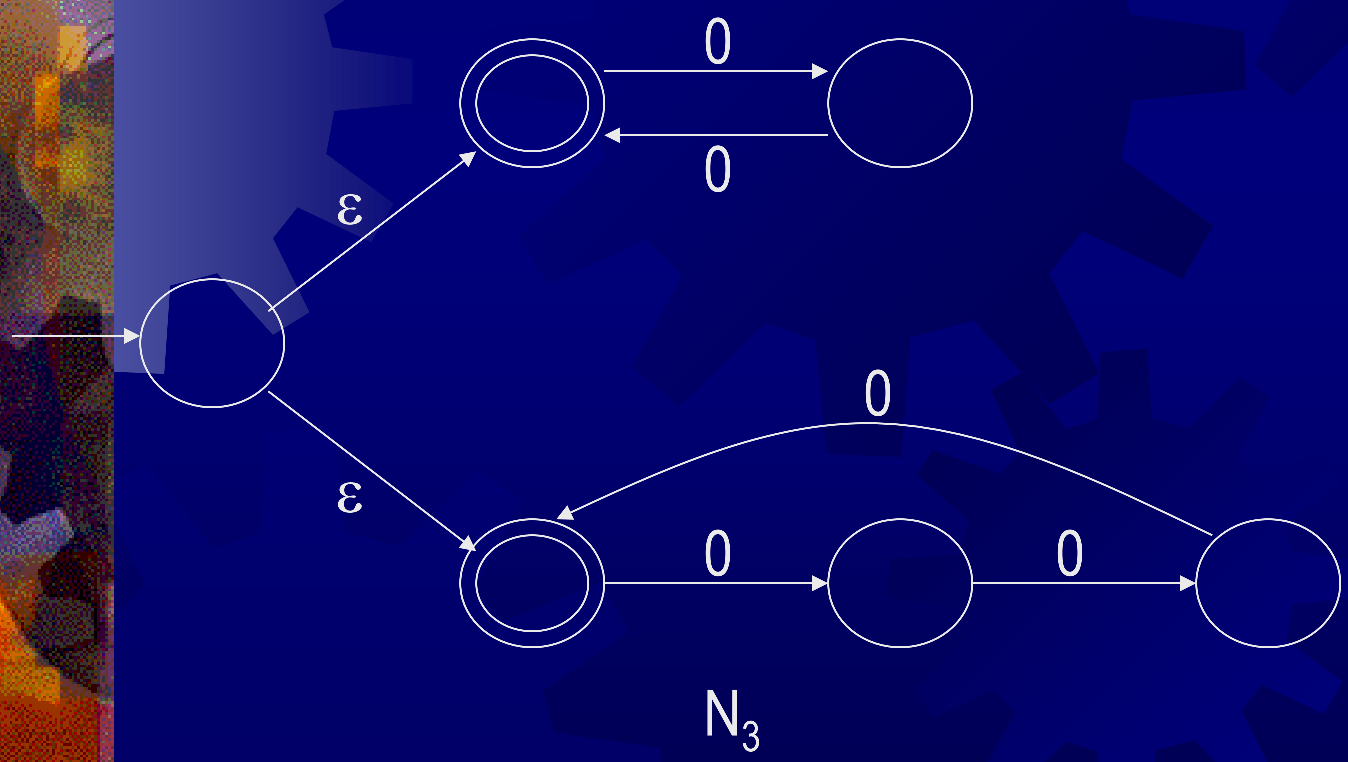
- 一元字母表上语言称为筹码集(Tally Sets)



N_3

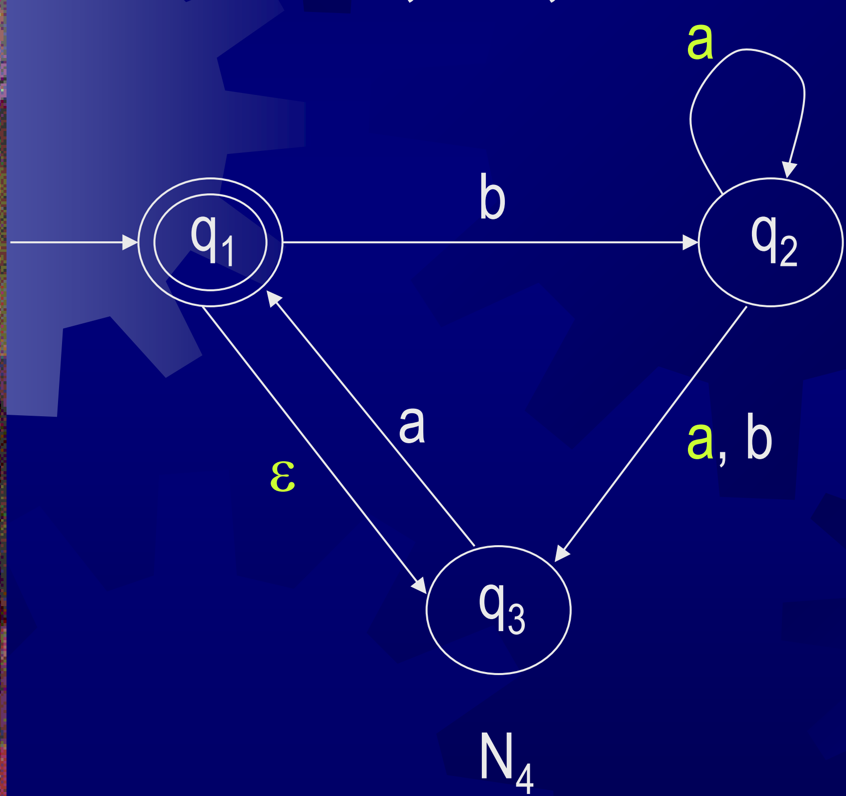
例2.15

- $L(N_3) = \{ 0^k \mid k$是2或3的倍数 $\}$



例2.16

- N_4接受ε, a, baba, baa,
  拒绝b, bb, babba.



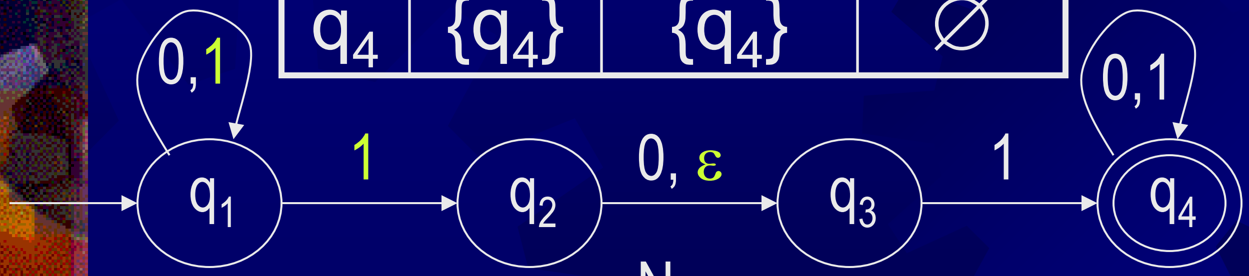
NFA的形式定义

- 定义2.17: 非确定型有穷自动机
  $N = (Q,\Sigma,\delta,q_0,F)$, 其中
  - Q: 有穷状态集
  - Σ: 输入字母表; （ $\Sigma_\varepsilon=\Sigma\cup\{\varepsilon\}$ ）
  - $\delta: Q\times\Sigma_\varepsilon\rightarrow P(Q)$, 转移函数
  - $q_0\in Q$: 初始状态
  - $F\subseteq Q$: 接受状态(终结状态)

例2.18

- $N_1=(Q,\Sigma,\delta,q_1,F)$; $Q=\{q_1,q_2,q_3,q_4\}$; $\Sigma=\{0,1\}$; $F=\{q_4\}$; δ表

	0	1	ε
q_1	$\{q_1\}$	$\{q_1,q_2\}$	$\varnothing$
q_2	$\{q_3\}$	$\varnothing$	$\{q_3\}$
q_3	$\varnothing$	$\{q_4\}$	$\varnothing$
q_4	$\{q_4\}$	$\{q_4\}$	$\varnothing$



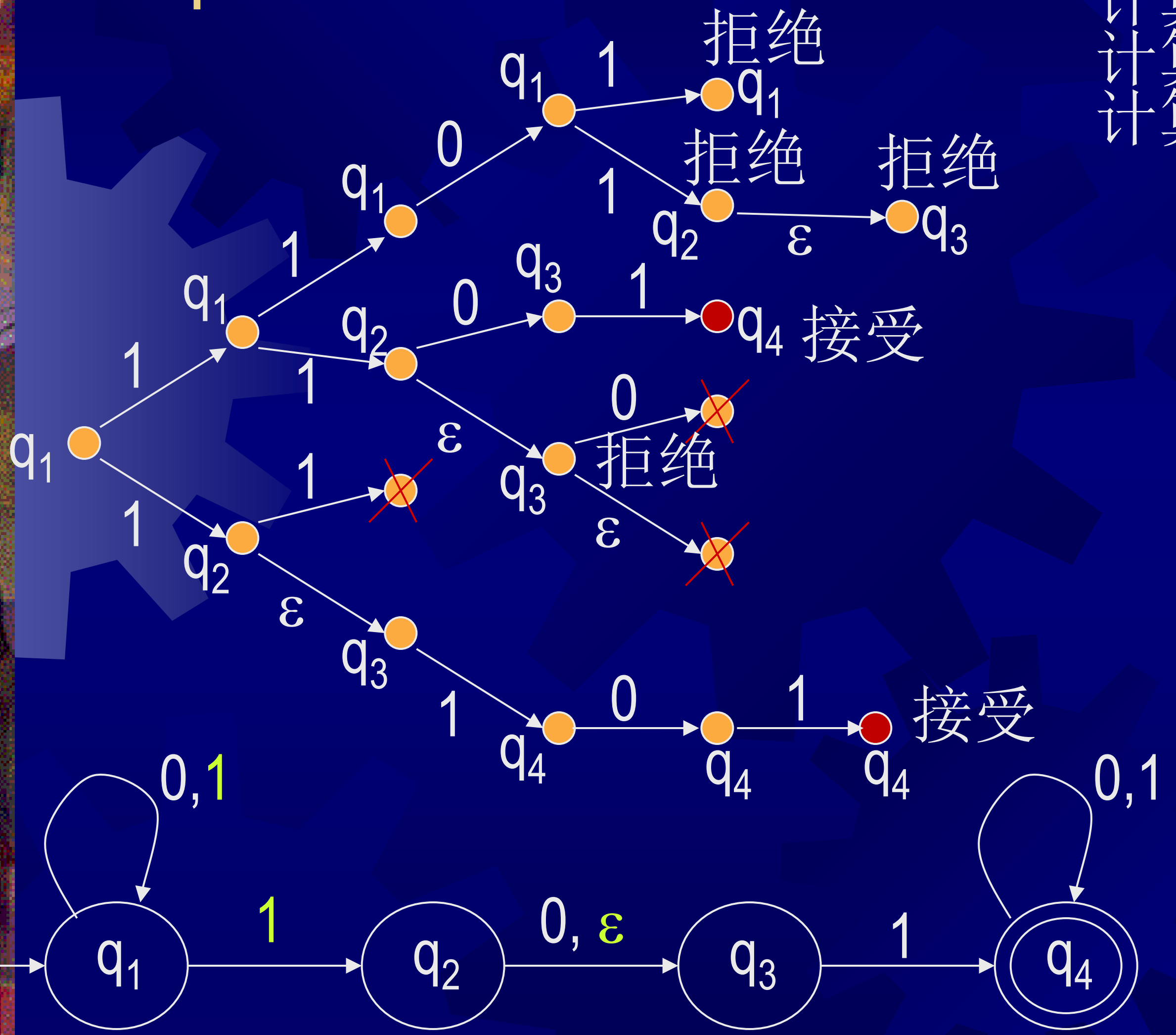
N_1

NFA计算的形式定义

- NFA $N=(Q,\Sigma,\delta,q_0,F)$
  - 输入$w=w_1w_2\ldots w_m$
- 计算: 状态序列 $r_0,r_1,\ldots,r_m$
  - $r_0=q_0$
  - $r_{i+1}\in\delta(r_i,w_{i+1})$ $(i=0,1,\ldots,m-1)$
- 接受计算:
  - $r_m\in F$
- M接受w: 存在接受计算
  - $L(M)=\{x \mid M接受x\}$

N_1在1101上的计算

计算1: q_1,q_1,q_1,q_1,q_1
计算2: q_1,q_1,q_1,q_1,q_2
计算3: q_1,q_1,q_1,q_1,q_2,q_3
计算4: q_1,q_1,q_2,q_3,q_4
计算5: q_1,q_1,q_2,q_3
计算6: q_1,q_2
计算7: q_1,q_2,q_3,q_4,q_4,q_4



NFA与DFA的等价性

- 等价: 两台机器识别同样的语言.

NFA与DFA的等价性

- 等价: 两台机器识别同样的语言.
- 定理2.19:每台NFA都有等价DFA.

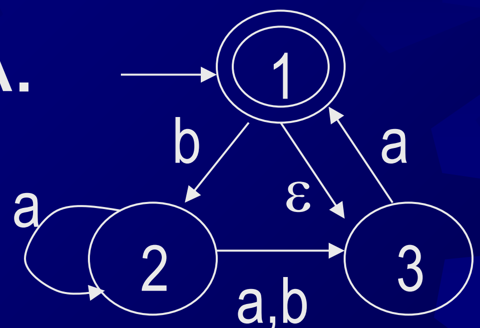
NFA与DFA的等价性

- 定理2.19:每台NFA都有等价DFA.
- 证明思路:
  - 给定NFA, 构造等价DFA
  - 用DFA模拟NFA
    - DFA记住NFA的所有分支
    - 设NFA有k个状态,
      则共有2^k个不同状态子集合
  - ε闭包:
    - 对每个状态子集合, 经ε移动
      可到达的新状态子集合

例2.21(续例2.16)

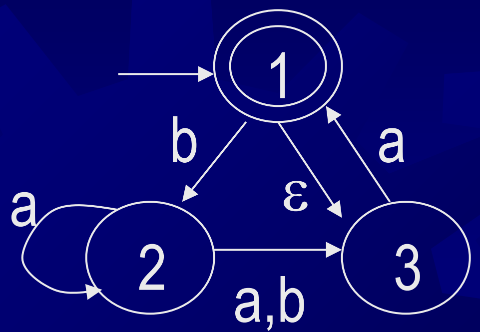
- $N_4=(\{1,2,3\},\{a,b\},\delta,1,\{1\})$

求等价的DFA.



例2.21

- 写出子集状态



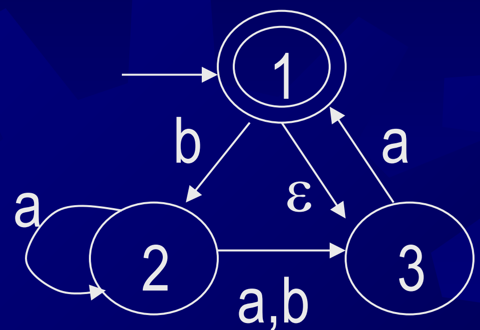
∅　{1}　{2}　{1,2}

{3}　{1,3}　{2,3}　{1,2,3}

例2.21

- 求ε闭包

E({1})={1,3}

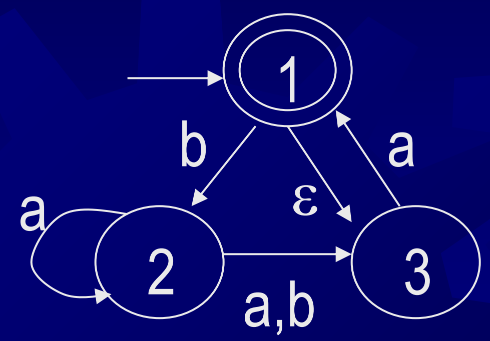


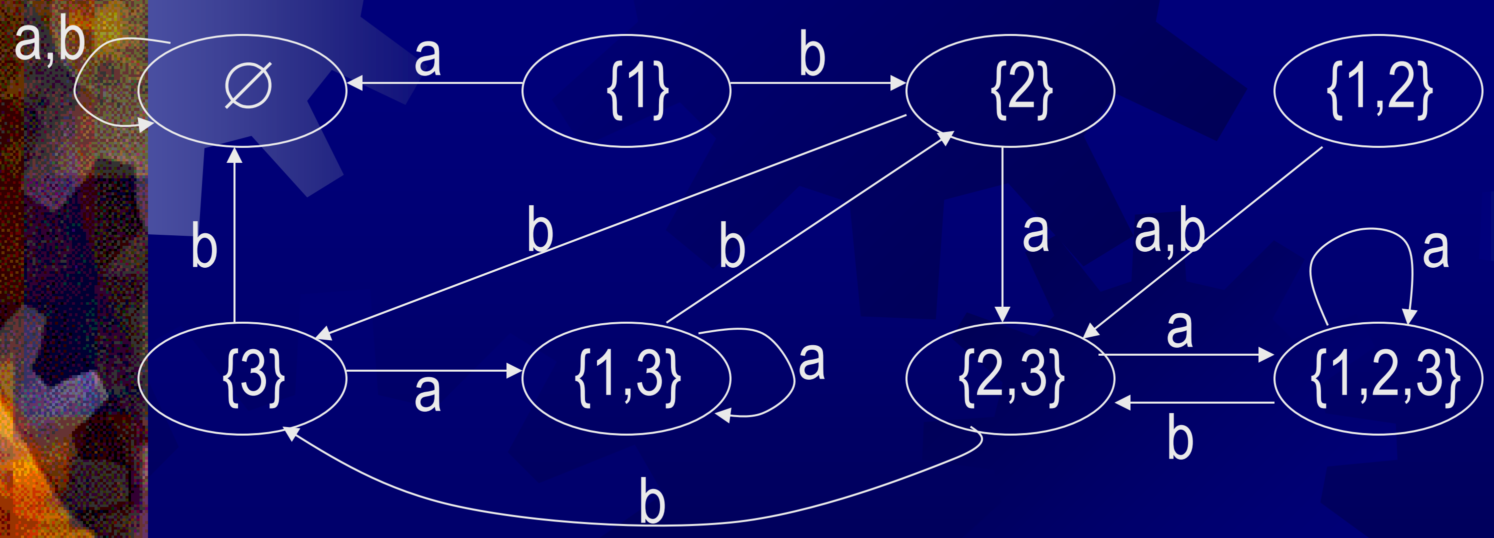
∅ {1} {2} {1,2}

{3} {1,3} {2,3} {1,2,3}

例2.21

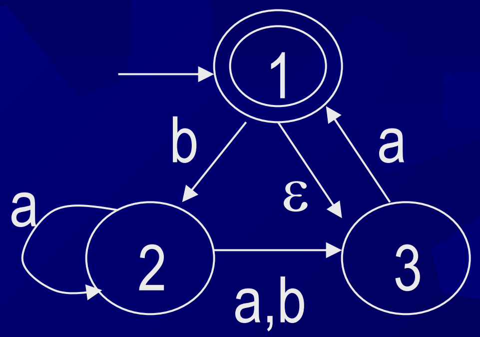
- 添加转移

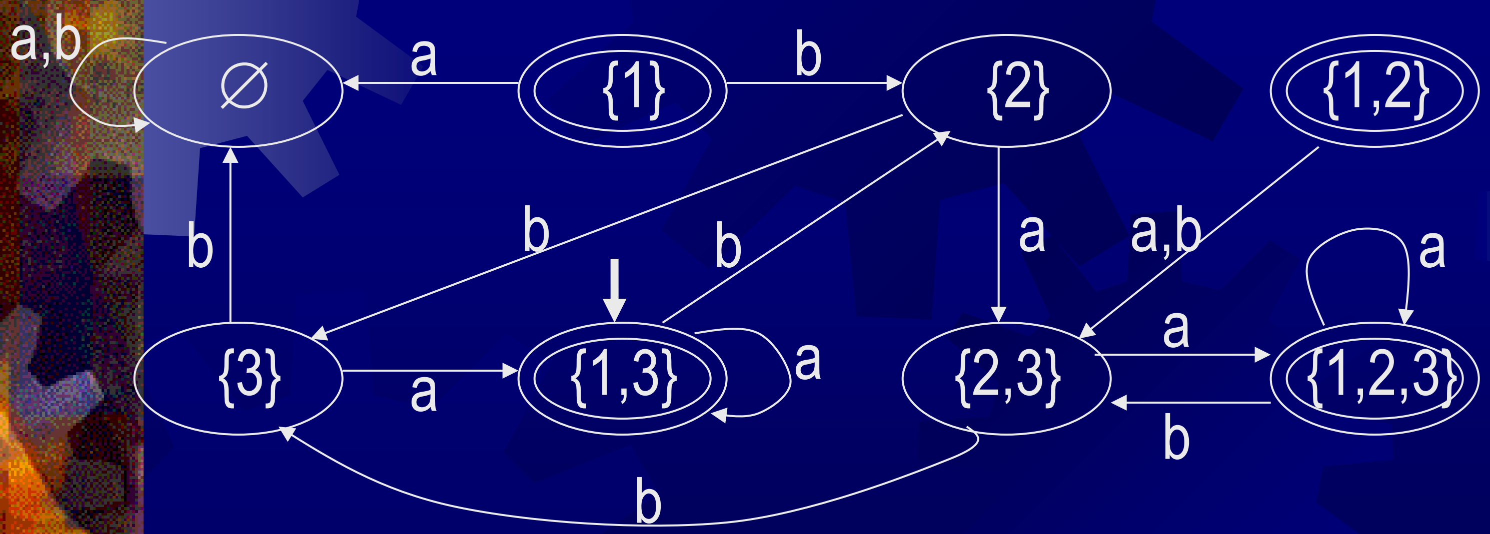




例2.21

- 初始状态
- 接受状态
- 不可达状态

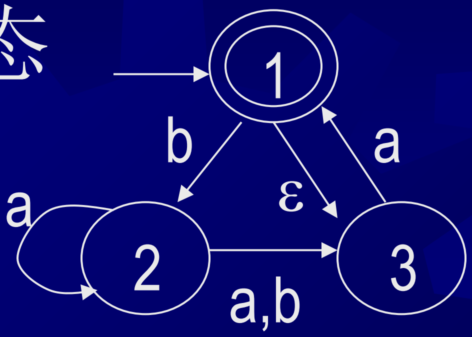


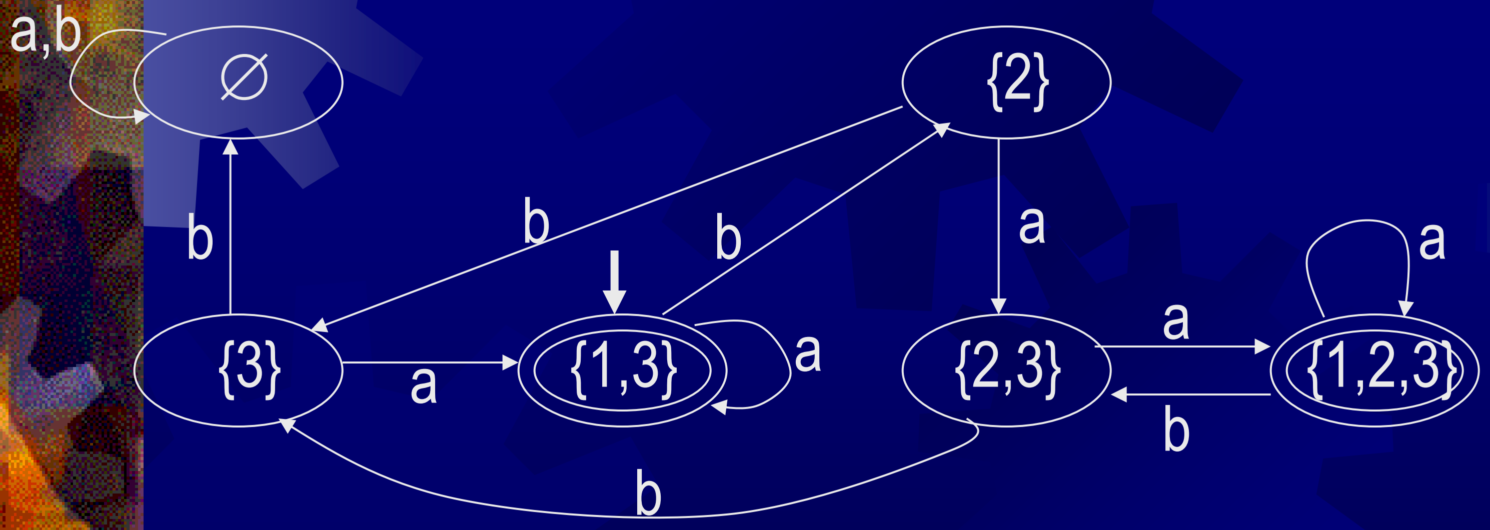


例2.21

- 删除不可达状态

{1}, {1,2};





NFA与DFA的等价性

- 定理2.19: 每台NFA都有等价DFA.
- 证明: 设NFA $N=(Q,\Sigma,\delta,q_0,F)$, 构造 DFA $M=(Q',\Sigma,\delta',q_0',F')$, $L(M)=L(N)$.

令$Q'=P(Q)$. 对$R\in Q'$和$a\in\Sigma$,

$E(R)=\{q \mid$ 从R出发沿0个或多个ε移动可达q $\}$;

$\delta'(R,a)=\bigcup_{r\in R} E(\delta(r,a))$. $q_0'=E(\{q_0\})$.

$F'=\{R\in Q' \mid R\cap F\neq\varnothing\}$. #

推论2.20

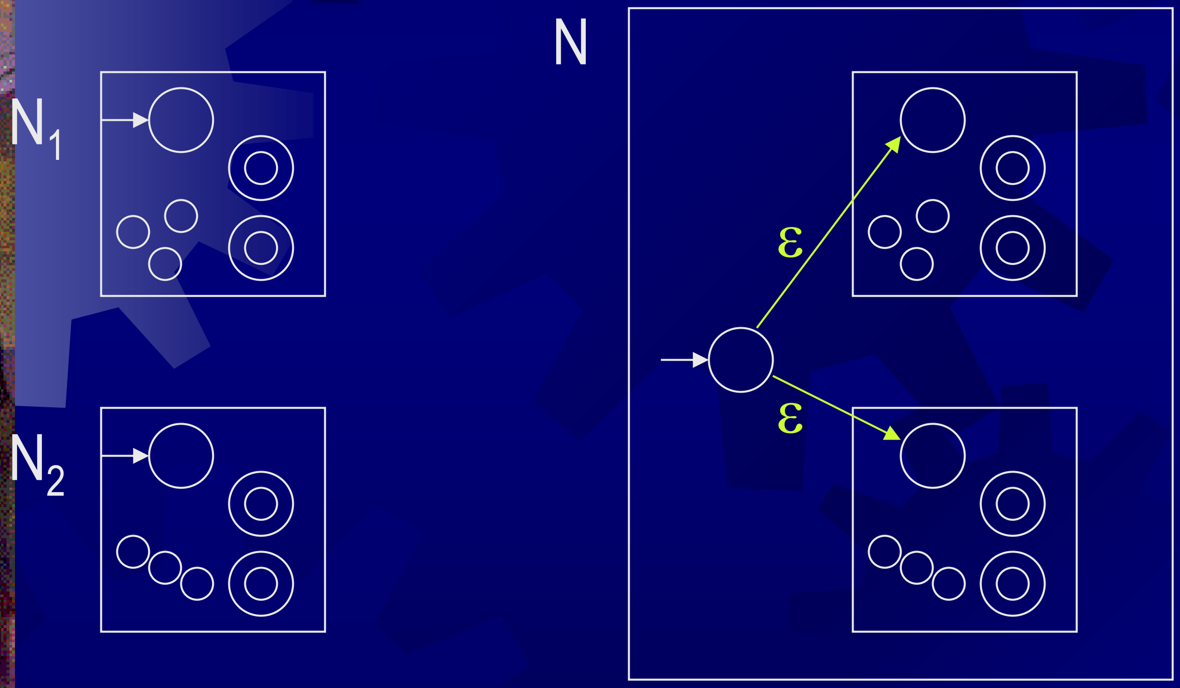
- 推论2.20: 一个语言是正则的,
当且仅当有一台NFA识别它.

对正则运算的封闭性

- 定理2.22: 正则语言对并封闭.

对并运算的封闭性

- 定理2.22: 正则语言对并封闭.
- 证明思路:



对并运算的封闭性

- 定理2.22: 正则语言对并封闭.
- 证明:设NFA $N_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$ 识别A_i, i=1,2. 构造NFA $N=(Q,\Sigma,\delta,q_0,F)$识别$A_1\cup A_2$. 令$Q=Q_1\cup Q_2\cup\{q_0\}$; $F=F_1\cup F_2$.

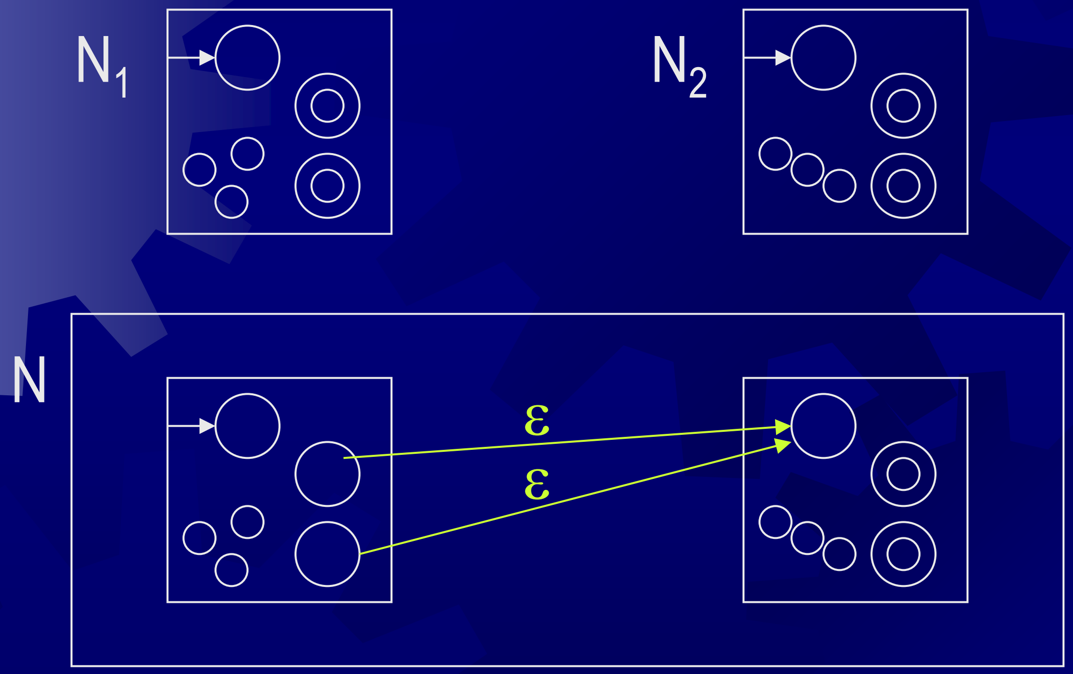
$$\delta(q,a)=\begin{cases}\delta_1(q,a), & 若 q\in Q_1\\ \delta_2(q,a), & 若 q\in Q_2\\ \{q_1,q_2\}, & 若 q=q_0\wedge a=\varepsilon\\ \varnothing, & 若 q=q_0\wedge a\neq\varepsilon\end{cases}$$

对连接运算的封闭性

- 定理2.23: 正则语言对连接封闭.

对连接运算的封闭性

- 定理2.23: 正则语言对连接封闭.
- 证明思路:



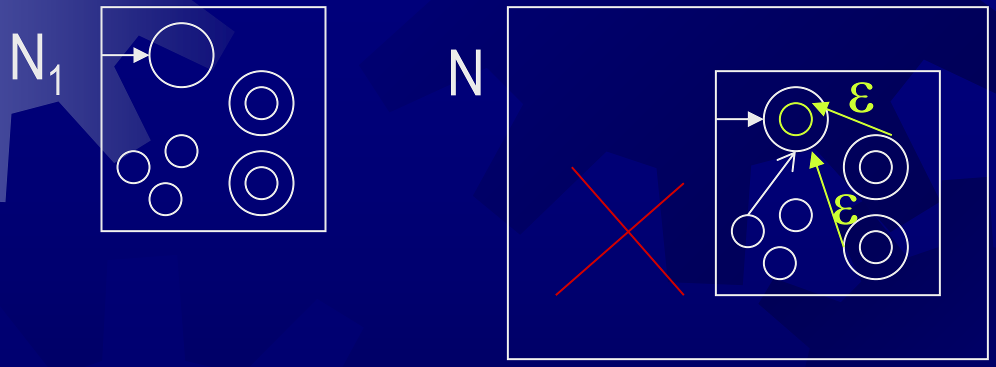
对连接运算的封闭性

- 定理2.23: 正则语言对连接封闭.
- 证明:设NFA $N_i=(Q_i,\Sigma,\delta_i,q_i,F_i)$ 识别A_i,i=1,2. 构造NFA $N=(Q,\Sigma,\delta,q_1,F_2)$识别$A_1A_2$. 令$Q=Q_1\cup Q_2$;

$$\delta(q,a)=\begin{cases}\delta_1(q,a), & 若q\in Q_1且q\notin F_1\\ \delta_1(q,a), & 若q\in F_1且a\neq\varepsilon\\ \delta_1(q,a)\cup\{q_2\}, & 若q\in F_1且a=\varepsilon\\ \delta_2(q,a), & 若q\in Q_2\end{cases}$$

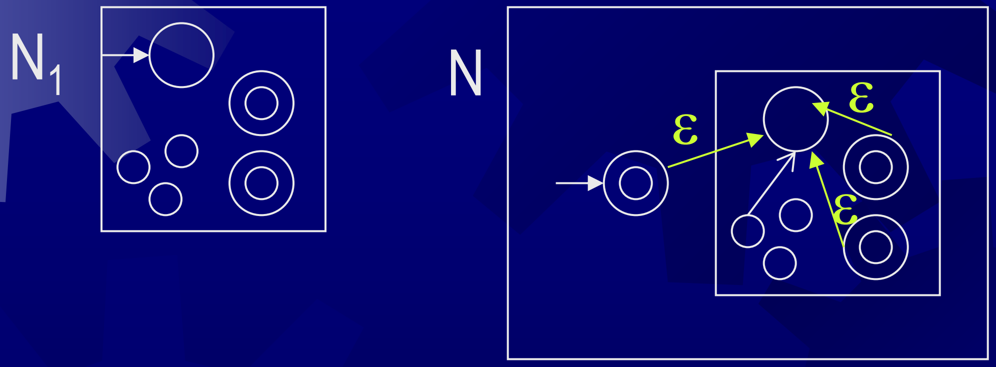
对星号运算的封闭性

- 定理2.24: 正则语言对星号封闭.
- 证明思路:



对星号运算的封闭性

- 定理2.24: 正则语言对星号封闭.
- 证明思路:



对星号运算的封闭性

- 定理2.24: 正则语言对星号封闭.
- 证明思路:设 $A_1=L(N_1)$, $A_1^*=L(N)$, NFA $N_1=(Q_1,\Sigma,\delta_1,q_1,F_1)$. 构造 NFA $N=(Q,\Sigma,\delta,q_0,F)$.

$Q=Q_1\cup\{q_0\}$; $F=F_1\cup\{q_0\}$;

$$\delta(q,a)=\begin{cases}\delta_1(q,a), & \text{若} q\in Q_1 \text{且} q\notin F_1\\ \delta_1(q,a), & \text{若} q\in F_1 \text{且} a\neq\varepsilon\\ \delta_1(q,a)\cup\{q_1\}, & \text{若} q\in F_1 \text{且} a=\varepsilon\\ \{q_1\}, & \text{若} q=q_0 \text{且} a=\varepsilon\\ \varnothing, & \text{若} q=q_0 \text{且} a\neq\varepsilon\end{cases}$$

正则表达式

- 描述模式的手段
- 例: $(0 \cup 1)0^*$

  $= (\{0\} \cup \{1\})\{0\}^*$

  $= \{0,1\}\{0\}^*$

例2.25

- Σ={0,1}
  - $(0\cup 1)^* = \{0,1\}^* = \Sigma^*$
- Σ是任意字母表
  - Σ表示所有长为1的串组成的语言
  - Σ^* 表示所有串组成的语言
  - Σ^*1 表示所有以1结尾的串组成的语言
  - $(0\Sigma^*)\cup(\Sigma^*1)$表示所有以0开头或以1结尾的串组成的语言

正则表达式的形式定义

- 定义2.26: R是正则表达式
当且仅当R是
  - a, a∈Σ;　　/* 表示 {a} */
  - ε;　　　　/* 表示 {ε} */
  - ∅;　　　　/* 表示 ∅ */
  - $(R_1 \cup R_2)$, R_1和R_2都是正则表达式;
  - (R_1R_2), R_1和R_2都是正则表达式;
  - $(R_1{}^*)$, R_1是正则表达式.

- L(R): R表示的语言

说明

- 归纳定义
  - 用较小的表达式定义较大的表达式
- 省略括号
  - 最外层
  - 规定优先级
- 优先级
  - 星号 > 连接 > 并

例2.27

- 设Σ={0,1}
  - $0^*10^* = \{ w \mid w$恰好有一个1 $\}$
  - $\Sigma^*1\Sigma^* = \{ w \mid w$至少有一个1 $\}$
  - $\Sigma^*001\Sigma^* = \{ w \mid w$含有子串001 $\}$
  - $(\Sigma\Sigma)^* = \{ w \mid w$为偶数长度 $\}$
  - $(\Sigma\Sigma\Sigma)^* = \{ w \mid w$长度为3的倍数 $\}$

例2.27(续)

- $01 \cup 10 = \{ 01, 10 \}$
- $0\Sigma^*0 \cup 1\Sigma^*1 \cup 0 \cup 1 =$
  $\{ w \mid w$首尾符号相同$\}$
- $(0 \cup \varepsilon)1^* = 01^* \cup 1^*$
- $(0 \cup \varepsilon)(1 \cup \varepsilon) = \{\varepsilon, 0, 1, 01\}$
- $1^*\varnothing = \varnothing$
- $\varnothing^* = \{\varepsilon\}$

几个恒等式

- $R \cup \varnothing = R$
- $R\varepsilon = R$
  - 一般 $R\varnothing \neq R$, $R \cup \varepsilon \neq R$
- $R \cup \varepsilon = R \cup \{\varepsilon\}$
- $R\varnothing = \varnothing$

例

- 数值常量
- $\{+,-,\varepsilon\}(DD^*\cup DD^*.D^*\cup D^*.DD^*)$
- $D=\{0,1,2,3,4,5,6,7,8,9\}$
  - 72
  - 3.14159
  - +7.
  - -.01

作业

- 1.7 1.10 1.11