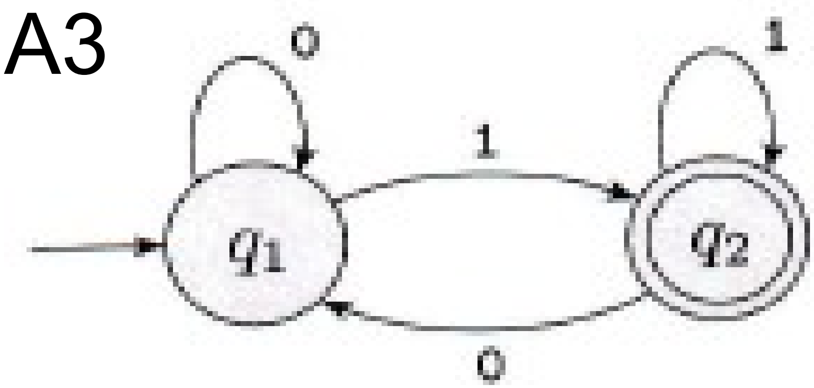
ACH2043
INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 2

Profa. Ariane Machado Lima
ariane.machado@usp.br

Autômatos finitos

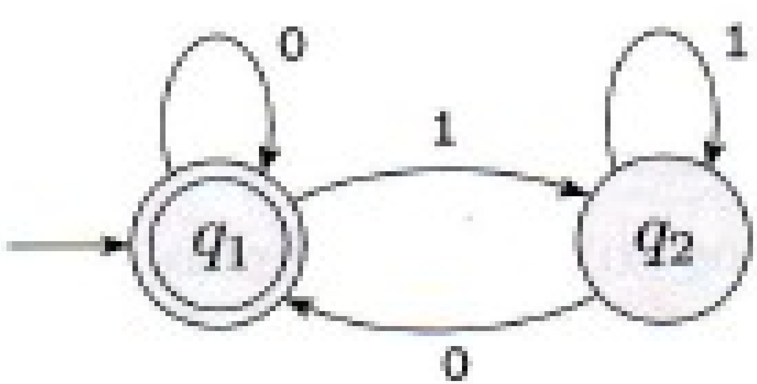
A3



L(A3) = {w | w é binária e termina com 1}

- Que linguagem esse autômato reconhece? (apenas mudou o estado final)

A4



L(A4) = {w | w é binária e termina com 0 ou é vazia}

Provas

- P1: 01/10
- P2: 28/11
- Sub: 5/12

Linguagem Regular

- Uma linguagem é chamada linguagem regular se algum autômato finito a reconhece
- Vamos ver suas propriedades
  - Saber se uma linguagem é regular ou não para sabermos se podemos ou não implementar um autômato finito que a reconheça

Operações regulares

Sejam A e B linguagens. Definimos as operações regulares ***união***, ***concatenação*** e ***estrela*** da seguinte forma.

- **União**: $A \cup B = \{x \mid x \in A \text{ ou } x \in B\}$.
- **Concatenação**: $A \circ B = \{xy \mid x \in A \text{ e } y \in B\}$.
- **Estrela**: $A^* = \{x_1 x_2 \ldots x_k \mid k \geq 0 \text{ e cada } x_i \in A\}$.

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

$$A \cup B =$$

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\texttt{a}, \texttt{b}, \ldots, \texttt{z}\}$. Se $A = \{\texttt{legal}, \texttt{ruim}\}$ e $B = \{\texttt{garoto}, \texttt{garota}\}$, então

$$A \cup B = \{\texttt{legal}, \texttt{ruim}, \texttt{garoto}, \texttt{garota}\}$$

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

$$A \cup B = \{\mathtt{legal}, \mathtt{ruim}, \mathtt{garoto}, \mathtt{garota}\}$$

$$A \circ B =$$

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

$$A \cup B = \{\mathtt{legal}, \mathtt{ruim}, \mathtt{garoto}, \mathtt{garota}\}$$

$$A \circ B = \{\mathtt{legalgaroto}, \mathtt{legalgarota}, \mathtt{ruimgaroto}, \mathtt{ruimgarota}\}$$

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

$$A \cup B = \{\mathtt{legal}, \mathtt{ruim}, \mathtt{garoto}, \mathtt{garota}\}$$

$$A \circ B = \{\mathtt{legalgaroto}, \mathtt{legalgarota}, \mathtt{ruimgaroto}, \mathtt{ruimgarota}\}$$

$$A^* =$$

Operações regulares

Suponha que o alfabeto Σ seja o alfabeto padrão de 26 letras $\{\mathtt{a}, \mathtt{b}, \ldots, \mathtt{z}\}$. Se $A = \{\mathtt{legal}, \mathtt{ruim}\}$ e $B = \{\mathtt{garoto}, \mathtt{garota}\}$, então

$$A \cup B = \{\mathtt{legal}, \mathtt{ruim}, \mathtt{garoto}, \mathtt{garota}\}$$

$$A \circ B = \{\mathtt{legalgaroto}, \mathtt{legalgarota}, \mathtt{ruimgaroto}, \mathtt{ruimgarota}\}$$

$$\begin{aligned} A^* = \{&\varepsilon, \mathtt{legal}, \mathtt{ruim}, \mathtt{legallegal}, \mathtt{legalruim}, \mathtt{ruimlegal}, \mathtt{ruimruim}, \\ &\mathtt{legallegallegal}, \mathtt{legallegalruim}, \mathtt{legalruimlegal}, \\ &\mathtt{legalruimruim}, \ldots\}. \end{aligned}$$

Fechamento sob união

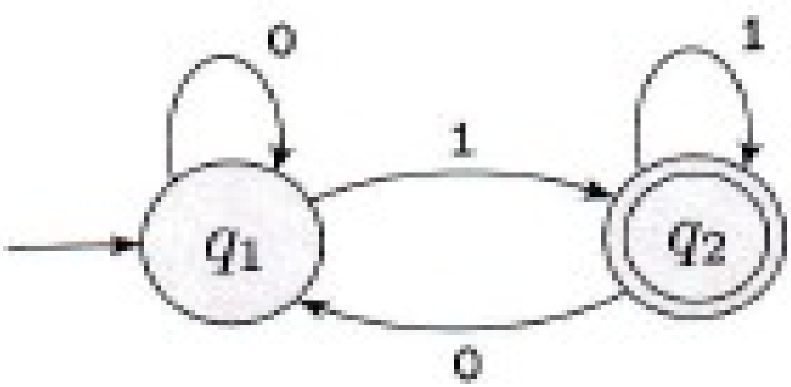
TEOREMA 1.25

A classe de linguagens regulares é fechada sob a operação de união.

Em outras palavras, se A_1 e A_2 são linguagens regulares, o mesmo acontece com $A_1 \cup A_2$.

Autômatos Finitos Determinísticos (AFD)

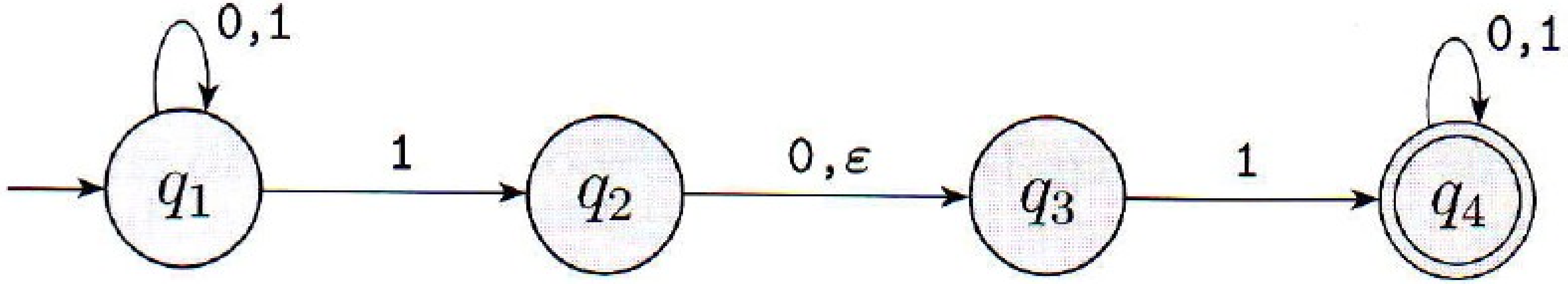
- Dado um estado atual e um símbolo de entrada sabemos exatamente para onde ir (está determinado)



Um ***autômato finito*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

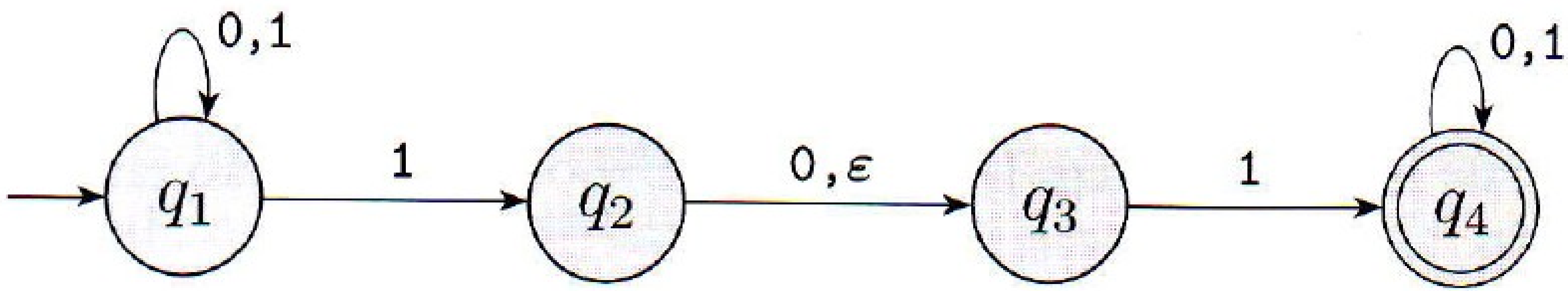
1. Q é um conjunto finito conhecido como os ***estados***,
2. Σ é um conjunto finito chamado o ***alfabeto***,
3. $\delta : Q \times \Sigma \longrightarrow Q$ é a ***função de transição***,[1]
4. $q_0 \in Q$ é o ***estado inicial***, e
5. $F \subseteq Q$ é o ***conjunto de estados de aceitação***.[2]

Autômatos Finitos Não Determinísticos (AFN)



- Um estado pode ter 0 ou mais transições (setas saindo) para cada símbolo de Σ
- Um estado pode ter setas rotuladas por ε

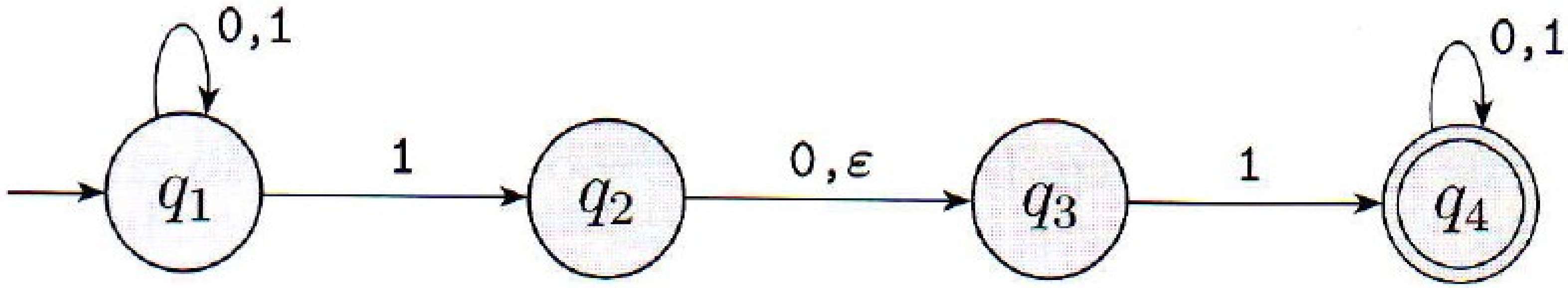
Autômatos Finitos
Não Determinísticos (AFN)



Um ***autômato finito não-determinístico*** é uma 5-upla $(Q, \Sigma, \delta, q_0, F)$, onde

1. Q é um conjunto finito de estados,
2. Σ é um alfabeto finito,
3. $\delta: Q \times \Sigma_\varepsilon \longrightarrow \mathcal{P}(Q)$ é a função de transição,
4. $q_0 \in Q$ é o estado inicial, e
5. $F \subseteq Q$ é o conjunto de estados de aceitação.

Autômatos Finitos Não Determinísticos (AFN)



1. $Q = \{q_1, q_2, q_3, q_4\}$,
2. $\Sigma = \{0,1\}$,
3. δ é dado como

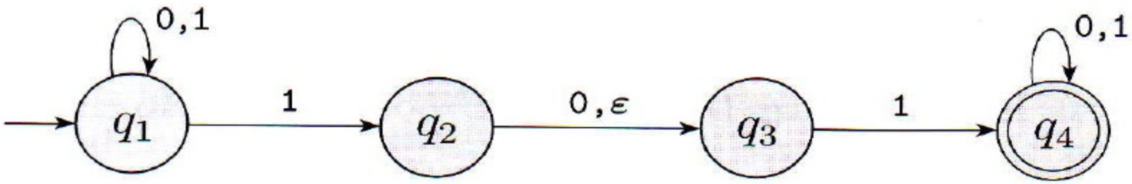
	0	1	ε
q_1	$\{q_1\}$	$\{q_1, q_2\}$	$\emptyset$
q_2	$\{q_3\}$	$\emptyset$	$\{q_3\}$
q_3	$\emptyset$	$\{q_4\}$	$\emptyset$
q_4	$\{q_4\}$	$\{q_4\}$	$\emptyset$

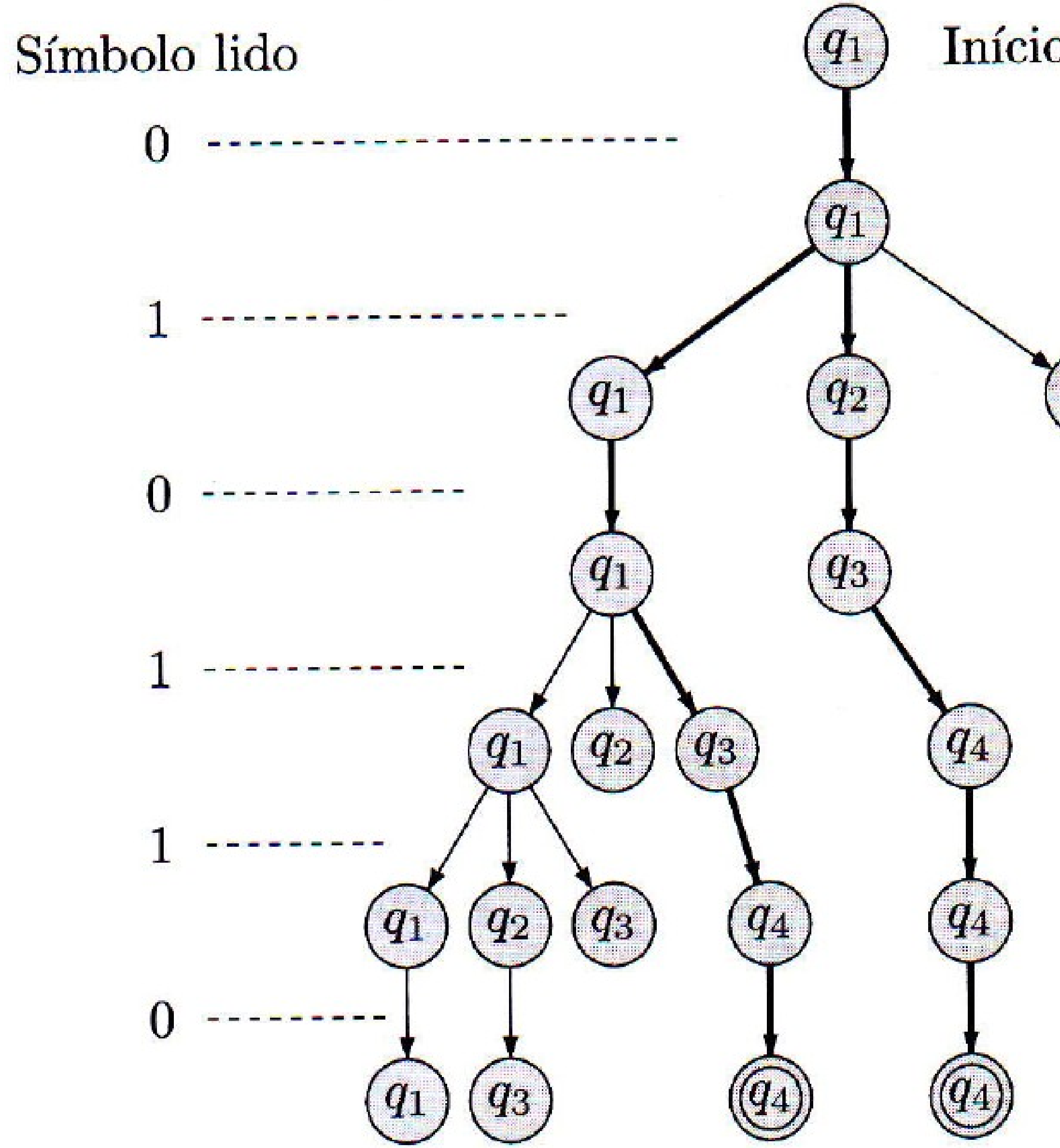
,

4. q_1 é o estado inicial, e
5. $F = \{q_4\}$.

Funcionamento de um AFN

- Sempre que o autômato se depara com um não-determinismo (símbolo repetido ou ε) faz uma cópia de si e cada cópia segue com uma alternativa, em paralelo.
- Se uma cópia aceitar a cadeia, então o AFN aceita a cadeia


0,1
1
0,ε
1
0,1
q_1
q_2
q_3
q_4


Símbolo lido
Início
0
1
0
1
1
0

AFDs e AFNs

- Quem reconhece mais linguagens?

AFDs e AFNs

- Quem reconhece mais linguagens?
- Os dois reconhecem a mesma classe de linguagens

Equivalência entre AFDs e AFNs

- Duas máquinas são equivalentes se elas reconhecem a mesma linguagem

TEOREMA 1.39

Todo autômato finito não-determinístico tem um autômato finito determinístico equivalente.

COROLÁRIO 1.40

Uma linguagem é regular se e somente se algum autômato finito não-determinístico a reconhece.

AFDs e AFNs

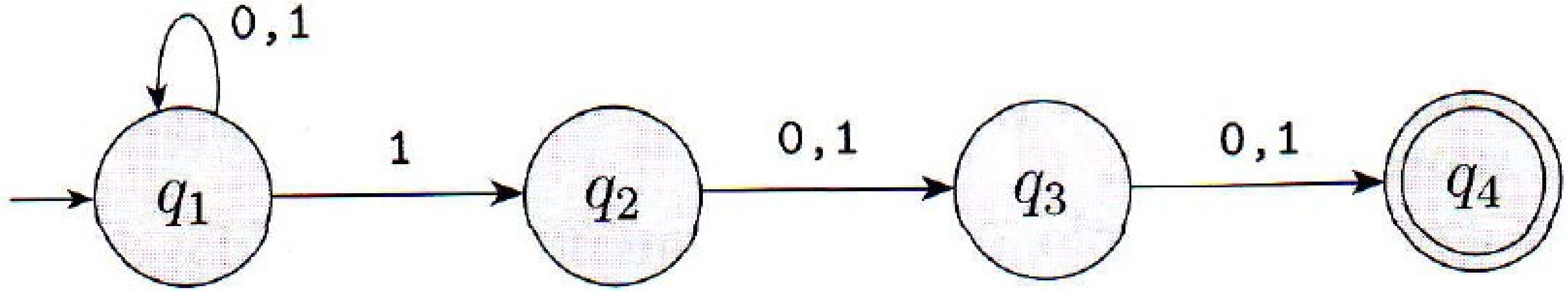
- Por que o teorema de equivalência é importante?
- Pode-se optar por um outro dependendo do objetivo
- AFDs são mais eficientes
- AFNs podem:
  - ser mais fáceis de serem projetados
  - facilitar demonstração de teoremas
  - ser úteis em versões probabilísticas

AFNs mais fáceis de serem projetados

- Ex: projete um AFN que reconheça strings binárias que contenham 1 na antepenúltima posição

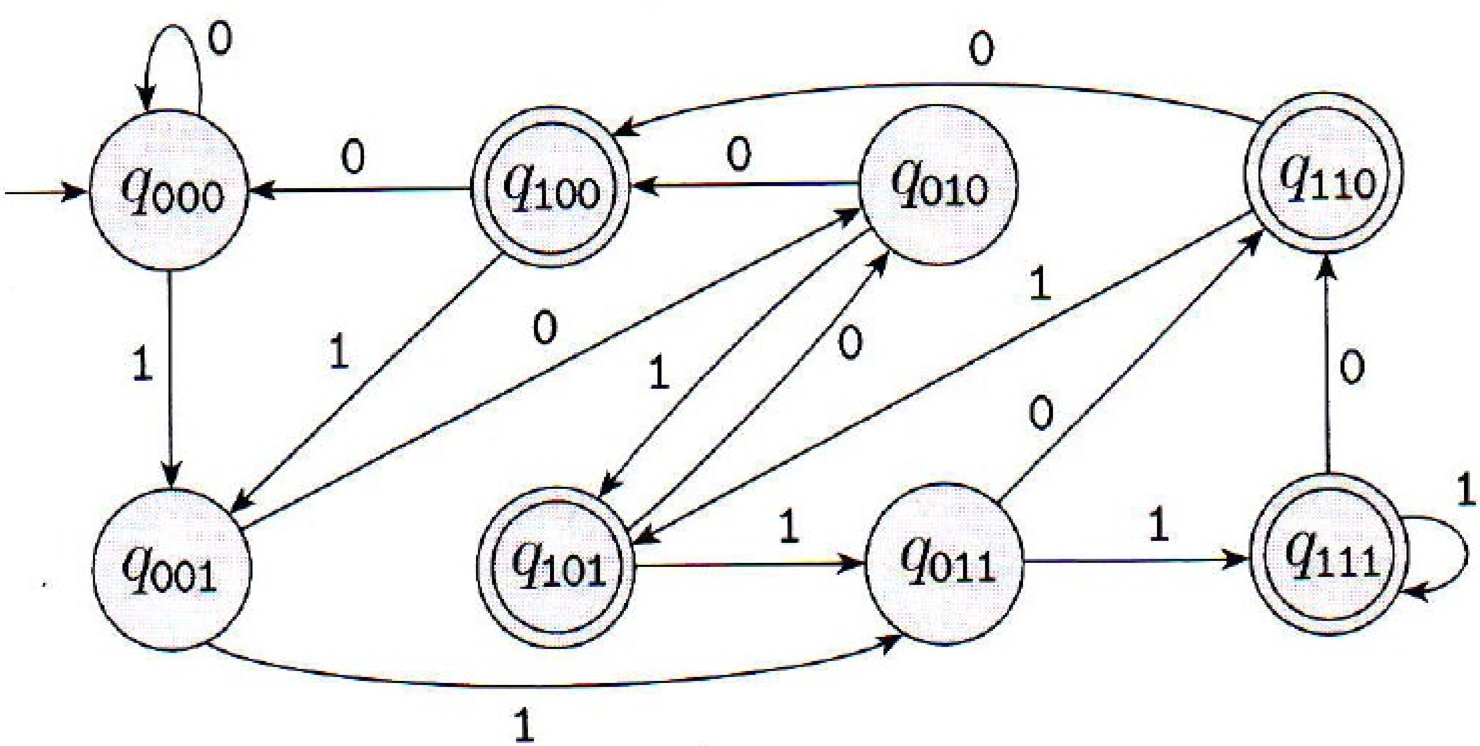
AFNs mais fáceis de serem projetados

- Ex: projete um AFN que reconheça strings binárias que contenham 1 na antepenúltima posição



AFNs mais fáceis de serem projetados

- AFD equivalente:



AFNs facilitando provas de teoremas

- Veremos nas próximas aulas os fechos de linguagens regulares sob as operações regulares

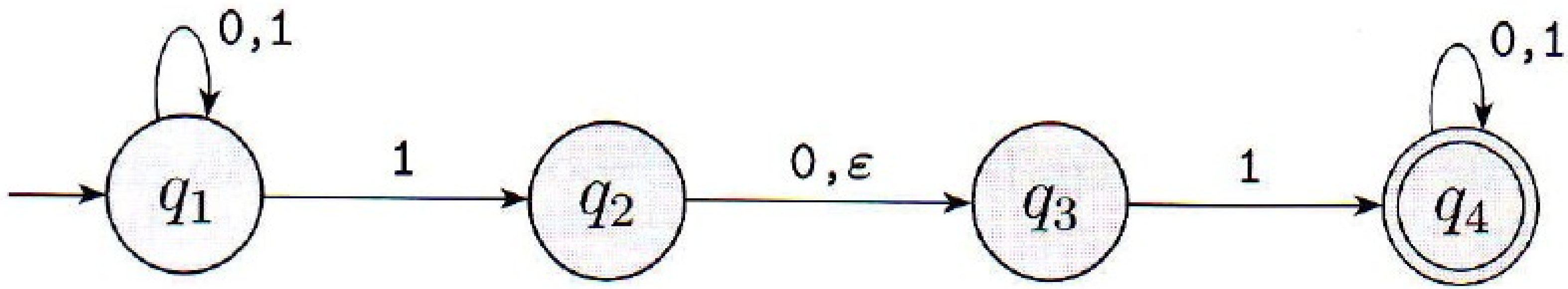
AFNs probabilísticos

- Um autômato probabilístico possui uma distribuição de probabilidades sobre as transições de cada estado
- δ: Q x $\Sigma \rightarrow \mathcal{P}$(Q)
- P: Q x $\Sigma \rightarrow [0,1]$

onde $\sum_j (q_i, a_j) = 1$ para q_i em Q

AFNs probabilísticos
Exemplo

- Transformar esse autômato em probabilístico

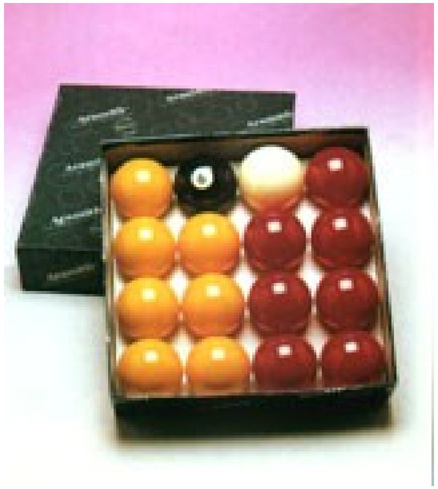


Modelo Oculto de Markov
Hidden Markov Model (HMM)

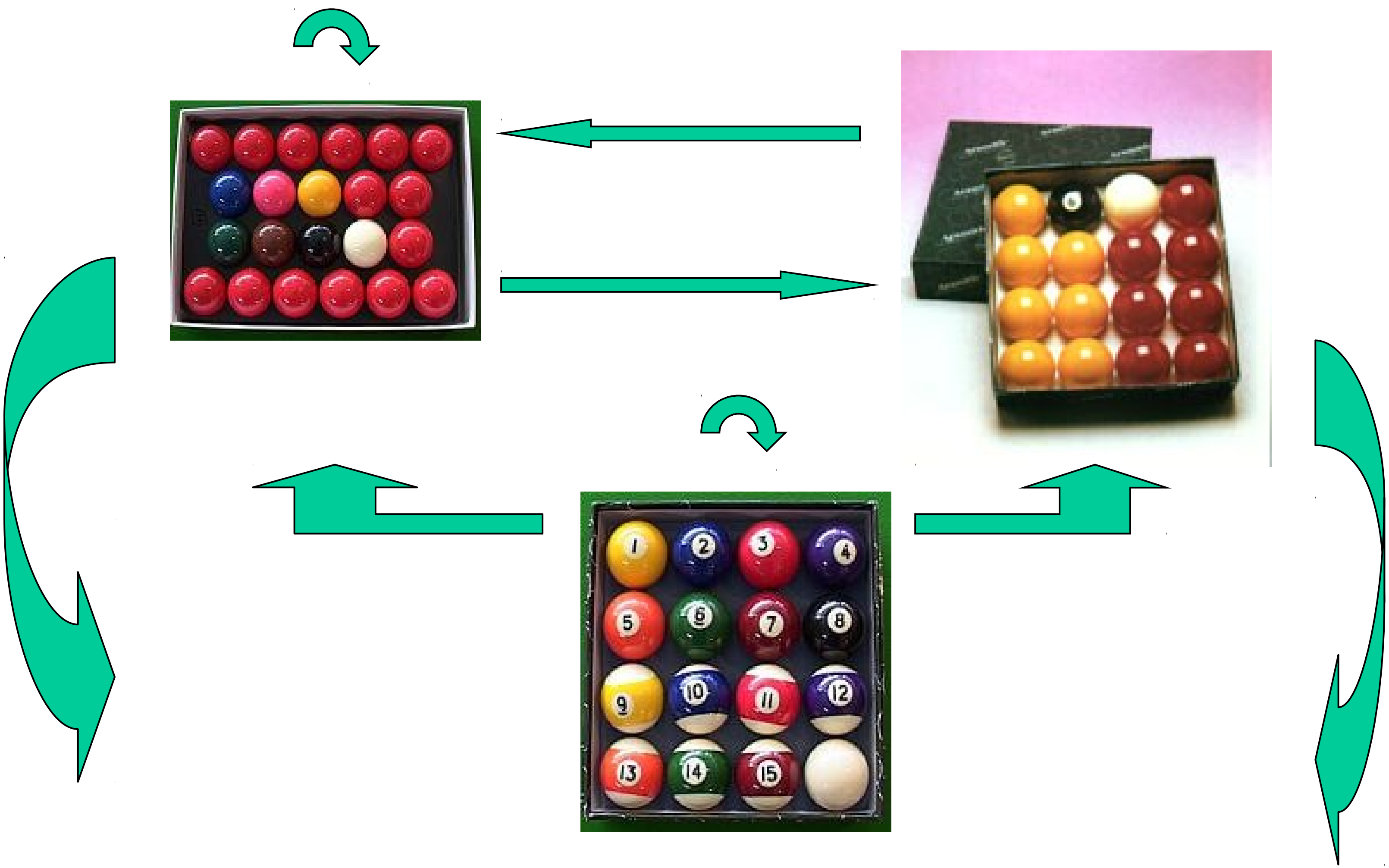
- Imagine que eu tenho uma urna de bolas coloridas (ou seja, tenho uma distribuição de probabilidades sobre essas cores)

- Alguém tem que adivinhar qual a próxima cor. Como isso poderia ser feito?

Modelo Oculto de Markov
Hidden Markov Model (HMM)



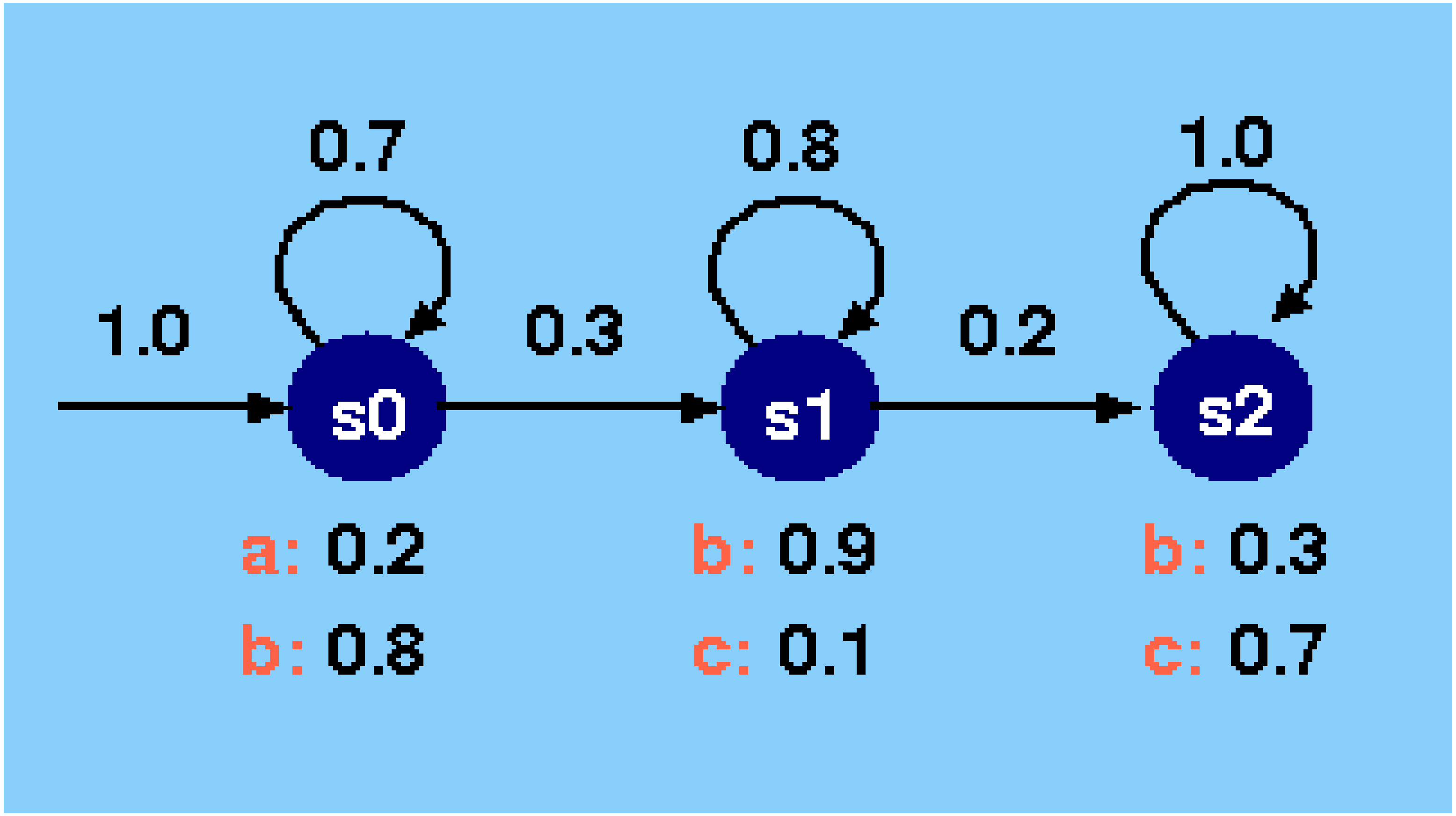
Modelo Oculto de Markov
Hidden Markov Model (HMM)



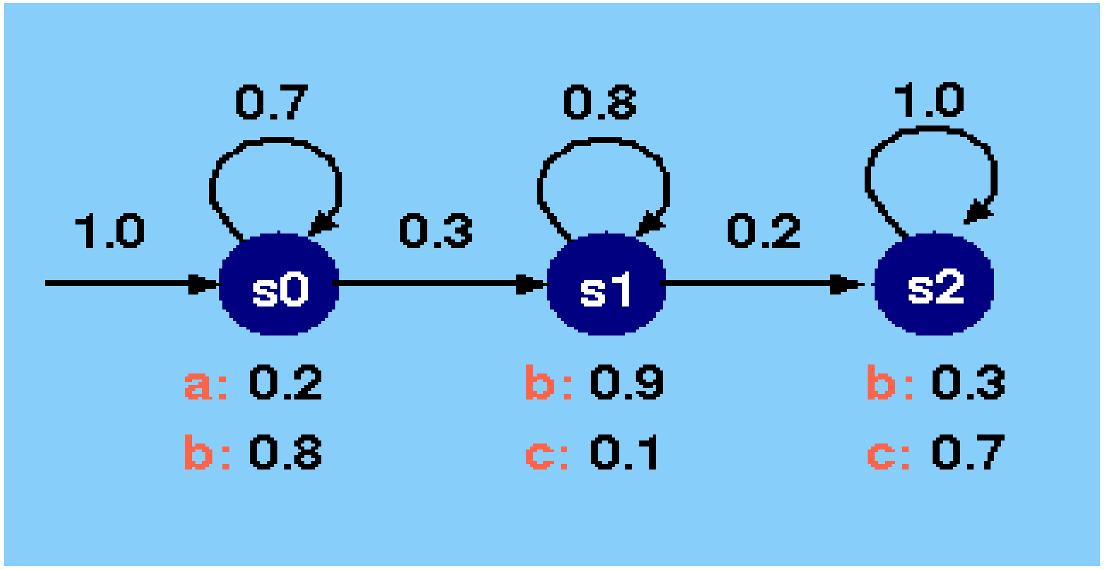
Modelo Oculto de Markov
Hidden Markov Model (HMM)

- Símbolos de emissão
- Estados ocultos
- Uma distribuição de probabilidades de emissão de símbolos associada a cada estado
- Probabilidade de transição entre estados
- Distribuição de probabilidades do estado inicial

Modelo Oculto de Markov
Hidden Markov Model (HMM)

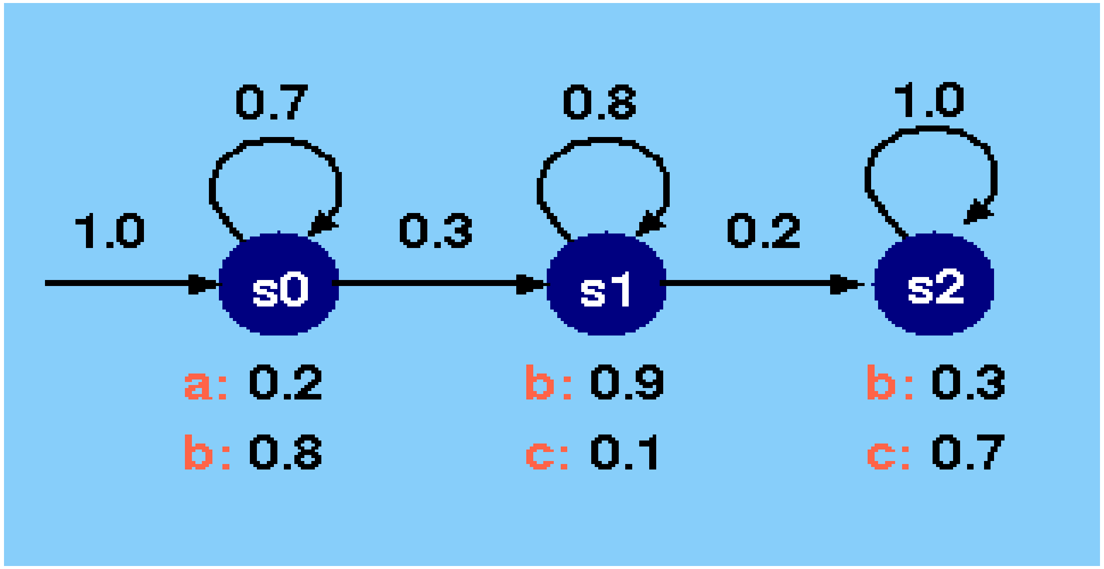


Modelo Oculto de Markov
Hidden Markov Model (HMM)



- Semelhança com algo?

Modelo Oculto de Markov
Hidden Markov Model (HMM)



- Semelhança com algo?
- Como transformo essa HMM em um AF probabilístico?

Modelo Oculto de Markov
Hidden Markov Model (HMM)

- Autômatos finitos (não determinísticos) probabilísticos são equivalentes a modelos ocultos de Markov

Problemas relacionados a HMM

- Dados um HMM e uma cadeia, calcular a probabilidade dessa cadeia
  - Soma das probabilidades de cada caminho
  - Probabilidade de cada caminho: produto das probabilidades do caminho (transição e emissão)
  - Algoritmo forward ou backword

Problemas relacionados a HMM

- Dados um HMM e uma cadeia, calcular o caminho mais provável dessa cadeia
  - Algoritmo viterbi

3) Dados um HMM e conjunto de cadeias (treinamento), estimar os parâmetros (probabilidades de emissão e transição)
  - Algoritmo Baum-Welch

Problemas relacionados a HMM

- Projetar a topologia de uma HMM