

Mínimização de Autômatos Finitos

Dado um autômato finito determinístico qualquer, o objetivo da minimização é querer o autômato finito equivalente com o menor número de estados possível, denominado de Autômato Finito Determinístico Mínimo ou simplesmente Autômato Finito Mínimo.

Exemplo: Circuitos Lógicos ou Redes Lógicas.
 [Pode ser necessário introduzir estados intermediários]

O Autômato Finito Mínimo é único (a menos de isomorfismo). Assim, dois autômatos distintos que aceitam a mesma linguagem serão minimizados, geram o mesmo autômato finito mínimo, diferindo, eventualmente, na identificação dos estados.

Basicamente, o algoritmo de minimização unifica os estados equivalentes, definidos a seguir.

Estados Equivalentes

Definição: Seja $M = (\Sigma, Q, \delta, q_0, F)$ um autômato finito determinístico qualquer. Dois estados p e q de Q são ditos Estados Equivalentes se, e somente se, para qualquer palavra w pertencente a Σ^* , simultaneamente $\delta(q, w) \in \text{ACEITAM}(M)$ e $\delta(p, w) \in \text{ACEITAM}(M)$, ou simultaneamente $\delta(q, w) \in \text{REJETAM}(M)$ e $\delta(p, w) \in \text{REJETAM}(M)$.

Portanto, o processamento de uma entrada a partir de estados equivalentes resulta na mesma condição de ACEITA/REJEITA.

Autômato Finito Mínimo

Para uma dada linguagem regular L , o correspondente autômato determinístico mínimo, ou simplesmente autômato finito mínimo. É um AFD:

$$M_m = (\Sigma, Q_m, \delta_m, q_{0m}, F_m)$$

tal que $\text{ACEITAM}(M_m) = L$ e que, para qualquer outro autômato finito determinístico

$M = (\Sigma, Q, \delta, q_0, F)$ tal que $\text{ACEITAM}(M) = L$, ocorre:

$$\#Q \geq \#Q_m \quad \text{Cardinalidade } \# \text{ } \#$$

Pré-requisitos do Algoritmo de Minimização.

Um autômato finito a ser minimizado deve satisfazer os seguintes pré-requisitos:

- a. Deve ser determinístico;
- b. Todos os estados do autômato devem ser estados alcançáveis a partir do estado inicial. Ou seja, não pode ter estados inacessíveis;
- c. A função programa (função de transição) deve ser total (a partir de qualquer estado, são previstas transições para todos os símbolos do alfabeto).

Caso o automato não satisfaça algum dos pré-requisitos, é necessário gerar um automato equivalente, como segue, respectivamente:
3) a. Gerar um automato determinístico equivalente, usando algoritmos de traduções apresentados.
b. Eliminar os estados inacessíveis e suas transições correspondentes.
c. Para transformar a função programa em total, é suficiente introduzir um novo estado não final, e incluir as transições não previstas, tendo-o como estado destino! Por fim, incluir um ciclo em d para todos os símbolos do alfabeto. (uma construção análoga, mas com objetivo diferente foi realizada na prova do teorema 4.3. - Operações fechadas sobre linguagens regulares).

O algoritmo apresentado a seguir 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

Suponha um autômato finito determinístico $M(\Sigma, Q, \delta, q_0, F)$ que satisfaça os pré-requisitos do algoritmo de minimização. Os passos do algoritmo de minimização são os seguintes:

Passo 1: Construção da Tabela. Construir uma tabela, relacionando os estados distintos, sendo que cada par (não ordenado) de estados ocorre somente uma vez, como ilustrado na figura 4.7.

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

Figura 4.7. Minimização: tabela de estados.

PASSO 2: Marcação dos estados trivialmente não equivalentes. Marcar todos os pares do tipo {estado final, estado não final}, pois, obviamente estados finais não são equivalentes a estados não finais.

Passo 3: Marcacão dos estados não equivalentes. (5)

Para cada par $\{q_u, q_v\}$ não marcados e para cada símbolo $a \in \Sigma$, suponha que:

$$\delta(q_u, a) = p_u \quad e \quad \delta(q_v, a) = p_v$$

Axim,

- ⓐ Se $p_u = p_v$, então $\{q_u, q_v\}$ é equivalente a p_u para o símbolo a e não deve ser marcado.
- ⓑ Se $p_u \neq p_v$ e o par $\{p_u, p_v\}$ não está marcado, então $\{q_u, q_v\}$ é incluído em uma lista a partir de $\{p_u, p_v\}$ para posterior análise;
- ⓒ Se $p_u \neq p_v$ e o par $\{p_u, p_v\}$ está marcado,
então:
 - * $\{q_u, q_v\}$ não é equivalente e deve ser marcado;
 - * se $\{q_u, q_v\}$ encabeça uma lista de pares, então marcar todos os pares da lista (e, recursivamente, se algum par da lista encabeça outra lista).

Passo 4: Unificações dos Estados Equivalentes.

Os estados dos pares não marcados são equivalentes e podem ser unificados como segue:

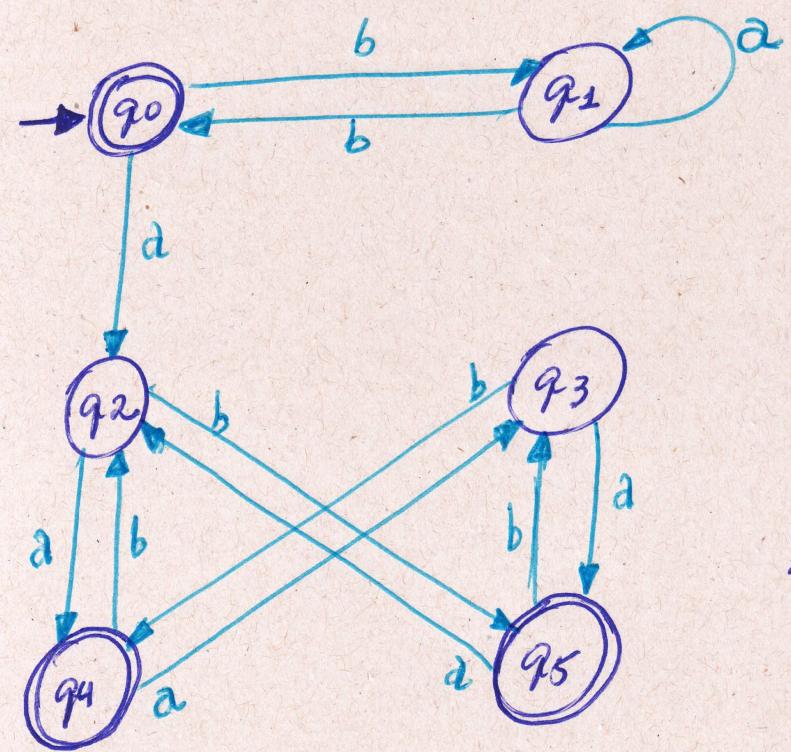
- * A equivalência de Estados é transitiva;
- * pares de estados não finais equivalentes podem ser unificados com um único estado não final.
- * pares de estados finais equivalentes podem ser unificados como um único estado final;
- * se algum dos estados equivalentes é inicial, então o correspondente estado unificado é inicial;
- * todas as transições com origem (respectivamente, com destino) em um estado equivalente, são preservadas, mas passam a ter origem (respectivamente, destino) no correspondente estado unificado;

Passo 5: Exclusão de Estados Inúteis. Por fim, os estados chamados inúteis devem ser excluídos. Um estado que é dito estado inútil, se é não final e a partir de que não é possível atingir um estado final. Deve-se observar que o estado d (se incluído) sempre é inútil (o algoritmo para excluir os estados inúteis é simples e é sugerido como exercício): Todas as transições com origem ou destino em um estado inútil são excluídas.

7

Exemplo : Minimização

Considero o AFD iustrado: Autômato Finito a ser minimizado.



O autômato satisfaz os pré-requisitos de minimização (e, consequentemente, não é necessária incluir o estado d).

Os passos do algoritmo são como segue:

PASSO 1 : Construção da Tabela :

q_2	X				
q_2	X				
q_3	X				
q_4		X	X	X	
q_5		X	X	X	
	q_0	q_1	q_2	q_3	q_4

PASSO 2 : Marcação dos paus do tipo {estado final, estado não final}.

Passo 3: Análise dos pares de estado não marcados, (8)
sendo que a tabela resultante é ilustrada
na figura 4.10, e sendo o símbolo \times usado
para ~~marcar~~ representar os pares marcados
nesta etapa:

a) Análise do par $\{q_0, q_4\}$:

$$\begin{array}{ll} \delta(q_0, a) = q_2 & \delta(q_0, b) = q_1 \\ \delta(q_4, a) = q_3 & \delta(q_4, b) = q_2 \end{array}$$

Como $\{q_1, q_2\}$ e $\{q_2, q_3\}$ são não marcados, então
 $\{q_0, q_4\}$ é incluído nas listas encabeçadas por $\{q_1, q_2\}$ e
 $\{q_2, q_3\}$.

b) Análise do par $\{q_0, q_5\}$:

$$\begin{array}{ll} \delta(q_0, a) = q_2 & \delta(q_0, b) = q_1 \\ \delta(q_5, a) = q_2 & \delta(q_5, b) = q_3 \end{array}$$

$\{q_2, q_2\}$ é trivialmente equivalente.

Como $\{q_1, q_3\}$ é não marcado, então $\{q_0, q_5\}$ é incluído
na lista encabeçada por $\{q_1, q_3\}$;

(9)

④ Análise do par $\{q_1, q_2\}$:

$$\begin{array}{ll} \delta(q_1, a) = q_1 & \delta(q_1, b) = q_0 \\ \delta(q_2, a) = q_4 & \delta(q_2, b) = q_5 \end{array}$$

Como $\{q_1, q_4\}$ é marcado, então $\{q_1, q_2\}$ também é marcado.

Como $\{q_0, q_5\}$

Como $\{q_1, q_2\}$ encabeça uma lista, o par $\{q_0, q_4\}$ também é marcado;

⑤ Análise do par $\{q_1, q_3\}$:

$$\begin{array}{ll} \delta(q_1, a) = q_1 & \delta(q_1, b) = q_0 \\ \delta(q_3, a) = q_5 & \delta(q_3, b) = q_4 \end{array}$$

Como $\{q_1, q_5\}$ bem como $\{q_0, q_4\}$ não são marcados, então $\{q_1, q_3\}$ também é marcado.

Como $\{q_1, q_3\}$ encabeça uma lista, o par $\{q_0, q_5\}$ também é marcado;

⑥ Análise do par $\{q_2, q_3\}$:

$$\begin{array}{ll} \delta(q_2, a) = q_4 & \delta(q_2, b) = q_5 \\ \delta(q_3, a) = q_5 & \delta(q_3, b) = q_4 \end{array}$$

Como $\{q_4, q_5\}$ é não marcado, então $\{q_2, q_3\}$ é incluído na lista encabeçada por $\{q_4, q_5\}$;

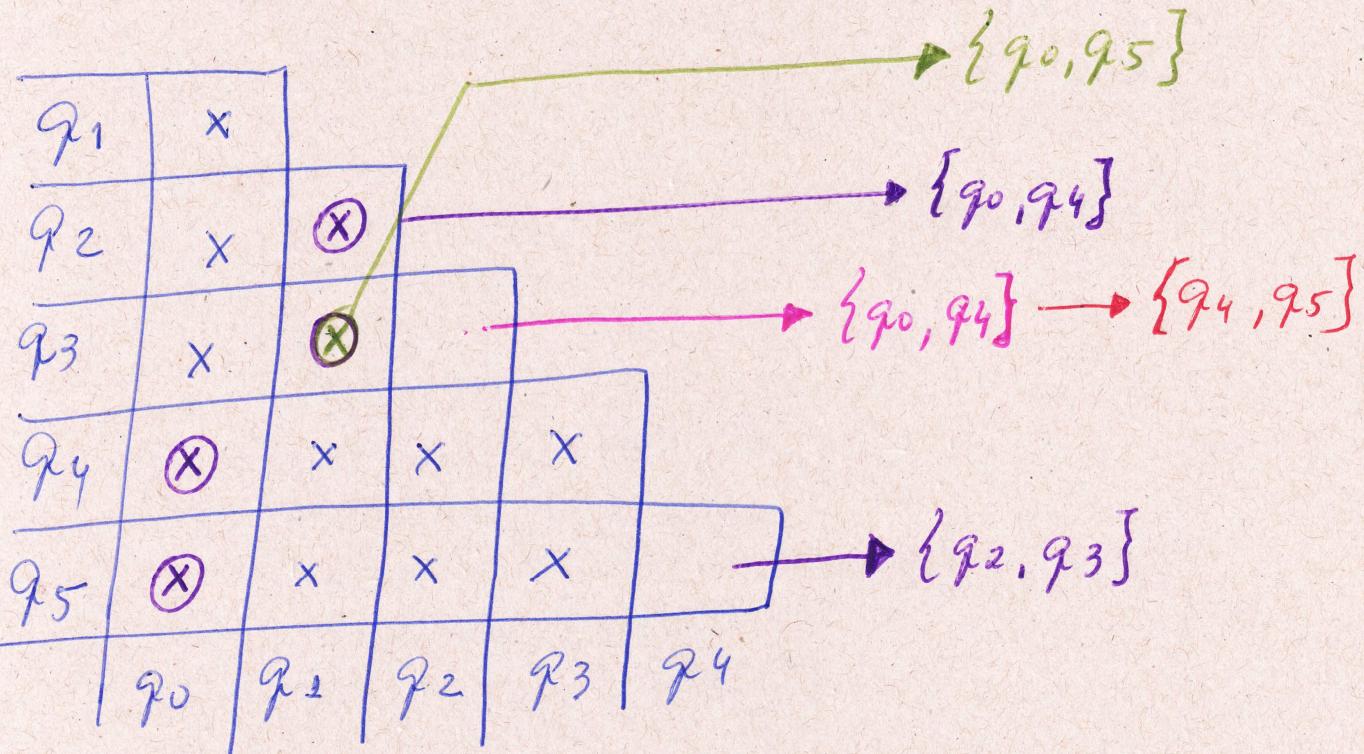
Análise do par $\{q_4, q_5\}$:

$$\begin{array}{ll} \delta(q_4, a) = q_3 & \delta(q_4, b) = q_2 \\ \delta(q_5, a) = q_2 & \delta(q_5, b) = q_3 \end{array}$$

Como $\{q_2, q_3\}$ é não marcado, então $\{q_4, q_5\}$ é incluído na lista encadeada por $\{q_2, q_3\}$.

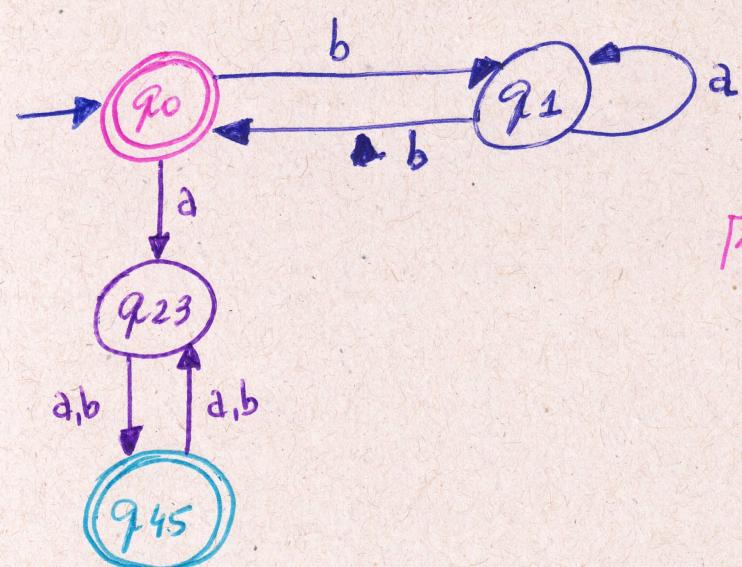
Passo 4: Como os pares $\{q_2, q_3\}$ e $\{q_4, q_5\}$ são não marcados, as seguintes unificações podem ser feitas:

- * q_{23} representa a unificação dos estados finais q_2 e q_3 ;
- * q_{45} representa a unificação dos estados finais q_4 e q_5 ;



(11)

O autômato mínimo resultante possui quatro estados, observe a preservação das transições cujos estados origem e destino foram unificados.



Autônomo Finito Mínimo

Teorema: ~~Unicidade~~ Unicidade do Autônomo Finito Mínimo.

O autônomo finito determinístico mínimo de uma linguagem é único, a menos de isomorfismo.

Assim, dois autômatos distintos que aceitam a mesma linguagem, ao serem minimizados, geram o mesmo autônomo finito mínimo, diferenciando-se, eventualmente, na identificação dos estados.

A definição formal de isomorfismo de autômatos finitos não será apresentada.

Como um AF mínimo é único, a menos de isomorfismo, é usual ser referido como (e não como um) autômato finito mínimo.

Exercícios: Minimize os automatos finitos:
Dica: Não esqueça de verificar os pré-requisitos do algoritmo de minimização.

