Capítulo 4

NFAs com Transições Épsilon (ε-NFA)

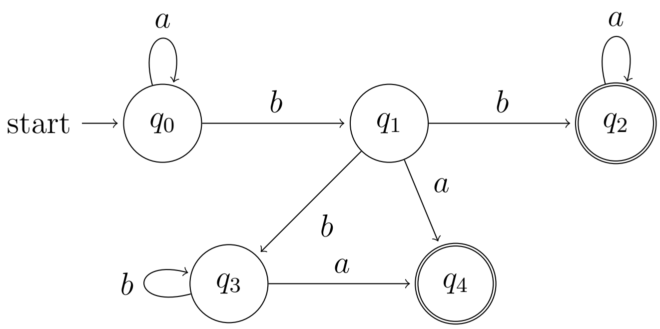
Os autómatos finitos não deterministas com transições épsilon (e-NFAs ou ε-NFAs, do inglês *Non-Deterministic Finite Automata with ε-Moves*) são uma extensão dos NFAs, permitindo transições espontâneas entre estados (isto é, sem consumir nenhum símbolo de entrada). Estas transições introduzem não-determinismo aos autómatos, uma vez que num determinado momento podem estar no estado de origem ou de destino de uma transição ε. Novamente, este não-determinismo não acrescenta novas capacidades a estes autómatos, mas pode permitir simplificar a criação de autómatos em alguns problemas. Vamos também ver como os ε-NFAs podem ser convertidos em DFAs.

Definição

A definição de um ε-NFA continua a ser o tuplo ε-NFA $= (Q, \Sigma, \delta, q_0, F)$, sendo que as transições podem ser efetuadas com um símbolo do alfabeto ou com a cadeia vazia (representada pelo símbolo ε).

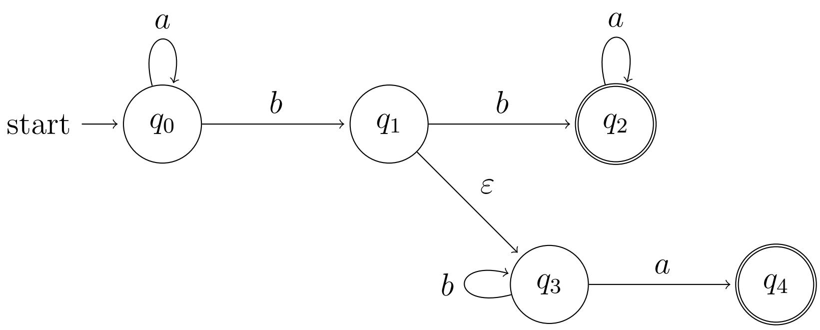
Vamos ver o exemplo de um ε-NFA para a linguagem das cadeias no alfabeto {a, b} constituídas por cadeias no formato a^ibb^ja com $i \geq 0, j \geq 0$, ou no formato a^kbba^m com $k \geq 0, m \geq 0$.

Este problema pode ser resolvido com um NFA sem transições-ε, mas um ε-NFA pode facilitar a criação e interpretação do autómato. Vamos olhar para a solução usando um NFA sem transições-ε:



Com o NFA, temos a parte comum entre os dois formatos de cadeias representada entre q_0 e q_1; no 'ramo de cima' temos a representação das cadeias no formato $a^k bba^m$; já nos ramos de baixo, temos as possibilidades para representação das cadeias no formato $a^i bb^j a$, podendo transitar diretamente de q_1 para q_4 no caso de não existirem b's, ou transitar de q_1 para q_3 caso exista pelo menos um b.

Um ε-NFA permite uma solução com menos transições, e com um aspeto que transmite mais facilmente o formato das cadeias da linguagem:



Neste ε-NFA, e em comparação com o NFA acima, a parte de cima do autómato continua igual, permitindo o reconhecimento das cadeias no formato $a^k bba^m$. Já a parte de baixo torna-se mais simples de interpretar, uma vez que se identifica mais facilmente o formato das cadeias reconhecidas ($a^i bb^j a$). A transição-ε permite então que no momento em que o autómato transita para q_1 com o primeiro b possa imediatamente transitar para q_3 sem consumir qualquer símbolo de entrada. Assim, se o próximo símbolo da cadeia for um a, o autómato transita para q_4; se o símbolo for um b, pode transitar para q_2 ou para q_3, permitindo assim um funcionamento semelhante ao do NFA acima mas permitindo uma leitura mais simplificada.

Vamos ver a tabela de transições para este autómato.

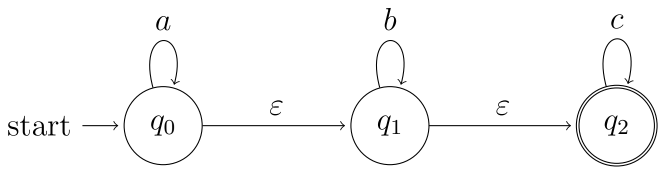
	ε	a	b
$\rightarrow q_0$	$\emptyset$	$\{q_0\}$	$\{q_1\}$
q_1	$\{q_3\}$	$\emptyset$	$\{q_2\}$
$* \; q_2$	$\emptyset$	$\{q_2\}$	$\emptyset$
q_3	$\emptyset$	$\{q_4\}$	$\{q_3\}$
$* \; q_4$	$\emptyset$	$\emptyset$	$\emptyset$

Repare-se que na tabela de transição do ε-NFA é acrescentada uma coluna para as transições-ε, como se se tratasse de mais um símbolo do alfabeto.

Exercício 1

Construa um ε-NFA para a linguagem das cadeias do alfabeto {a,b,c} no formato $a^x b^y c^z$ em que $x \geq 0, y \geq 0, z \geq 0$. Apresente também a tabela de transições.

Uma solução para este problema tem de contemplar a cadeia vazia (que faz também parte da linguagem), e não deve permitir símbolos distintos intercalados. Vamos ver uma solução possível:



Esta solução permite transitar diretamente de q_0 para q_1 e daí para q_2 sem consumir qualquer símbolo de entrada, aceitando assim a cadeia vazia. Também permite um qualquer número de a's, b's e c's nos estados q_0, q_1 e q_2 respetivamente, garantindo assim a ordem e o formato das cadeias da linguagem.

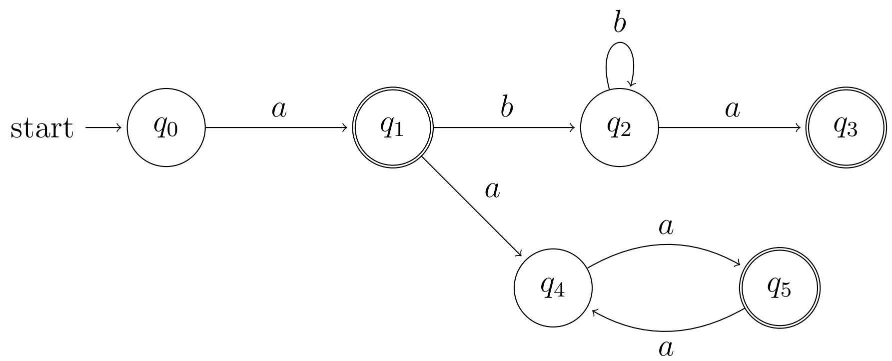
A tabela de transições é agora fácil de obter.

	ε	a	b	c
$\rightarrow q_0$	$\{q_1\}$	$\{q_0\}$	$\emptyset$	$\emptyset$
q_1	$\{q_2\}$	$\emptyset$	$\{q_1\}$	$\emptyset$
$* \, q_2$	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_2\}$

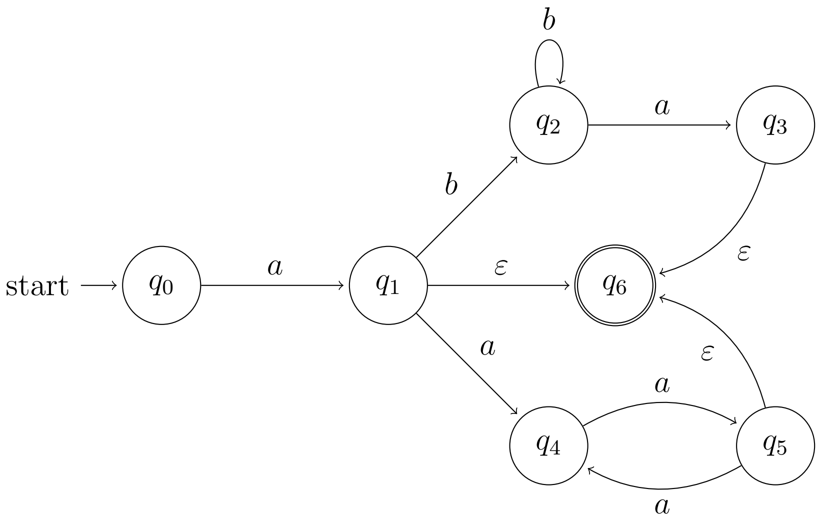
Exercício 2

Construa um ε-NFA para a linguagem das cadeias do alfabeto {a,b} no formato a^{2n+1} com $n \geq 0$ e no formato ab^ia com $i \geq 1$ com apenas um estado de aceitação.

Para que o autómato tenha apenas um estado de aceitação, podemos construir o autómato como se fosse um NFA e acrescentar depois um novo estado (o de aceitação) para o qual acrescentamos transições-ε a partir de todos os estados de aceitação do NFA (os quais deixam de o ser). Assim, podemos olhar para um possível NFA para resolver o problema:



Podemos ver nesta solução que o 'ramo' de cima permite reconhecer as cadeias no formato ab^ia, enquanto que o 'ramo' de baixo permite reconhecer as cadeias com um número ímpar de a's. Note-se que q_1 é também um estado de aceitação para garantir que a cadeia só com 1 a é aceite. Fazendo então a substituição de estados finais, ficamos com o seguinte ε-NFA:

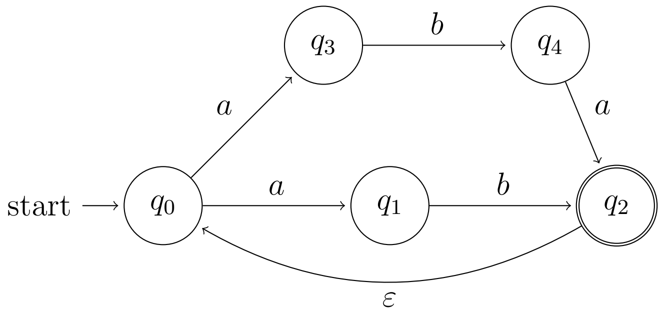


Note-se que a solução permanece igual, exceto que agora apenas existe um estado de aceitação, o qual não tem qualquer transição de saída. Uma outra solução poderia também ter sido atingida aproveitando o estado q_3 como estado final único, e adicionando apenas transições-ε de q_1 e de q_5 para q_3, permitindo assim poupar o estado extra adicionado na solução acima.

Exercício 3

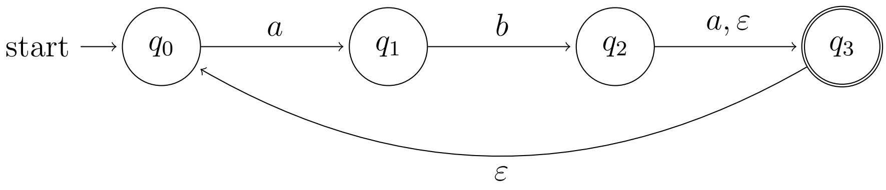
Apresente um ε-NFA que permita reconhecer cadeias do alfabeto {a, b} constituída por uma ou mais repetições de *aba* ou *ab*.

Uma possibilidade de resolução envolve criar um NFA para reconhecer as duas possibilidades, e acrescentar depois uma transição-ε do estado final para o estado inicial, de forma a permitir repetições dessas mesmas cadeias.



Podemos ver as duas possíveis sub-cadeias constituíntes das cadeias da linguagem nos 'ramos' de cima e do meio do autómato. No estado final, a transição-ε permite então começar de novo uma outra sequência *aba* ou *ab*.

Podemos ver abaixo uma outra solução possível, tirando um pouco mais de partido do não-determinismo:

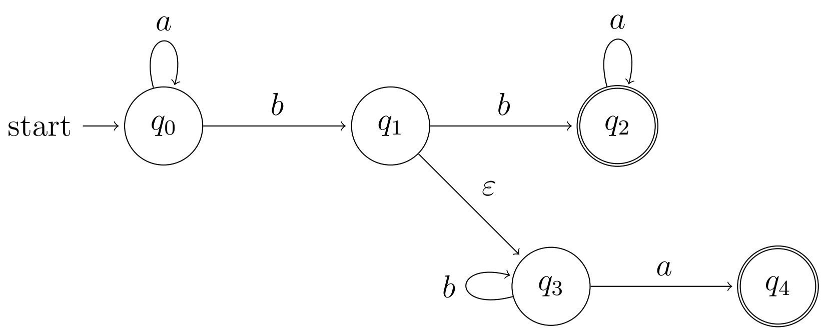


Nesta solução, as partes comuns são colocadas conjuntamente, e é usada uma transição-ε para permitir cadeias *ab* ou *aba*.

Conversão de ε-NFA para DFA

Para converver um ε-NFA para DFA socorremo-nos do fecho-ε dos estados do ε-NFA como auxiliar para o método da construção de sub-conjuntos já visto no capítulo anterior. O fecho-ε (ou *E-Close* ou ε-*Close*) de um estado inclui o próprio estado e todos aqueles que se possam alcançar com transições ε. Incluem-se aqui não só os estados atingíveis diretamente através de uma transição-ε mas também todos aqueles atingíveis a partir desses, e assim sucessivamente.

Voltemos ao exemplo acima, e recordemos o ε-NFA obtido:



Para transformar em DFA, começamos primeiro por identificar o fecho-ε de cada estado:

Podemos agora criar a tabela do DFA pelo método de construção de sub-conjuntos. O estado inicial do DFA resultante (primeira linha na tabela abaixo) será o fecho-ε do estado inicial do ε-NFA original. Para cada

Estado	Fecho-ε
q_0	$\{q_0\}$
q_1	$\{q_1, q_3\}$
q_2	$\{q_2\}$
q_3	$\{q_3\}$
q_4	$\{q_4\}$

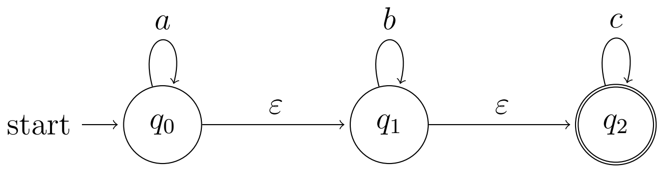
transição (células na tabela), deve ser sempre usado o fecho-ε dos estados de destino. Por exemplo, podemos ver na primeira linha que a partir de q_0 e com um b é possível ir para q_1; devemos na célula colocar não apenas q_1 mas sim o fecho-ε de q_1, ou seja $\{q_1, q_3\}$. Continuamos depois tal como anteriormente, acrescentando linhas na tabela que ainda não estejam presentes.

	a	b
$\rightarrow \{q_0\}$	$\{q_0\}$	$\{q_1, q_3\}$
$\{q_1, q_3\}$	$\{q_4\}$	$\{q_2, q_3\}$
$* \{q_4\}$	$\emptyset$	$\emptyset$
$* \{q_2, q_3\}$	$\{q_2, q_4\}$	$\{q_3\}$
$* \{q_2, q_4\}$	$\{q_2\}$	$\emptyset$
$\{q_3\}$	$\{q_4\}$	$\{q_3\}$
$* \{q_2\}$	$\{q_2\}$	$\emptyset$
$\emptyset$	$\emptyset$	$\emptyset$

Exercício 4

Calcule o fecho-ε de cada estado do ε-NFA obtido no exercício 1 e converta-o para um DFA.

Recordando o ε-NFA do exercício 1:



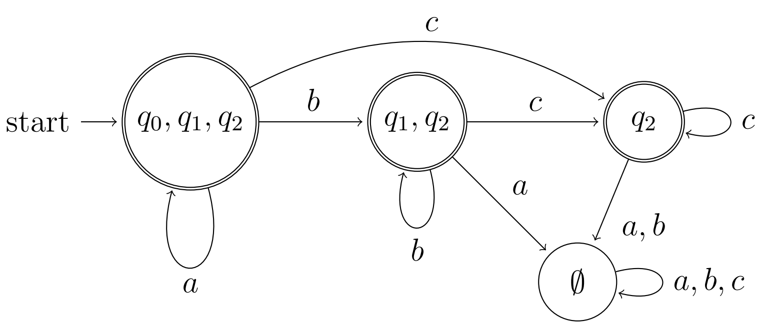
Vamos então calcular o fecho-ε de cada estado. Não esqueçamos que o fecho tem uma definição recursiva, e por isso o fecho-ε de q_0 vai inclui não só q_1 como também q_2.

Estado	Fecho-ε
q_0	$\{q_0, q_1, q_2\}$
q_1	$\{q_1, q_2\}$
q_2	$\{q_2\}$

A partir daqui, podemos construir a tabela para conversão para DFA, começando a preencher a tabela pelo estado inicial, ou seja, usando o fecho-ε do estado inicial do ε-NFA.

	a	b	c
$\rightarrow * \{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_1, q_2\}$	$\{q_2\}$
$* \{q_1, q_2\}$	$\emptyset$	$\{q_1, q_2\}$	$\{q_2\}$
$* \{q_2\}$	$\emptyset$	$\emptyset$	$\{q_2\}$
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

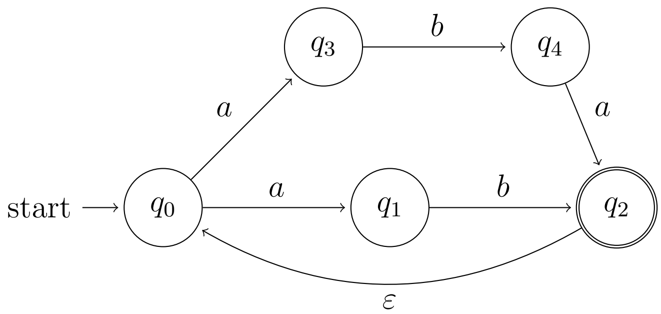
Temos aqui um DFA resultante com o mesmo número de estados do que o ε-NFA que lhe deu origem (mais o estado morto), mas com mais algumas transições, e em que todos os estados são de aceitação. Vamos ver o diagrama equivalente.



Exercício 5

Calcule o fecho-ε de cada estado de cada um dos ε-NFAs obtidos no exercício 3 e converta-os para DFAs.

Recordando o primeiro ε-NFA do exercício 3:



Fazendo o cálculo do fecho-ε dos estados do autómato:

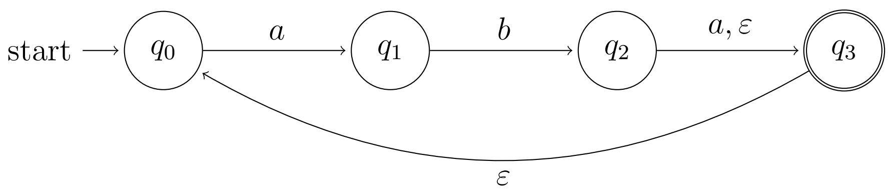
Estado	Fecho-ε
q_0	$\{q_0\}$
q_1	$\{q_1\}$
q_2	$\{q_2,\ q_0\}$
q_3	$\{q_3\}$
q_4	$\{q_4\}$

A partir daqui, usamos então a técnica de construção de sub-conjuntos para obter o DFA equivalente (por questões de simplificação de representação, vamos omitir o estado morto).

Como podemos ver, o DFA resultante tem apenas 4 estados, sendo dois deles estados de aceitação.

	a	b
$\rightarrow \{q_0\}$	$\{q_1,\, q_3\}$	$\emptyset$
$\{q_1,\, q_3\}$	$\emptyset$	$\{q_0,\, q_2,\, q_4\}$
$*\ \{q_0,\, q_2,\, q_4\}$	$\{q_0,\, q_1,\, q_2,\, q_3\}$	$\emptyset$
$*\ \{q_0,\, q_1,\, q_2,\, q_3\}$	$\{q_1,\, q_3\}$	$\{q_0,\, q_2,\, q_4\}$

Olhando agora para a segunda solução do mesmo exercício:



Fazendo o cálculo do fecho-ε dos estados do autómato:

Estado	Fecho-ε
q_0	$\{q_0\}$
q_1	$\{q_1\}$
q_2	$\{q_2,\, q_3,\, q_0\}$
q_3	$\{q_3,\, q_0\}$

A partir daqui, usamos então a técnica de construção de sub-conjuntos para obter o DFA equivalente:

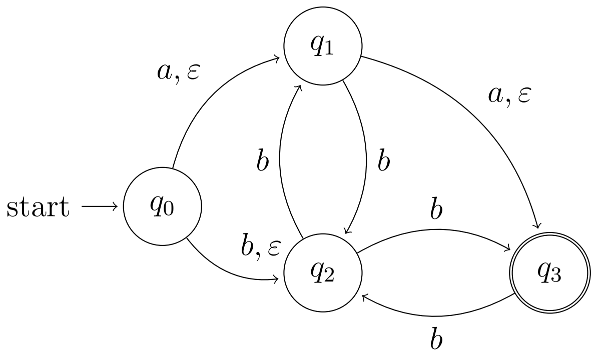
	a	b
$\rightarrow \{q_0\}$	$\{q_1\}$	$\emptyset$
$\{q_1\}$	$\emptyset$	$\{q_2,\, q_3,\, q_0\}$
$*\ \{q_2,\, q_3,\, q_0\}$	$\{q_3,\, q_0,\, q_1\}$	$\emptyset$
$*\ \{q_3,\, q_0,\, q_1\}$	$\{q_1\}$	$\{q_2,\, q_3,\, q_0\}$

O DFA resultante tem também 4 estados, sendo dois deles estados de aceitação. Podemos também notar que o DFA obtido a partir deste segundo

ε-NFA é equivalente estruturalmente ao DFA obtido a partir do primeiro ε-NFA.

Exercício 6

Considere o ε-NFA abaixo. Calcule o fecho-ε de cada estado e converta o autómato para um DFA.



Fazendo o cálculo do fecho-ε dos estados do autómato:

Estado	Fecho-ε
q_0	$\{q_0, q_1, q_2, q_3\}$
q_1	$\{q_1, q_3\}$
q_2	$\{q_2\}$
q_3	$\{q_3\}$

Note-se que o fecho-ε de q_0 inclui não só q_1 e q_2, mas também q_3, dada a definição recursiva de fecho. A partir daqui usamos então a técnica de construção de sub-conjuntos para obter o DFA equivalente.

O DFA resultante tem apenas então cinco estados, sendo quatro deles estados de aceitação.

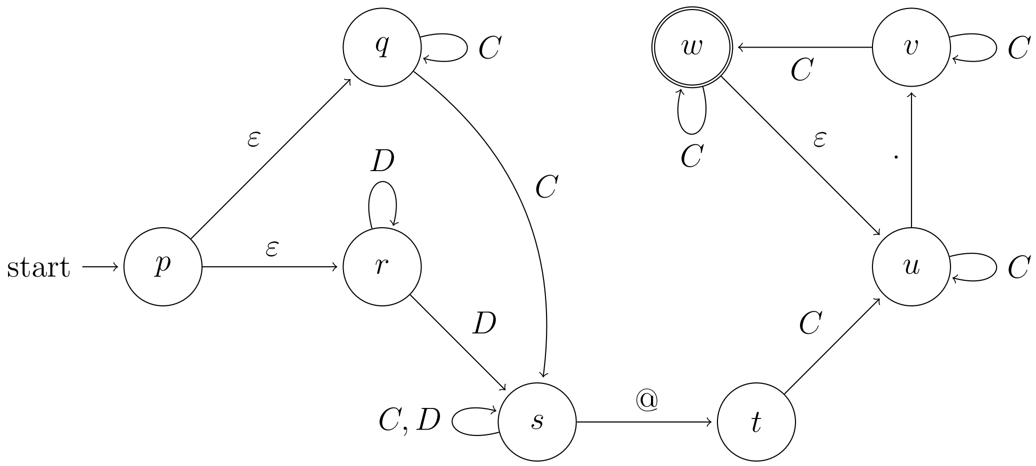
	a	b
$\rightarrow * \{q_0, q_1, q_2, q_3\}$	$\{q_1,\ q_3\}$	$\{q_1, q_2, q_3\}$
$* \{q_1,\ q_3\}$	$\{q_3\}$	$\{q_2\}$
$* \{q_1, q_2, q_3\}$	$\{q_3\}$	$\{q_1, q_2, q_3\}$
$* \{q_3\}$	$\emptyset$	$\{q_2\}$
$\{q_2\}$	$\emptyset$	$\{q_1, q_3\}$

Exercício 7 (Desafio 2014/15)

Considere o ε-NFA apresentado de seguida, onde C representa um qualquer caráter do alfabeto (a-z) e D um dígito (0-9). Para que poderá servir este autómato? Converta-o no DFA equivalente.

	ε	C	D	@	.
$\rightarrow$ p	{q,r}	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
q	$\emptyset$	{q,s}	$\emptyset$	$\emptyset$	$\emptyset$
r	$\emptyset$	$\emptyset$	{r,s}	$\emptyset$	$\emptyset$
s	$\emptyset$	{s}	{s}	{t}	$\emptyset$
t	$\emptyset$	{u}	$\emptyset$	$\emptyset$	$\emptyset$
u	$\emptyset$	{u}	$\emptyset$	$\emptyset$	{v}
v	$\emptyset$	{v,w}	$\emptyset$	$\emptyset$	$\emptyset$
* w	{u}	{w}	$\emptyset$	$\emptyset$	$\emptyset$

Olhando apenas para a tabela de transições, torna-se complicado perceber qual a funcionalidade do autómato, muito embora a presença do símbolo '@' no alfabeto nos possa já dar uma dica. Vamos representar o autómato graficamente, para facilitar a interpretação do mesmo:



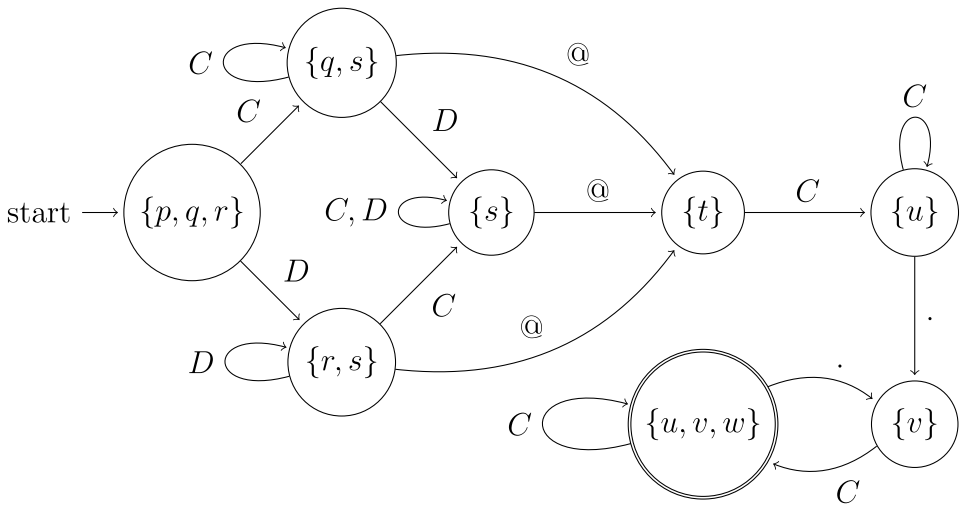
Com o diagrama já se torna mais simples perceber que este autómato reconhece cadeias no formato de um endereço de correio eletrónico, podendo ter um conjunto de dígitos ou letras de comprimento maior ou igual a 1, seguido de um '@', seguido de uma ou mais letras, seguido de um ponto e novamente uma ou mais letras, sendo que a parte final (ponto e letras) pode ser repetida várias vezes.

Para converter para DFA, convém primeiro determinar o fecho-ε de cada estado. Temos então que:

Estado	Fecho-ε
p	$\{p, q, r\}$
q	$\{q\}$
r	$\{r\}$
s	$\{s\}$
t	$\{t\}$
u	$\{u\}$
v	$\{v\}$
w	$\{w, u\}$

Podemos agora fazer a transformação do ε-NFA num DFA, tendo tanto a tabela como o diagrama respetivos:

	C	D	@	.
$\rightarrow \{p, q, r\}$	$\{q, s\}$	$\{r, s\}$	$\emptyset$	$\emptyset$
$\{q, s\}$	$\{q, s\}$	$\{s\}$	$\{t\}$	$\emptyset$
$\{r, s\}$	$\{s\}$	$\{r, s\}$	$\{t\}$	$\emptyset$
$\{s\}$	$\{s\}$	$\{s\}$	$\{t\}$	$\emptyset$
$\{t\}$	$\{u\}$	$\emptyset$	$\emptyset$	$\emptyset$
$\{u\}$	$\{u\}$	$\emptyset$	$\emptyset$	$\{v\}$
$\{v\}$	$\{u, v, w\}$	$\emptyset$	$\emptyset$	$\emptyset$
$* \{u, v, w\}$	$\{u, v, w\}$	$\emptyset$	$\emptyset$	$\{v\}$



Exercício Proposto 1 Considere o ε-NFA abaixo. Indique todas as cadeias de comprimento menor do que 3 aceites pelo autómato, e converta-o para um DFA.