$\delta_9'(q_0, a) = \underline{\delta_9}(\{ q_0 \}, a) = F\varepsilon(\{ r \mid r \in \delta(s, a) \text{ e } s \in \underline{\delta}(\{ q_0 \}, \varepsilon) \}) = \{ q_0, q_1, q_2 \}$
$\delta_9'(q_0, b) = \underline{\delta_9}(\{ q_0 \}, b) = F\varepsilon(\{ r \mid r \in \delta(s, b) \text{ e } s \in \underline{\delta}(\{ q_0 \}, \varepsilon) \}) = \{ q_1, q_2 \}$
$\delta_9'(q_1, a) = \underline{\delta_9}(\{ q_1 \}, a) = F\varepsilon(\{ r \mid r \in \delta(s, a) \text{ e } s \in \underline{\delta}(\{ q_1 \}, \varepsilon) \}) = \{ q_2 \}$
$\delta_9'(q_1, b) = \underline{\delta_9}(\{ q_1 \}, b) = F\varepsilon(\{ r \mid r \in \delta(s, b) \text{ e } s \in \underline{\delta}(\{ q_1 \}, \varepsilon) \}) = \{ q_1, q_2 \}$
$\delta_9'(q_2, a) = \underline{\delta_9}(\{ q_2 \}, a) = F\varepsilon(\{ r \mid r \in \delta(s, a) \text{ e } s \in \underline{\delta}(\{ q_2 \}, \varepsilon) \}) = \{ q_2 \}$
$\delta_9'(q_2, b) = \underline{\delta_9}(\{ q_2 \}, b) = F\varepsilon(\{ r \mid r \in \delta(s, b) \text{ e } s \in \underline{\delta}(\{ q_2 \}, \varepsilon) \})$ é indefinida. ❑

2.5 Expressão Regular

Toda Linguagem Regular pode ser descrita por uma expressão simples, denominada Expressão Regular. Trata-se de um formalismo denotacional, também considerado gerador, pois pode-se inferir como construir ("gerar") as palavras de uma linguagem. Uma Expressão Regular é definida a partir de conjuntos (linguagens) básicos e operações de concatenação e união. As Expressões Regulares são consideradas adequadas para a comunicação homem × homem e, principalmente, para a comunicação homem × máquina.

Definição 2.12 Expressão Regular.

Uma *Expressão Regular* (*ER*) sobre um alfabeto Σ é indutivamente definida como segue:

a) $\varnothing$ é uma ER e denota a linguagem vazia;

b) ε é uma ER e denota a linguagem contendo exclusivamente a palavra vazia, ou seja, $\{\varepsilon\}$

c) Qualquer símbolo x pertencente a Σ é uma ER e denota a linguagem contendo a palavra unitária x, ou seja, $\{x\}$

d) Se r e s são ER e denotam as linguagens R e S, respectivamente, então:
   d.1) $(r+s)$ é ER e denota a linguagem $R \cup S$
   d.2) (rs) é ER e denota a linguagem $RS = \{uv \mid u \in R \text{ e } v \in S\}$
   d.3) (r^*) é ER e denota a linguagem R^* ❑

A omissão de parênteses em uma ER é usual, respeitando as seguintes convenções:

- a concatenação sucessiva tem precedência sobre a concatenação e a união;
- a concatenação tem precedência sobre a união.

Uma linguagem gerada por uma Expressão Regular r é representada por L(r) ou GERA(r).

EXEMPLO 12 Expressão Regular.

Na tabela abaixo, são apresentadas Expressões Regulares e as correspondentes linguagens:

Expressão Regular	Linguagem Representada
aa	somente a palavra aa
ba*	todas as palavras que iniciam por b, seguido por zero ou mais a
(a + b)*	todas as palavras sobre {a, b}
(a + b)*aa(a + b)*	todas as palavras contendo aa como subpalavra
a*ba*ba*	todas as palavras contendo exatamente dois b
(a + b)*(aa + bb)	todas as palavras que terminam com aa ou bb
(a + ε)(b + ba)*	todas as palavras que não possuem dois a consecutivos

❑

Os teoremas a seguir mostram que a classe das Expressões Regulares denota exatamente a classe das Linguagens Regulares.

Teorema 2.13 Expressão Regular → Linguagem Regular.

Se r é uma Expressão Regular, então GERA(r) é uma Linguagem Regular.

Prova: Por definição, uma linguagem é Regular se, e somente se, é possível construir um Autômato Finito (Determinístico, Não-Determinístico ou com Movimentos Vazios), que reconheça a linguagem. Assim, é necessário mostrar que, dada uma Expressão Regular r qualquer, é possível construir um Autômato Finito M tal que ACEITA(M) = GERA(r). Na construção do correspondente AFε M apresentada a seguir, a demonstração de que ACEITA(M) = GERA(r) é por indução no número de operadores.

a) *Base de indução.* Seja r uma ER com zero operadores. Então r só pode ser da forma:

$r = \varnothing$

$r = \varepsilon$

$r = x$ (x pertencente a Σ)

Os Autômatos Finitos:

$M_1 = (\varnothing, \{q_0\}, \delta_1, q_0, \varnothing)$

$M_2 = (\varnothing, \{q_f\}, \delta_2, q_f, \{q_f\})$

$M_3 = (\{x\}, \{q_0, q_f\}, \delta_3, q_0, \{q_f\})$

ilustrados na Figura 2.17, aceitam as linguagens acima, respectivamente;

b) *Hipótese de Indução.* Seja r uma ER com até $n > 0$ operadores. Suponha que é possível definir um Autômato Finito que aceita a linguagem gerada por r;

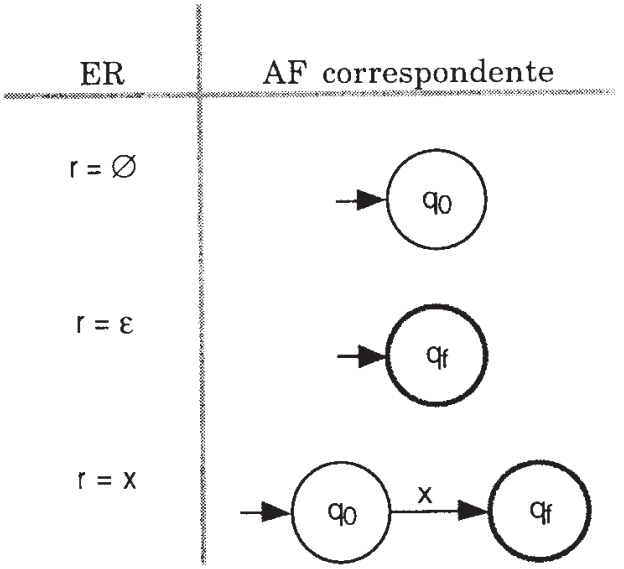


Figura 2.17 Autômatos finitos correspondentes às Expressões Regulares com zero operadores

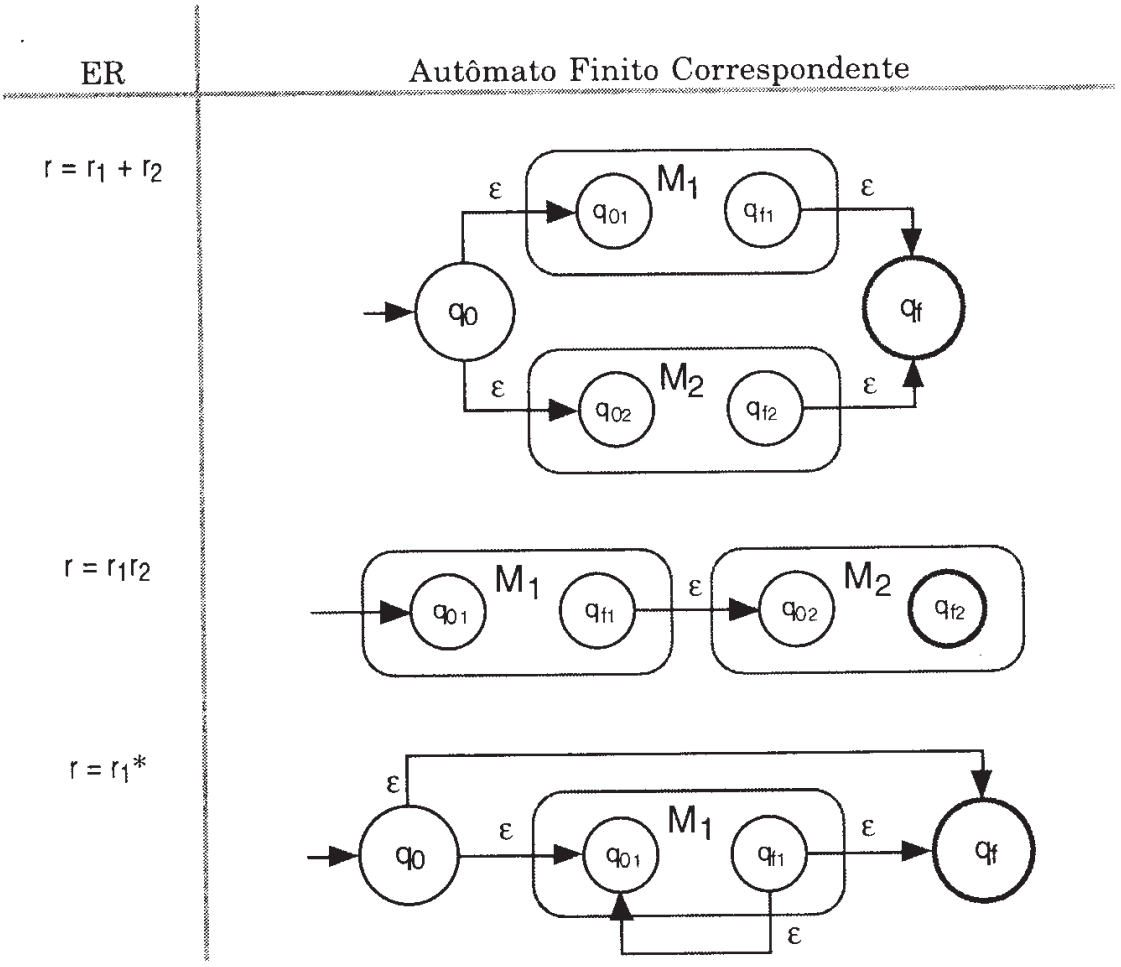


Figura 2.18 Autômatos finitos correspondentes às Expressões Regulares com n+1 operadores

c) *Passo de Indução.* Seja r uma ER com n + 1 operadores. Então r pode ser representada por um dos seguintes casos, onde r_1 e r_2 possuem conjuntamente no máximo n operadores:

$r = r_1 + r_2$

$r = r_1 r_2$

$r = r_1^*$

Portanto, por hipótese de indução é possível construir os autômatos:

$$M_1 = (\Sigma_1, Q_1, \delta_1, q_{0_1}, \{ q_{f_1} \}) \quad \text{e} \quad M_2 = (\Sigma_2, Q_2, \delta_2, q_{0_2}, \{ q_{f_2} \})$$

tais que $ACEITA(M_1) = GERA(r_1)$ e $ACEITA(M_2) = GERA(r_2)$. Note-se que, sem perda de generalidade, é possível assumir que M_1 e M_2 possuem exatamente um estado final (ver exercícios). Adicionalmente, suponha que os conjuntos de alfabetos e de estados dos dois autômatos são disjuntos (se não forem disjuntos, é suficiente realizar uma simples renomeação). Os Autômatos Finitos com Movimentos Vazios que aceitam a linguagem GERA(r) para cada caso são como segue:

c.1) $r = r_1 + r_2$. O autômato:

$$M = (\Sigma_1 \cup \Sigma_2, Q_1 \cup Q_2 \cup \{ q_0, q_f \}, \delta, q_0, \{ q_f \})$$

ilustrado na Figura 2.18 é tal que, a partir do estado inicial q_0, realiza transições vazias para os estados q_{0_1} e q_{0_2}. Assim, M_1 e M_2 processam de forma não-determinística e, portanto, é suficiente um dos módulos aceitar a entrada para o autômato M aceitar;

c.2) $r = r_1 r_2$. O autômato:

$$M = (\Sigma_1 \cup \Sigma_2, Q_1 \cup Q_2, \delta, q_{0_1}, \{ q_{f_2} \})$$

ilustrado na Figura 2.18, ao processar os módulos M_1 e M_2 em seqüência, aceita uma entrada se, e somente se, o prefixo pertencer a $ACEITA(M_1)$ e o sufixo pertencer a $ACEITA(M_2)$;

c.3) $r = r_1^*$. O autômato:

$$M = (\Sigma_1, Q_1 \cup \{ q_0, q_f \}, \delta, q_0, \{ q_f \})$$

ilustrado na Figura 2.18, é tal que a transição vazia de q_0 para q_f garante a aceitação da palavra vazia e o ciclo de q_{f_1} para q_{0_1} permite o sucessivo processamento de M_1 para assegurar o reconhecimento de duas ou mais concatenações sucessivas. ❑

É interessante observar que, na prova do Teorema 2.13, algumas possíveis alterações na construção do autômato resultante podem gerar resultados indesejados, como por exemplo (por quê?):

- no caso $r = r_1 + r_2$, não introduzir os estados q_0 e q_f e identificar ("unificar") os estados iniciais e finais de M_1 com os correspondentes de M_2;

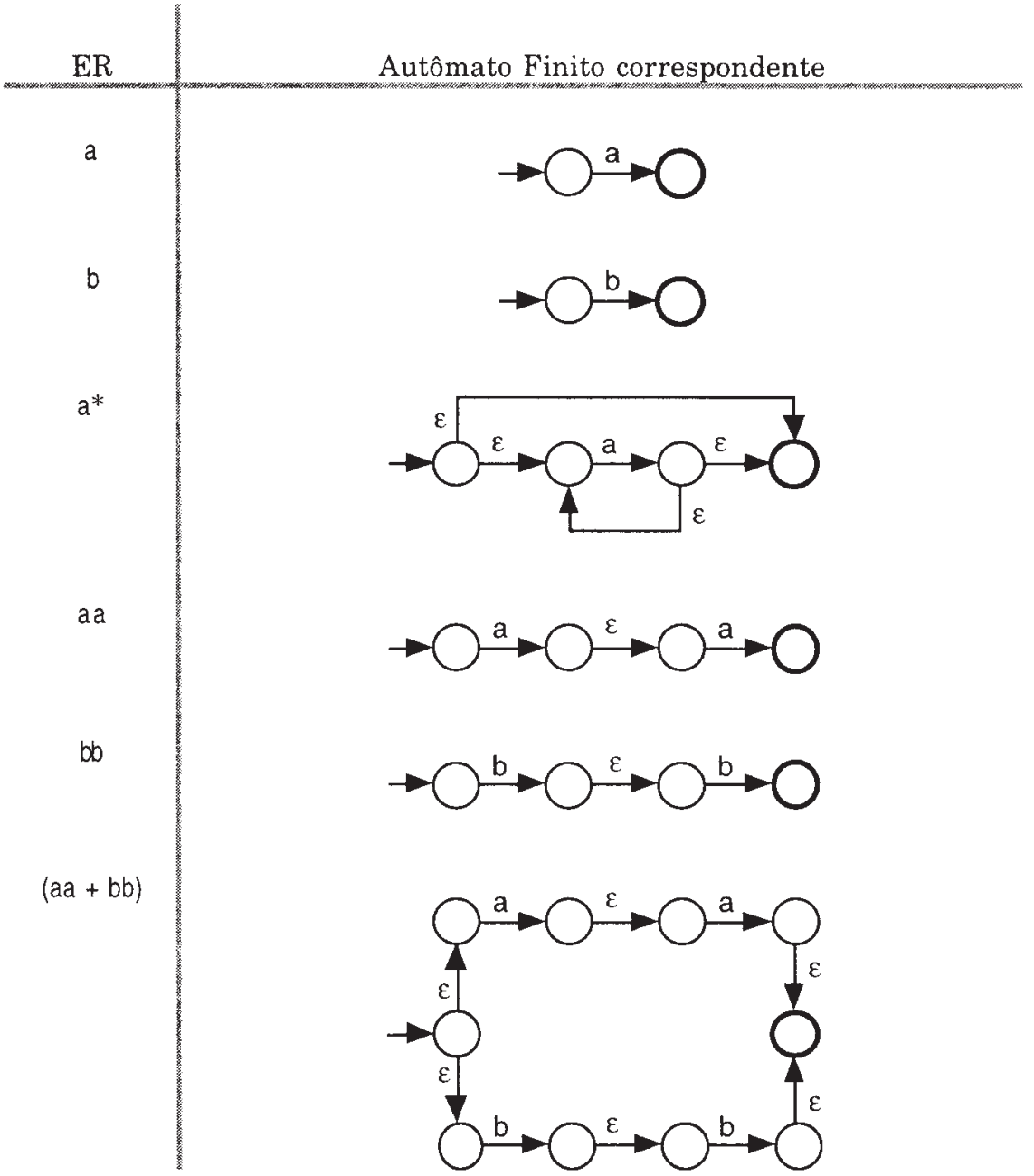


Figura 2.19 Expressões Regulares e os correspondentes Autômatos Finitos

- no caso $r = r_1^*$, não introduzir o estado q_f e manter q_{f_1} como o estado final do autômato resultante. Neste caso, a transição vazia que parte de q_0 teria q_{f_1} como imagem.

EXEMPLO 13 Construção de um AFε a partir de uma Expressão Regular.

A construção dos módulos de um AFε que aceita a linguagem gerada pela ER a*(aa + bb) é como ilustrada na Figura 2.19 e o autômato resultante é ilustrado na Figura 2.20 (por simplicidade, a identificação dos estados é omitida). ❑

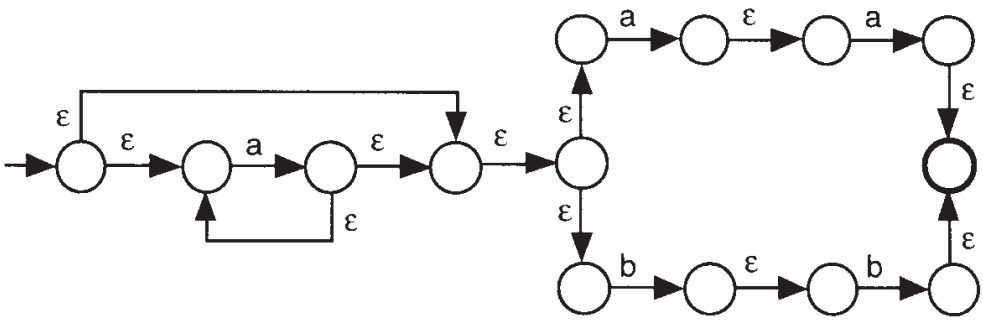


Figura 2.20 Autômato finito resultante

O seguinte teorema não será provado.

Teorema 2.14 Linguagem Regular → Expressão Regular.

Se L é uma Linguagem Regular, então existe uma Expressão Regular r tal que GERA(r) = L. ❑

2.6 Gramática Regular

Usando o conceito geral de gramáticas como introduzido no Capítulo 1 - Introdução e Conceitos Básicos, é possível definir Linguagens Regulares e Não-Regulares (conforme será visto adiante, é fácil definir Linguagens Não-Regulares). Entretanto, é possível estabelecer restrições nas regras de produção, de tal forma a definir exatamente a Classe de Linguagens Regulares.

Definição 2.15 Gramáticas Lineares.

Seja G = (V, T, P, S) uma gramática e sejam A e B elementos de V e w uma palavra de T*. Então G é uma:

a) *Gramática Linear à Direita* (*GLD*). Se todas as regras de produção são da forma:

$$A \rightarrow wB \quad \text{ou} \quad A \rightarrow w$$

b) *Gramática Linear à Esquerda* (*GLE*). Se todas as regras de produção são da forma:

$$A \rightarrow Bw \quad \text{ou} \quad A \rightarrow w$$

c) *Gramática Linear Unitária à Direita* (*GLUD*). Se todas as regras de produção são como na linear à direita e, adicionalmente, $|w| \leq 1$

d) *Gramática Linear Unitária à Esquerda* (*GLUE*). Se todas as regras de produção são como na linear à esquerda e, adicionalmente, $|w| \leq 1$ ❑

Note-se que, nas Gramáticas Lineares, o lado direito de uma produção é constituído por, no máximo, uma variável. Adicionalmente, esta variável, se existir, sempre antecede (linear à esquerda) ou sucede (linear à direita) qualquer subpalavra de terminais.

Teorema 2.16 Equivalência das Gramáticas Lineares.

Seja L uma linguagem. Então:

L é gerada por uma GLD se, e somente se,
L é gerada por uma GLE se, e somente se,
L é gerada por uma GLUD se, e somente se,
L é gerada por uma GLUE. ❑

Ou seja, as diversas formas das Gramáticas Lineares são formalismos equivalentes. A demonstração do teorema é sugerida como exercício.

Definição 2.17 Gramática Regular.

Uma *Gramática Regular* é qualquer Gramática Linear. ❑

Uma linguagem gerada por uma Gramática Regular G é representada por L(G) ou GERA(G).

EXEMPLO 14 Gramática Regular.

A linguagem a(ba)* é gerada pelas seguintes Gramáticas Regulares:

a) *Linear à Direita.* G = ({S, A}, {a, b}, P, S) onde P possui as seguintes regras de produção:
   $S \rightarrow aA$
   $A \rightarrow baA \mid \varepsilon$

b) *Linear à Esquerda.* G = ({S}, {a, b}, P, S) onde P possui as seguintes regras de produção:
   $S \rightarrow Sba \mid a$

c) *Linear Unitária à Direita.* G = ({S, A, B}, {a, b}, P, S) onde P possui as seguintes regras de produção:
   $S \rightarrow aA$
   $A \rightarrow bB \mid \varepsilon$
   $B \rightarrow aA$

d) *Linear Unitária à Esquerda.* G = ({S, A}, {a, b}, P, S) onde P possui as seguintes regras de produção:
   $S \rightarrow Aa \mid a$
   $A \rightarrow Sb$ ❑

EXEMPLO 15 Gramática Regular.

A linguagem (a + b)*(aa + bb) é gerada pelas seguintes Gramáticas Regulares:

a) *Linear à Direita.* $G = (\{S, A\}, \{a, b\}, P, S)$ onde P possui as seguintes regras de produção:

$S \rightarrow aS \mid bS \mid A$

$A \rightarrow aa \mid bb$

b) *Linear à Esquerda.* $G = (\{S, A\}, \{a, b\}, P, S)$ onde P possui as seguintes regras de produção:

$S \rightarrow Aaa \mid Abb$

$A \rightarrow Aa \mid Ab \mid \varepsilon$ ❑

Os teoremas a seguir mostram que a Classe das Gramáticas Regulares denota exatamente a Classe das Linguagens Regulares.

Teorema 2.18 Gramática Regular → Linguagem Regular.

Se L é uma linguagem gerada por uma Gramática Regular, então L é uma Linguagem Regular.

Prova: Para mostrar que uma linguagem é Regular, é suficiente construir um Autômato Finito que a reconheça. Suponha $G = (V, T, P, S)$ uma Gramática Linear Unitária à Direita. Então o AFε $M = (\Sigma, Q, \delta, q_0, F)$ construído abaixo simula as derivações de G, ou seja, ACEITA(M) = GERA(G):

$\Sigma = T$

$Q = V \cup \{q_f\}$

$F = \{q_f\}$

$q_0 = S$

δ é construída como segue (suponha A e B variáveis e a terminal):

Tipo da Produção	Transição Gerada
$A \rightarrow \varepsilon$	$\delta(A, \varepsilon) = q_f$
$A \rightarrow a$	$\delta(A, a) = q_f$
$A \rightarrow B$	$\delta(A, \varepsilon) = B$
$A \rightarrow aB$	$\delta(A, a) = B$

A demonstração que, de fato, ACEITA(M) = GERA(G) é no número de derivações, como segue (suponha α elemento de $(T \cup V)^*$ e w elemento de T^*):

a) *Base de indução.* Suponha que $S \Rightarrow^1 \alpha$. Então, se:

a.1) $\alpha = \varepsilon$, existe uma regra $S \rightarrow \varepsilon$. Por construção de M, $\delta(S, \varepsilon) = q_f$

a.2) $\alpha = a$, existe uma regra $S \rightarrow a$. Por construção de M, $\delta(S, a) = q_f$

a.3) $\alpha = A$, existe uma regra $S \rightarrow A$. Por construção de M, $\delta(S, \varepsilon) = A$

a.4) $\alpha = aA$, existe uma regra $S \rightarrow aA$. Por construção de M, $\delta(S, a) = A$

b) *Hipótese de indução.* Suponha que $S \Rightarrow^n \alpha$, $n > 1$ tal que, se:

b.1) $\alpha = w$, então $\underline{\delta}(S, w) = q_f$

b.2) $\alpha = wA$, então $\underline{\delta}(S, w) = A$

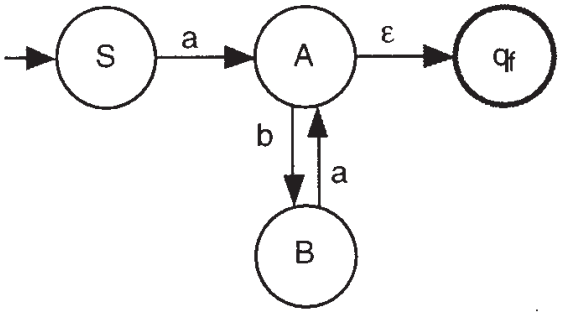


Figura 2.21 Autômato finito construído a partir de uma Gramática Regular

c) *Passo de Indução*. Suponha que $S \Rightarrow^{n+1} \alpha$. Então, obrigatoriamente ocorre somente a hipótese b.2) acima e $S \Rightarrow^{n} wA \Rightarrow^{1} \alpha$. Portanto, se:

c.1) $\alpha = w\varepsilon = w$, existe uma regra $A \rightarrow \varepsilon$. Assim:

$$\underline{\delta}(S, w\varepsilon) = \delta(\underline{\delta}(S, w), \varepsilon) = \delta(A, \varepsilon) = q_f$$

c.2) $\alpha = wb$, existe uma regra $A \rightarrow b$. Assim:

$$\underline{\delta}(S, wb) = \delta(\underline{\delta}(S, w), b) = \delta(A, b) = q_f$$

c.3) $\alpha = wB$, existe uma regra $A \rightarrow B$. Assim:

$$\underline{\delta}(S, w\varepsilon) = \delta(\underline{\delta}(S, w), \varepsilon) = \delta(A, \varepsilon) = B$$

c.4) $\alpha = wbB$, existe uma regra $A \rightarrow bB$. Assim:

$$\underline{\delta}(S, wb) = \delta(\underline{\delta}(S, w), b) = \delta(A, b) = B$$ ❑

EXEMPLO 16 Construção de um AFε a partir de uma Gramática Regular.

Considere a seguinte Gramática Linear Unitária à Direita, introduzida no Exemplo 14:

$$G = (\{S, A, B\}, \{a, b\}, P, S)$$

onde P é tal que:

$S \rightarrow aA$
$A \rightarrow bB \mid \varepsilon$
$B \rightarrow aA$

O AFε M que reconhece a linguagem gerada pela gramática G é:

$$M = (\{a, b\}, \{S, A, B, q_f\}, \delta, S, \{q_f\})$$

o qual é ilustrado na Figura 2.21 e onde δ é construída como segue:

Produção	Transição
$S \rightarrow aA$	$\delta(S, a) = A$
$A \rightarrow bB$	$\delta(A, b) = B$
$A \rightarrow \varepsilon$	$\delta(A, \varepsilon) = q_f$
$B \rightarrow aA$	$\delta(B, a) = A$

❑

Teorema 2.19 Linguagem Regular → Gramática Regular.

Se L é Linguagem Regular, então existe G, Gramática Regular que gera L.

Prova: Se L é uma Linguagem Regular, então existe um AFD $M = (\Sigma, Q, \delta, q_0, F)$ tal que ACEITA(M) = L. A idéia central da demonstração é construir uma Gramática Linear à Direita G a partir de M tal que GERA(G) = ACEITA(M), ou seja, cuja derivação simula a função programa estendida. Seja $G = (V, T, P, S)$ tal que:

$V = Q \cup \{S\}$

$T = \Sigma$

e P é construído como segue (suponha q_i, q_k elementos de Q, q_f elemento de F e a elemento de Σ):

Transição	Produção
-	$S \rightarrow q_0$
-	$q_f \rightarrow \varepsilon$
$\delta(q_i, a) = q_k$	$q_i \rightarrow aq_k$

A demonstração que, de fato, GERA(G) = ACEITA(M) é no tamanho da palavra, como segue (suponha w elemento de Σ^*):

a) *Base de Indução*. Seja w tal que $|w| = 0$. Por definição, tem-se que $S \rightarrow q_0$ é produção. Se $\varepsilon \in$ ACEITA(M), então q_0 é estado final e, por definição, tem-se que $q_0 \rightarrow \varepsilon$ é produção. Logo:

$$S \Rightarrow q_0 \Rightarrow \varepsilon$$

b) *Hipótese de Indução*. Seja w tal que $|w| = n$ $(n \geq 1)$ e $\underline{\delta}(q_0, w) = q$. Assim, se:

b.1) q não é estado final, então suponha que $S \Rightarrow wq$

b.2) q é estado final, então suponha que $S \Rightarrow wq \Rightarrow w$ (este caso não é importante para o passo de indução);

c) *Passo de Indução*. Seja w tal que $|wa| = n + 1$ e $\underline{\delta}(q_0, wa) = p$. Então:

$$\delta(\underline{\delta}(q_0, w), a) = \delta(q, a) = p$$

Portanto, obrigatoriamente ocorre somente a hipótese b.1) acima e, se:

c.1) p não é estado final, então $S \Rightarrow^n wq \Rightarrow^1 wap$

c.2) p é estado final, então $S \Rightarrow^n wq \Rightarrow^1 wap \Rightarrow^1 wa$ ❑

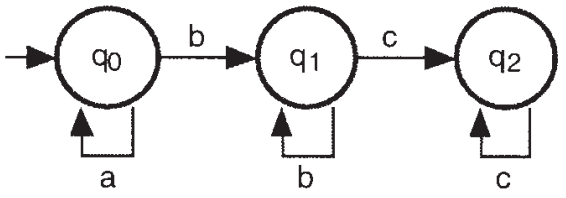


Figura 2.22 Grafo do Autômato Finito Determinístico

EXEMPLO 17 Construção de uma Gramática Regular a partir de um AFD.

Considere o Autômato Finito Determinístico:

$$M = (\{a, b, c\}, \{q_0, q_1, q_2\}, \delta, q_0, \{q_0, q_1, q_2\})$$

ilustrado na Figura 2.22. A correspondente Gramática Regular construída conforme o algoritmo da prova do Teorema 2.19 é:

$$G = (\{q_0, q_1, q_2, S\}, \{a, b, c\}, S, P)$$

onde P é como segue:

Transição	Produção
-	$S \rightarrow q_0$
-	$q_0 \rightarrow \varepsilon$
-	$q_1 \rightarrow \varepsilon$
-	$q_2 \rightarrow \varepsilon$
$\delta(q_0, a) = q_0$	$q_0 \rightarrow aq_0$
$\delta(q_0, b) = q_1$	$q_0 \rightarrow bq_1$
$\delta(q_1, b) = q_1$	$q_1 \rightarrow bq_1$
$\delta(q_1, c) = q_2$	$q_1 \rightarrow cq_2$
$\delta(q_2, c) = q_2$	$q_2 \rightarrow cq_2$

❑

2.7 Propriedades das Linguagens Regulares

Uma das principais características das Linguagens Regulares é o fato de serem representadas por formalismos de pouca complexidade, grande eficiência e fácil implementação. Entretanto, por ser uma classe relativamente simples, é restrita e limitada, sendo fácil definir Linguagens Não-Regulares. Assim, algumas questões sobre Linguagens Regulares necessitam ser analisadas:

a) Como determinar se uma linguagem é Regular?