2.8 Minimização de um Autômato Finito

♦ Objetivo

- gerar um AF equivalente
- com o menor número de estados possível

♦ Minimização do número de estados

- adotada na maioria das soluções práticas
- entretanto, em algumas aplicações
  - minimizar do número de estados pode não implicar no menor custo de implementação
- exemplo
  - desenho de circuitos eletrônicos
  - pode ser desejável introduzir estados intermediários

    * para melhorar a eficiência
    * ou simplesmente facilitar as ligações físicas

- nestes casos o algoritmo deve ser modificado
    * prevendo as variáveis específicas da aplicação

♦ O autômato mínimo é único

- a minimização de AF distintos
    * que aceitam a mesma linguagem
    * geram o mesmo AF mínimo

♦ Idéia básica do algoritmo

- unificar os *estados equivalentes*

♦ *Definição. Estados Equivalentes*

- q e p são equivalentes sse

* para qq w,
* $\delta(q, w)$ e $\delta(p, w)$
* resultam simultaneamente em estados finais, ou não-finais

- ou seja
  * processamento de uma entrada qq
  * a partir de estados equivalentes
  * gera o mesmo resultado aceita/rejeita

♦ *Pré-Requisitos do Algoritmo*

- AF deve ser determinístico
- não pode ter estados inacessíveis
  - ∗ não-atingíveis a partir do estado inicial
- a função programa deve ser total

♦ Caso o AF não satisfaça algum dos pré-requisitos

- gerar um AFD equivalente
  - ∗ algoritmos introduzidos nos teoremas
- eliminar os estados inacessíveis e suas correspondentes transições
  - ∗ algoritmo relativamente simples
  - ∗ sugerido como exercício
- função programa total

* introduzir um novo estado não-final d
* incluir as transições não-previstas, tendo como resultado o estado d
* incluir um ciclo em d para todos os símbolos do alfabeto

♦ Idéia básica do algoritmo

- identifica os estados equivalentes
  * por *exclusão*
- a partir de uma tabela de estados
  * são marcados os estados não-equivalentes
- ao final do algoritmo
  * as referências não-marcadas
  * representam os estados equivalentes.

♦ *Algoritmo de Minimização*

- seja $M = (\Sigma, Q, \delta, q_0, F)$ um AFD
  - * satisfaz aos pré-requisitos
- tabela: relaciona os estados distintos

q_1					
q_2					
...					
q_n					
d					
	q_0	q_1	...	q_{n-1}	q_n

- marcar na tabela os pares
  - * {estado final, estado não-final}
  - * estados finais não são equivalentes a não-finais

♦ Para $\{q_u, q_v\}$ não-marcado e $a \in \Sigma$

- suponha $\delta(q_u, a) = p_u$ e $\delta(q_v, a) = p_v$
- se $p_u = p_v$
  - q_u é equivalente a q_v para o símbolo a
  - *não marcar*
- se $p_u \neq p_v$ e o par $\{p_u, p_v\}$ é não-marcado
  - $\{q_u, q_v\}$ é incluído em uma lista a partir de $\{p_u, p_v\}$
  - para posterior análise
- se $p_u \neq p_v$ e o par $\{p_u, p_v\}$ é marcado
  - $\{q_u, q_v\}$ é não-equivalente
  - *marcar*
  - se $\{q_u, q_v\}$ encabeça uma lista marcar todos os pares da lista e, recursivamente, se algum par da lista encabeça outra lista

♦ Pares não-marcados são equivalentes: ***unificar***

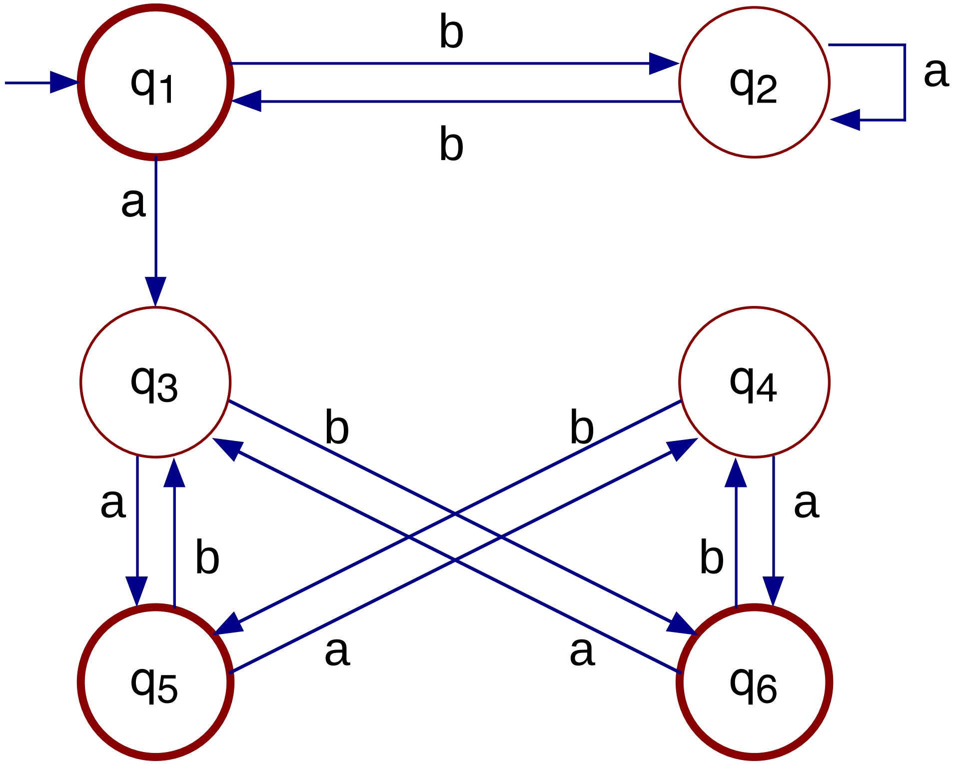
- a equivalência de estados é transitiva
- pares de estados equivalentes são unificados como um único estado
- se algum dos estados equivalentes é inicial
  * estado unificado é inicial

♦ Estados inúteis devem ser excluídos

- um estado q é inútil
  * se é não-final
  * e a partir de q não é possível atingir um estado final
- o estado d
  * se incluído
  * sempre é inútil

♦ ***Exemplo*: Considere o AFD**

- satisfaz os pré-requisitos de minimização



- contrução da tabela
- marcação dos pares
  - {estado final, estado não-final}

q_2	×				
q_3	×				
q_4	×				
q_5		×	×	×	
q_6		×	×	×	
	q_1	q_2	q_3	q_4	q_5

- análise dos pares de estado não-marcados

- **$\{q_1, q_5\}$**

$$\delta(q_1, a) = q_3 \quad \delta(q_5, a) = q_4$$
$$\delta(q_1, b) = q_2 \quad \delta(q_5, b) = q_3$$

$\{q_2, q_3\}$ e $\{q_3, q_4\}$ são não-marcados:

$\{q_1, q_5\}$ é *incluído* nas listas de $\{q_2, q_3\}$ e $\{q_3, q_4\}$

- **$\{q_1, q_6\}$**

$$\delta(q_1, a) = q_3 \quad \delta(q_6, a) = q_3$$
$$\delta(q_1, b) = q_2 \quad \delta(q_6, b) = q_4$$

$\{q_2, q_4\}$ é não-marcado (e como $\{q_3, q_3\}$ é trivialmente equivalente):

$\{q_1, q_6\}$ é *incluído* na lista de $\{q_2, q_4\}$

- **$\{q_2, q_3\}$**

$$\delta(q_2, a) = q_2 \quad \delta(q_3, a) = q_5$$
$$\delta(q_2, b) = q_1 \quad \delta(q_3, b) = q_6$$

$\{q_2, q_5\}$ é marcado: $\{q_2, q_3\}$ é *marcado*

$\{q_2, q_3\}$ encabeça uma lista: $\{q_1, q_5\}$ é *marcado*

- **$\{q_2, q_4\}$**

$$\delta(q_2, a) = q_2 \quad \delta(q_4, a) = q_6$$
$$\delta(q_2, b) = q_1 \quad \delta(q_4, b) = q_5$$

$\{q_2, q_6\}$ e $\{q_1, q_5\}$ são marcados: $\{q_2, q_4\}$ é *marcado*

$\{q_2, q_4\}$ encabeça uma lista: $\{q_1, q_6\}$ é *marcado*

- **$\{q_3, q_4\}$**

$$\delta(q_3, a) = q_5 \quad \delta(q_4, a) = q_6$$
$$\delta(q_3, b) = q_6 \quad \delta(q_4, b) = q_5$$

$\{q_5, q_6\}$ é não-marcado:

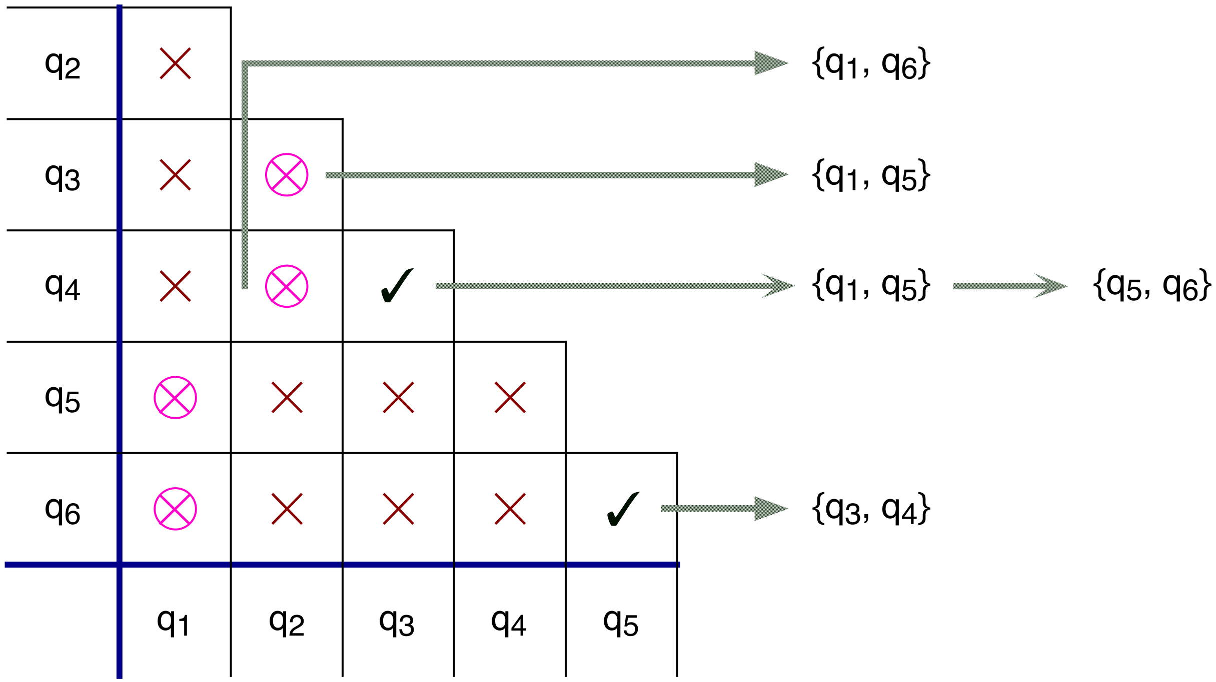
$\{q_3, q_4\}$ é *incluído* na lista de $\{q_5, q_6\}$

- **$\{q_5, q_6\}$**

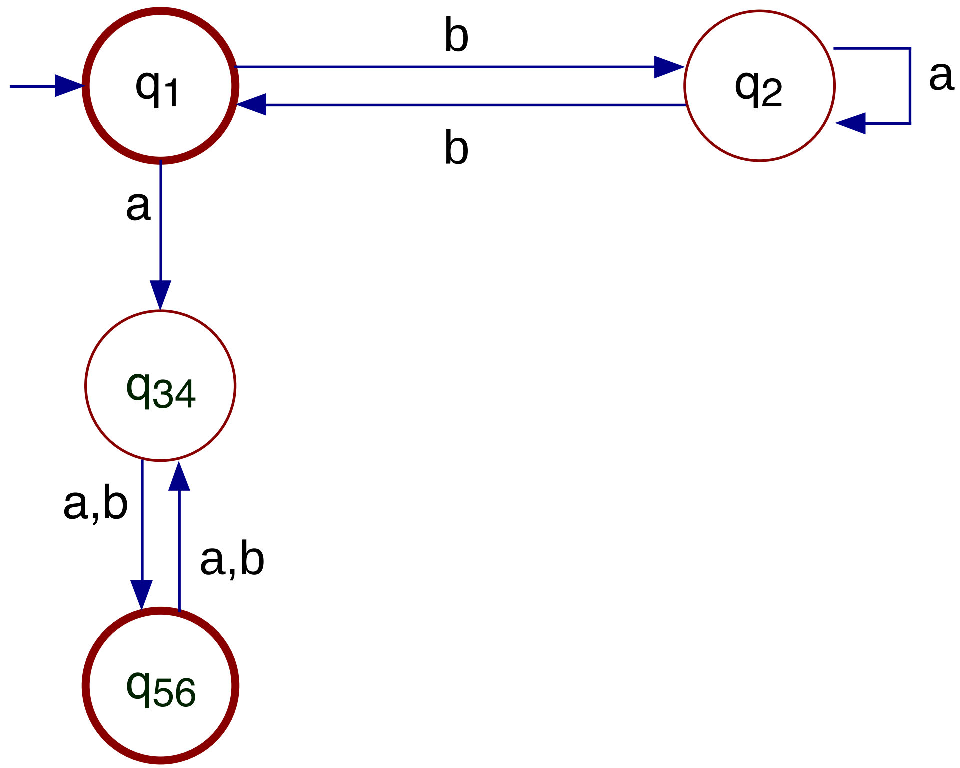
$$\delta(q_5, a) = q_4 \quad \delta(q_6, a) = q_3$$
$$\delta(q_5, b) = q_3 \quad \delta(q_6, b) = q_4$$

Como $\{q_3, q_4\}$ é não-marcado:

$\{q_5, q_6\}$ é *incluído* na lista de $\{q_3, q_4\}$



- Como os pares $\{q_3, q_4\}$ e $\{q_5, q_6\}$ são não-marcados
  - q_{34}: unificação dos estados não-finais q_3 e q_4;
  - q_{56}: unificação dos estados finais q_5 e q_6.


q1
b
q2
a
b
a
q34
a,b
a,b
q56

- ***Teorema*:**
  - O AFD construído
  - usando o algoritmo de minimização
  - é o autômato com menor número de estados para a linguagem

- ***Teorema*:**
  - O *AFD mínimo* de uma linguagem é *único*
  - a menos de ismomorfismo