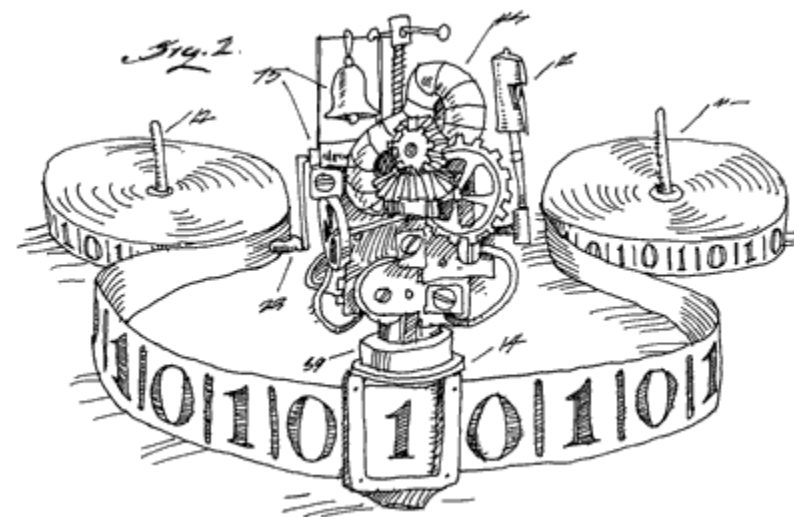


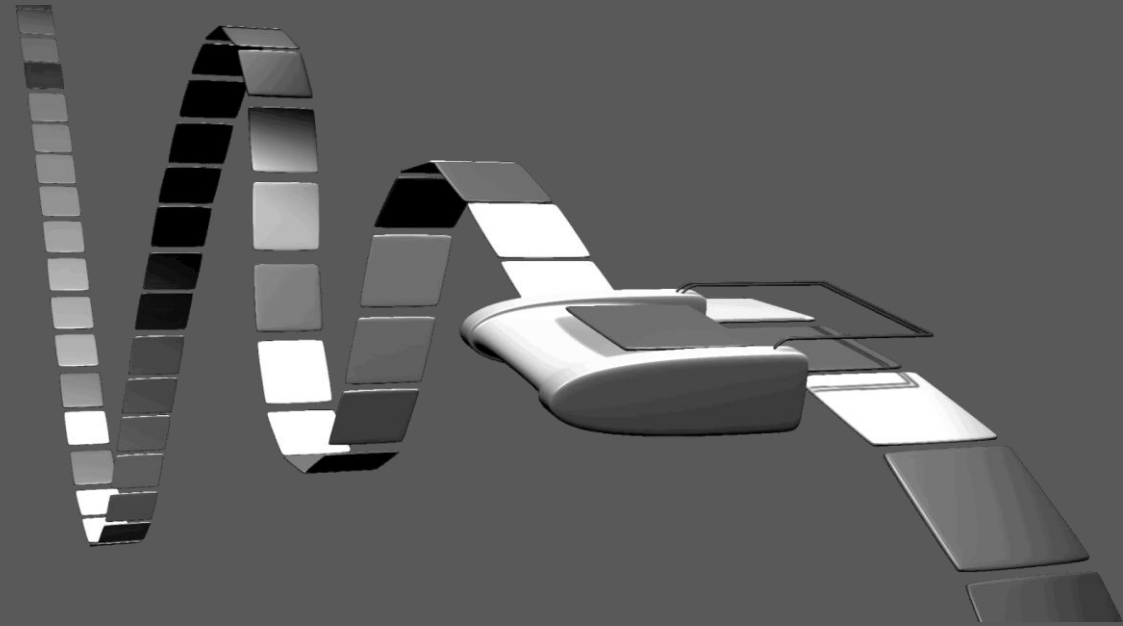
Autômatos Finitos Não Determinísticos

Introdução à Teoria da Computação

Nicollas Mocelin Sdroievski

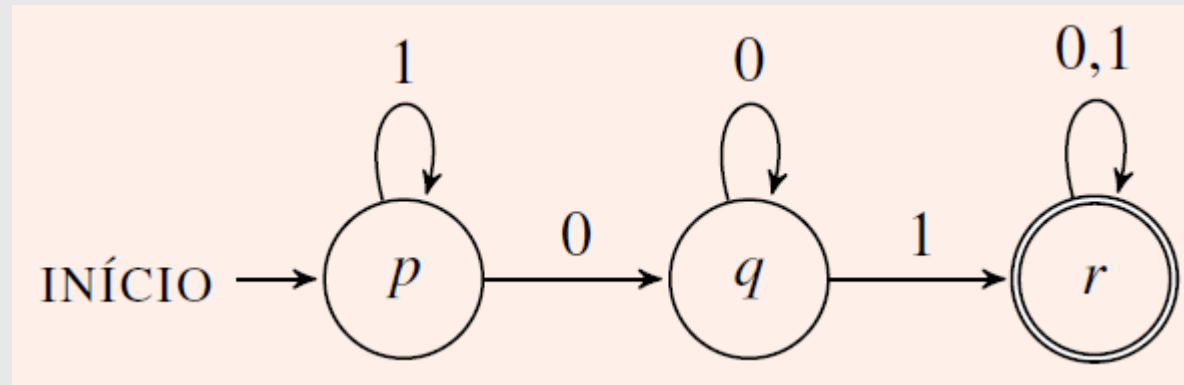


Relembrando



Autômatos Finitos Determinísticos

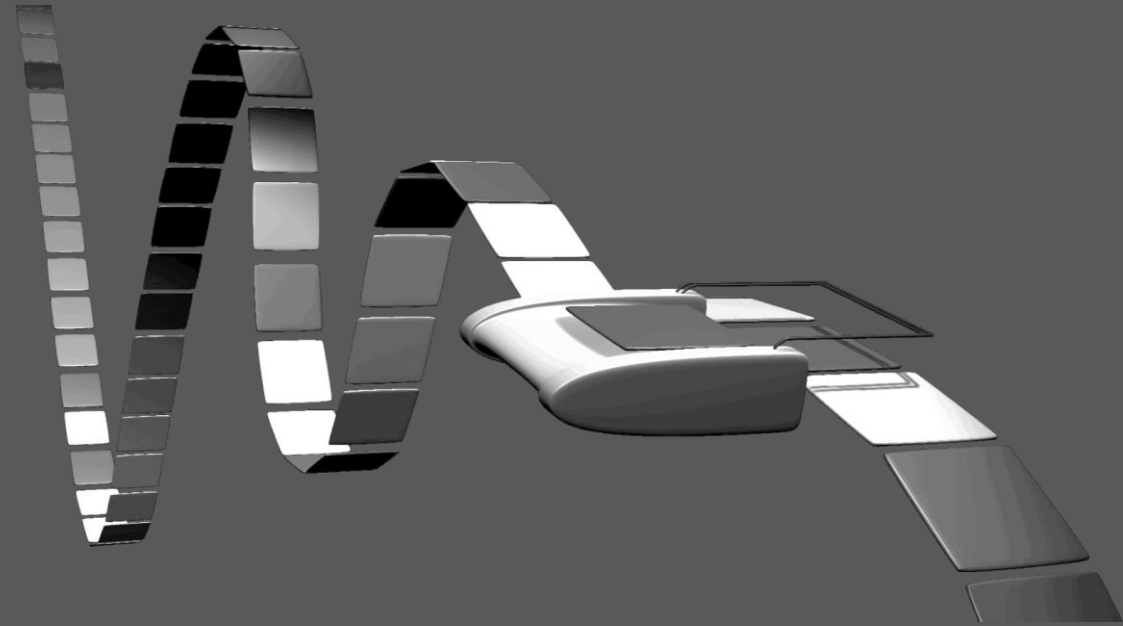
- Definidos por uma quintupla $D = (Q, \Sigma, \delta, q_0, F)$
 - Q = conjunto de estado
 - Σ = alfabeto
 - δ = função de transição
 - q_0 = estado inicial
 - F = conjunto de estados finais



Função de transição estendida

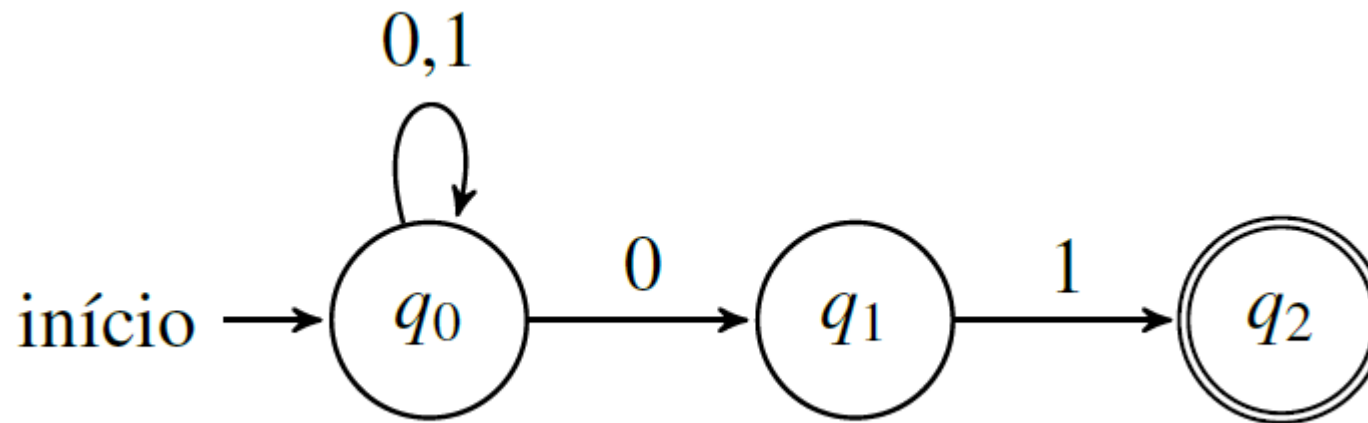
- $\delta(q, a)$ = estado alcançado iniciando no estado q e lendo o símbolo a
- $\hat{\delta}(q, w)$ = estado alcançado iniciando no estado q e lendo a **string** w
- **Definição.** Dado um AFD $D = (Q, \Sigma, \delta, q_0, F)$, a função de transição estendida $\hat{\delta}$ de D é a função $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$ definida indutivamente:
 - **Caso base** $w = \epsilon$
 - $\hat{\delta}(q, \epsilon) = q$
 - **Indução** $|w| > 0$
 - $w = xa, x \in \Sigma^*$ e $a \in \Sigma$. Então $\hat{\delta}(q, w) = \hat{\delta}(q, xa) = \delta(\hat{\delta}(q, x), a)$

AFNs



Autômatos Finitos Não Determinísticos

- AFDs são completamente determinísticos
 - Para cada (q, a) , exatamente uma transição
- AFNs possuem várias escolhas possíveis para cada transição



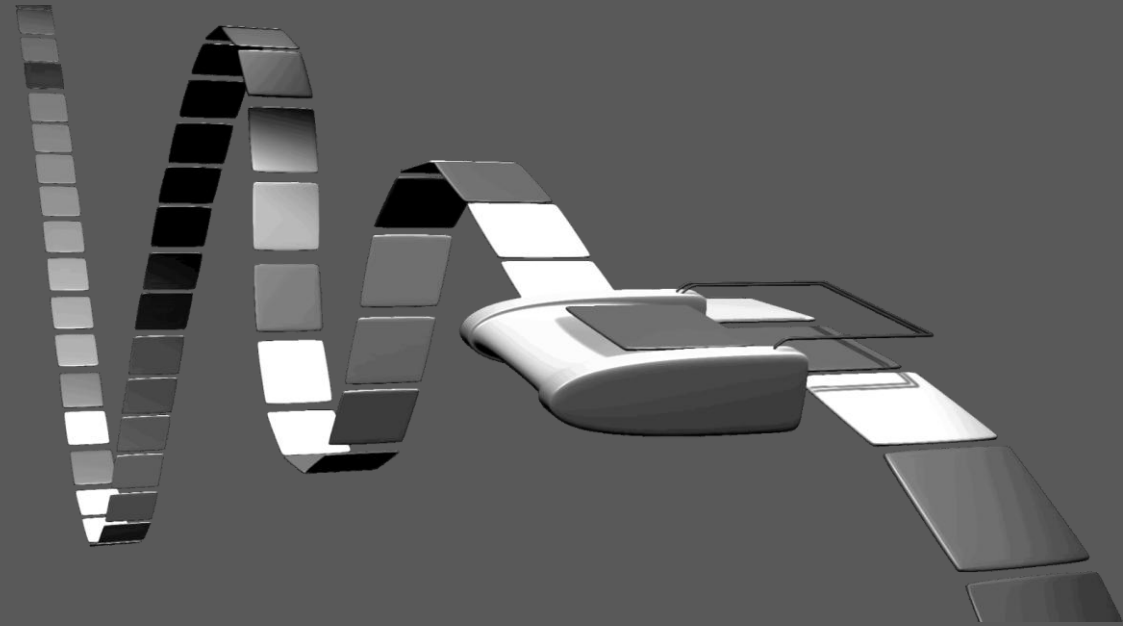
Como funciona um AFN?

- Um AFN “adivinha” qual é a transição correta a ser escolhida
- Por exemplo, o AFN do slide anterior aceita a string 001
 - Através do caminho $q_0 \rightarrow q_0 \rightarrow q_1 \rightarrow q_2$
 - Mesmo existindo o caminho $q_0 \rightarrow q_0 \rightarrow q_0 \rightarrow q_0$ de não-aceitação
- Se não houver nenhum caminho que leva à aceitação, então o AFN rejeita
 - Esse é o caso da string 000
- Linguagem do AFN de exemplo?
- Um AFN também pode não ter uma transição definida
 - Nesse caso, o autômato *morre*

Determinismo e não determinismo

- Modelos determinísticos são concretos e implementáveis
- Modelos não determinísticos parecem estranhos
 - Não possuem implementação física
- Por que estudar esses modelos?
 - Ferramentas matemáticas úteis

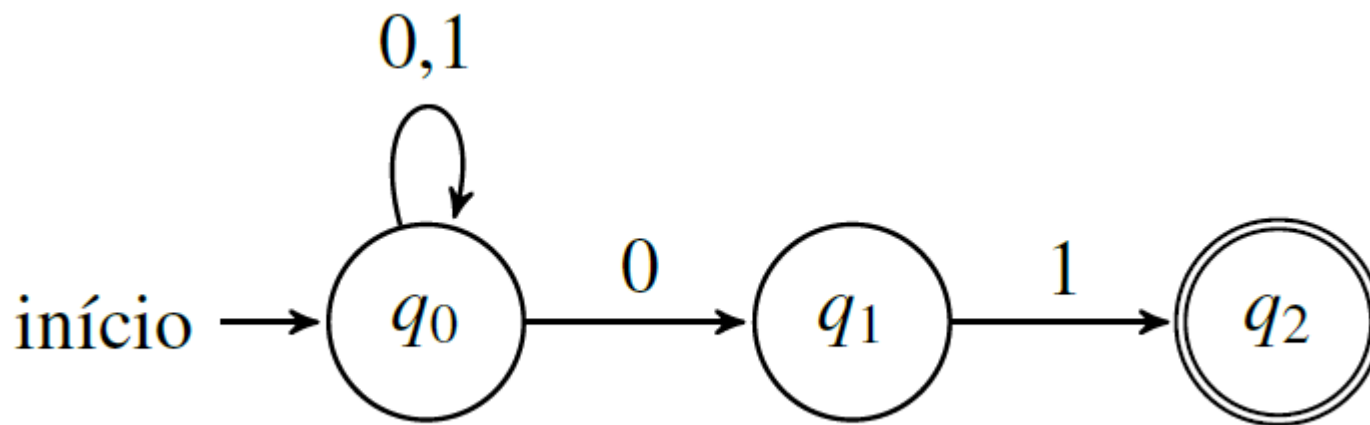
Descrição formal



Definição de AFN

- **Definição.** Um Autômato Finito não Determinístico, também chamado de AFN, é uma 5-tupla $A = (Q, \Sigma, \delta, q_0, F)$, tal que:
 - Q é o conjunto de estados
 - Σ é o alfabeto de entrada
 - δ é a função de transição $\delta: Q \times \Sigma \rightarrow \mathcal{P}(Q)$
 - $q_0 \in Q$ é o estado inicial
 - $F \subseteq Q$ é o conjunto de estados finais
- Diferença com a definição de um AFD?

Definição para o AFN de exemplo



■ **Exemplo 3.2** A definição formal para o diagrama da Figura 3.2 é o AFN $N = (Q, \Sigma, \delta, q_0, F)$, tal que $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{0, 1\}$ e a função δ é definida abaixo:

$$\delta(q_0, 0) = \{q_0, q_1\}$$

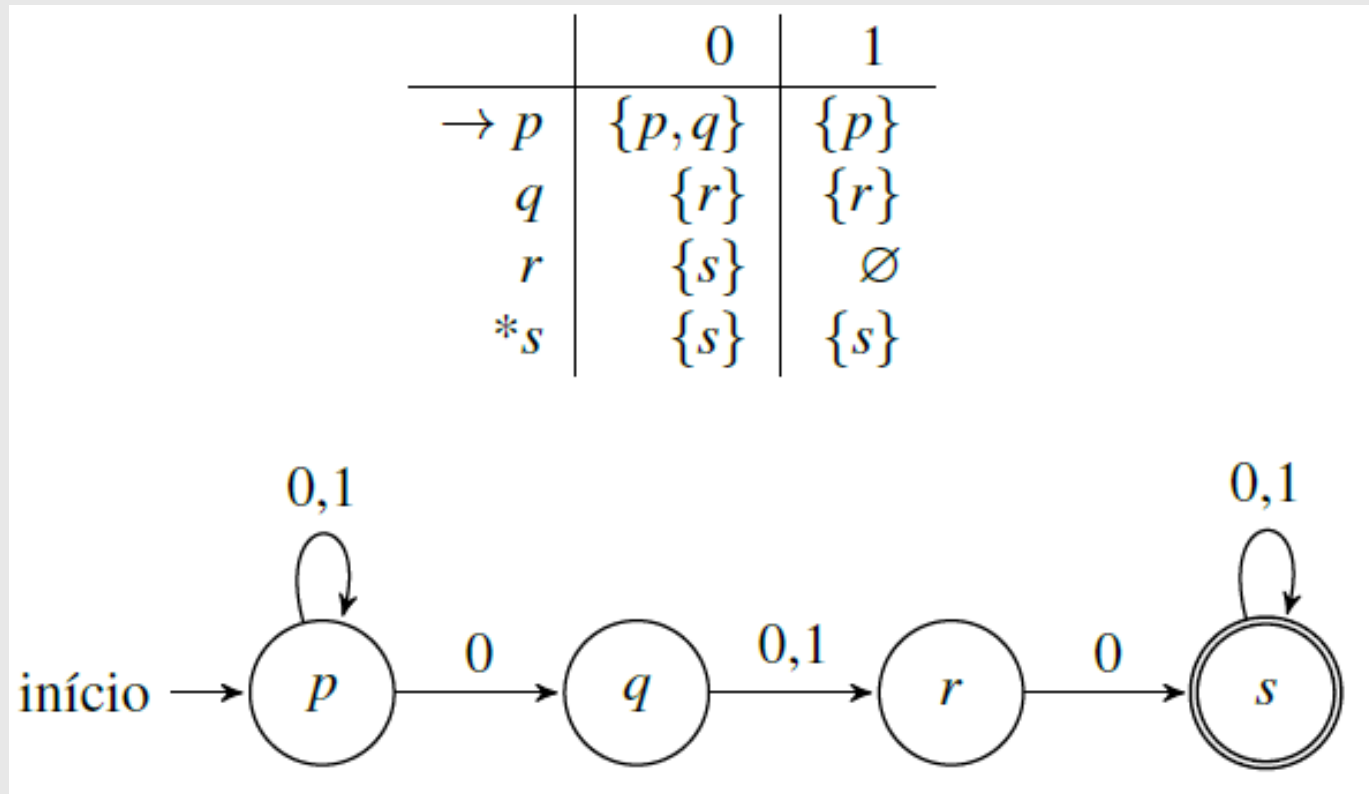
$$\delta(q_0, 1) = \{q_0\}$$

$$\delta(q_1, 1) = \{q_2\}$$

$$\delta(q_1, 0) = \delta(q_2, 0) = \delta(q_2, 1) = \emptyset$$

Tabela de transições para AFNs

- Mesma ideia que a tabela de transições para AFDs
- Lembrando que a imagem da função δ é diferente



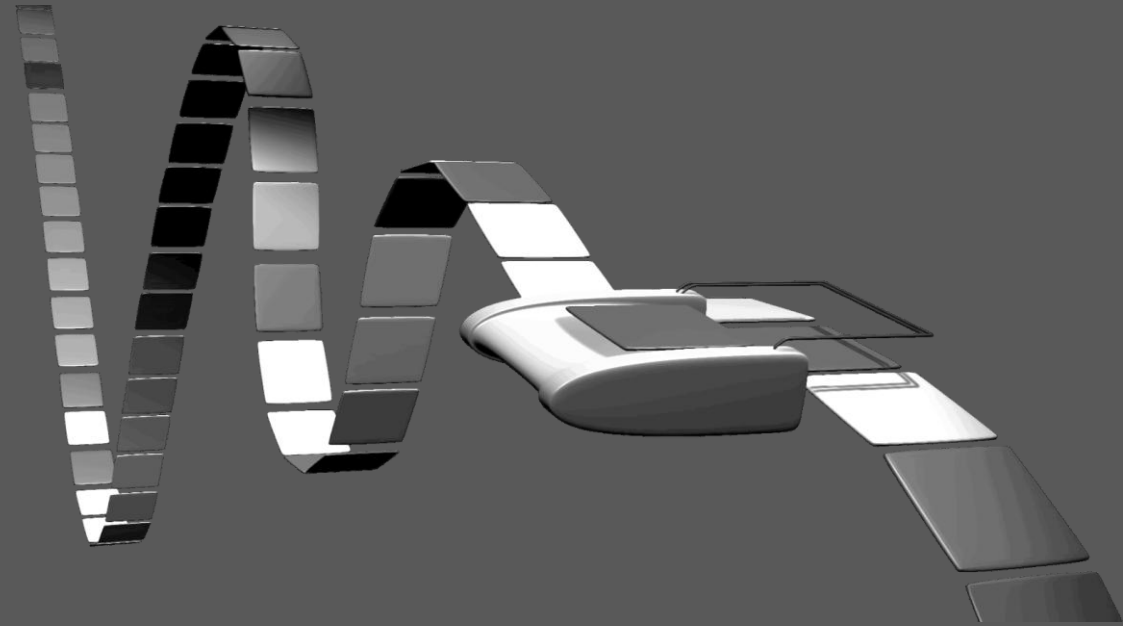
Exercício

Exercício 3.11 (Resolvido) Apresente uma definição formal para um AFN cujo diagrama seja idêntico ao diagrama do AFD obtido na solução do Exercício 3.1.

Solução: Relembramos que o AFD do Exercício 3.1 é definido pela seguinte tabela:

	0	1
$\rightarrow p$	q	p
q	q	r
$*r$	r	r

Árvore de computações

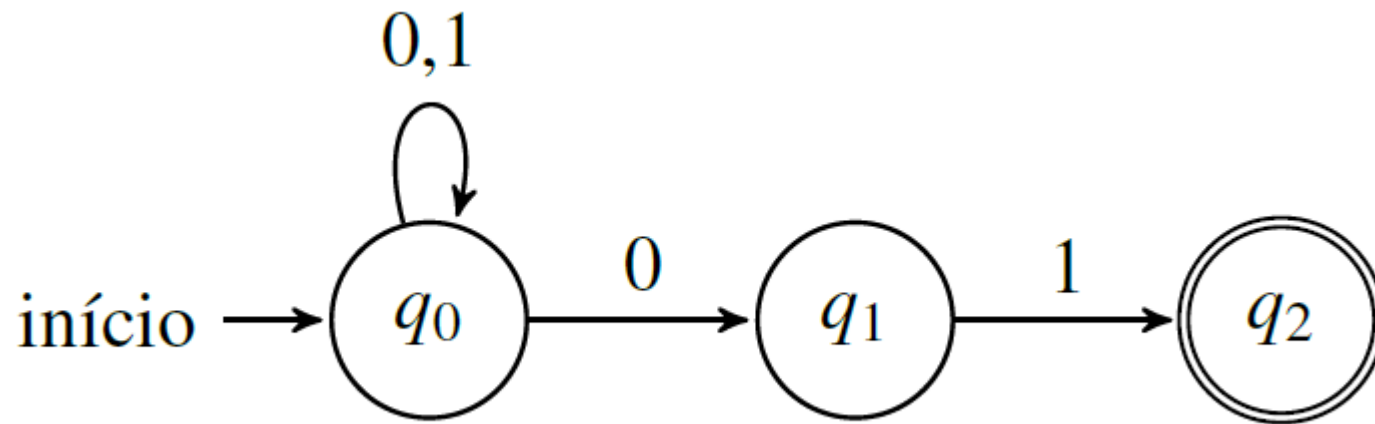


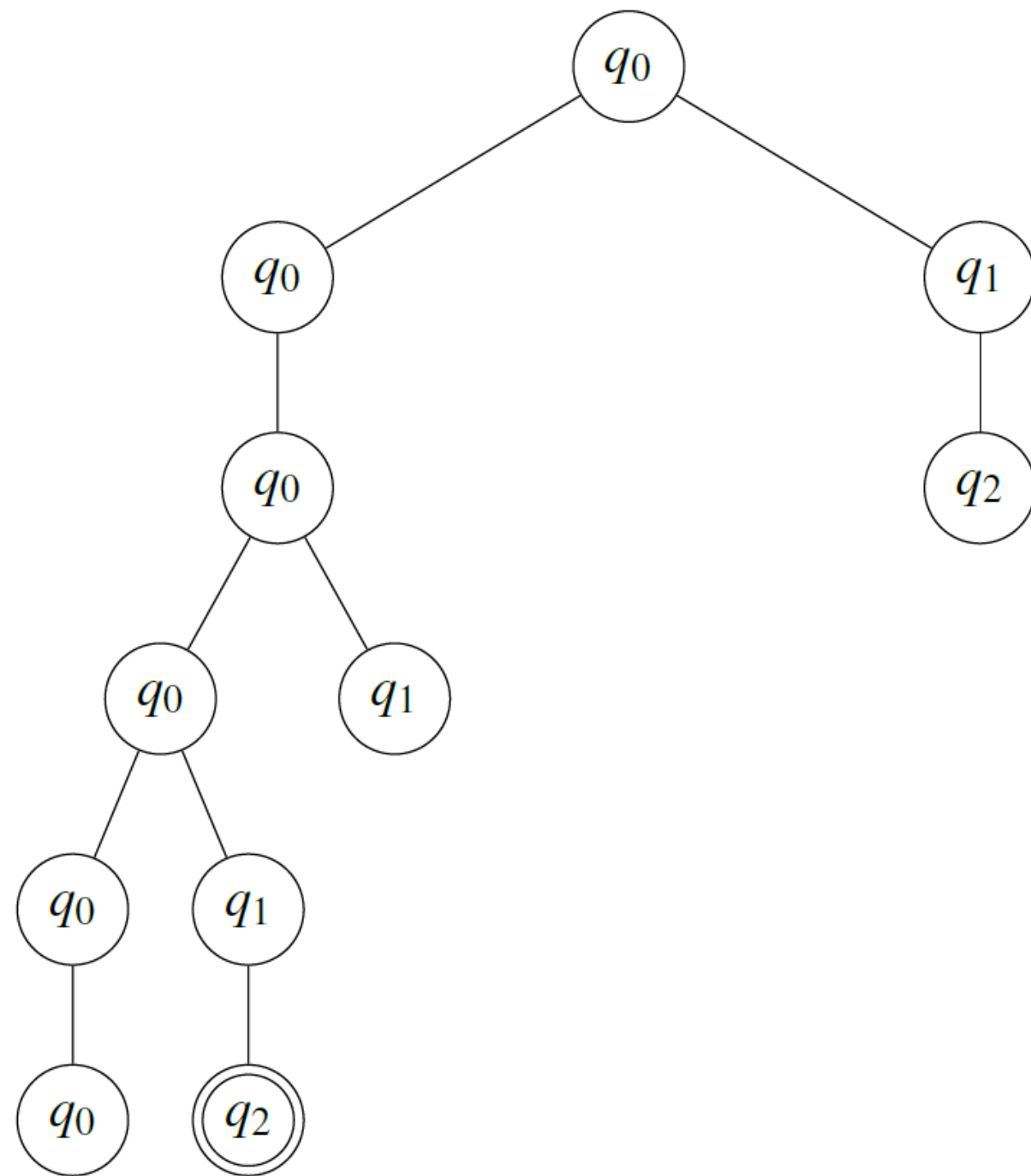
Árvores de computações

- Maneira de descrever os possíveis ramos de computação de um AFN A com uma string w
- **Definição.** Seja (A, q, w) uma tripla tal que A é um AFN, q é um estado de A e w uma string do alfabeto de A . Uma (A, q, w) -árvore é uma árvore enraizada em q tal que
 - **Base:** $w = \varepsilon$
 - A árvore contém apenas o nó raiz q
 - **Indução:** $|w| > 0$
 - Suponha que $w = ax$, com $a \in \Sigma$ e $x \in \Sigma^*$, e que δ é a função de transição de A . Se $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$, então a árvore contém o nó q , que possui os filhos $p_1, p_2, \dots, p_k$, que são raízes de uma (A, p_i, x) -árvore

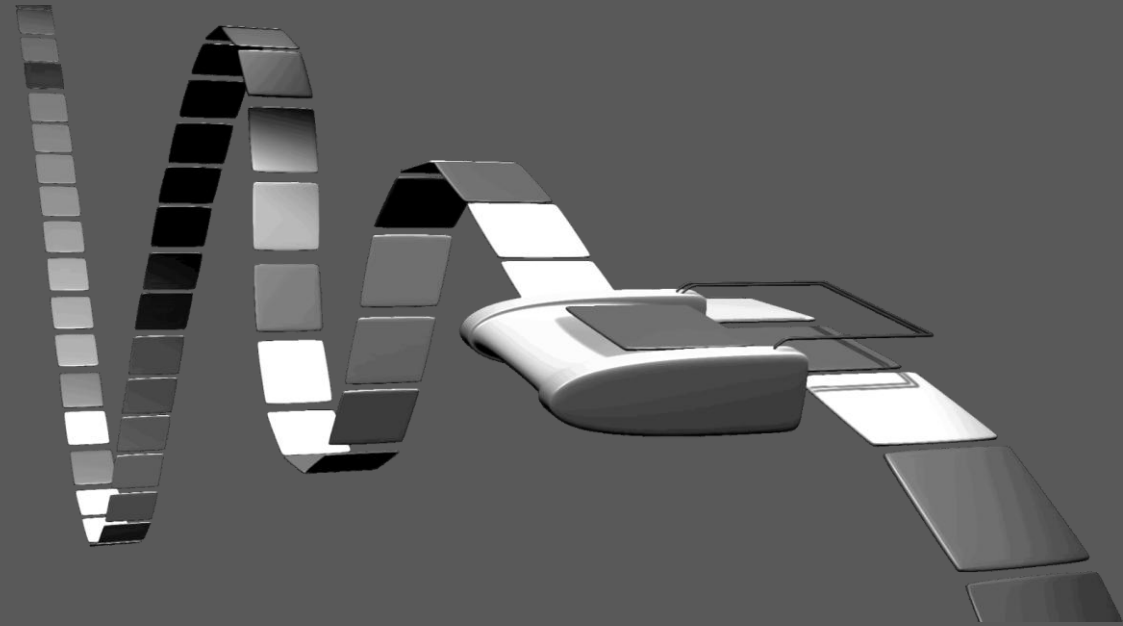
Árvore de computações possíveis de um AFN

- **Definição.** Seja $A = (Q, \Sigma, \delta, q_0, F)$ um AFN e $w \in \Sigma^*$. Uma árvore de computações possíveis de A com w é uma (A, q_0, w) -árvore.
- Exemplo no próximo slide para o autômato de exemplo e a string $w = 01001$





Aceitação e
rejeição de strings



A função $\hat{\delta}$ para AFNs

- Dessa vez, queremos saber o **conjunto de estados** em que o autômato pode estar após o processamento da string w
- **Definição.** Dado um AFN $N = (Q, \sigma, \delta, q_0, F)$, definimos $\hat{\delta}: Q \times \Sigma^* \rightarrow \mathcal{P}(Q)$:
 - **Base:** $w = \varepsilon$
 - $\hat{\delta}(q, \varepsilon) = \{q\}$
 - **Indução:** $|w| > 0$
 - Seja $w \in \Sigma^*$ da forma xa , onde $x \in \Sigma^*$ e $a \in \Sigma$. Suponha $\hat{\delta}(q, x) = \{p_1, p_2, \dots, p_k\}$. Então
 - $\hat{\delta}(q, w) = \cup_{i=1}^k \delta(p_i, a)$

Exercício 3.13 Calcule $\hat{\delta}(q_0, 00101)$ do AFN do Exemplo 3.2.

Linguagem de um AFN

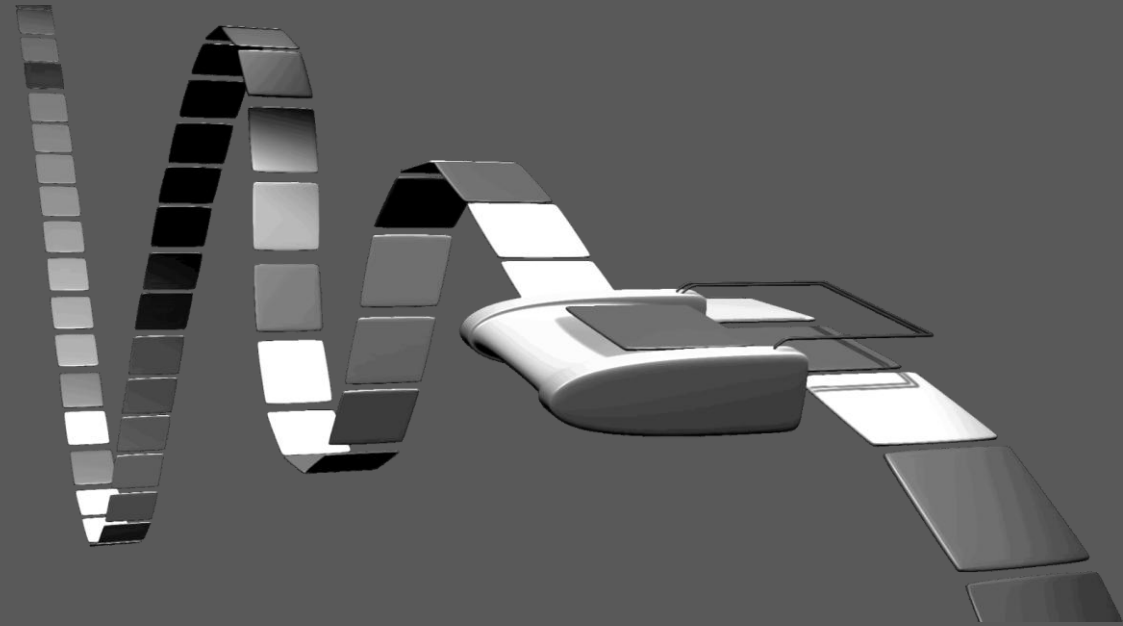
- **Definição.** Seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN, então $L(N) = \{ w \mid \hat{\delta}(q_0, w) \cap F \neq \emptyset \}$ é a linguagem de N .

Exercício 3.14 (Resolvido) Seja $D = (Q, \Sigma, \delta, q_0, F)$ um AFD. Apresente um AFN equivalente, ou seja, um AFN N tal que $L(D) = L(N)$.

- Livro tem a definição (3.2.6) de árvore de computações para um AFD
 - Essencialmente a árvore de computações para o AFN equivalente ao AFD em questão

Exercício 3.15 Qual é o formato de uma árvore de computações possíveis de um AFD com uma string qualquer? ■

Para fechar



Para fechar

- Hoje
 - AFNs
 - Versão não determinística de AFDs
 - Capazes de reconhecer linguagens regulares
 - AFDs são um caso específico
- Próxima aula
 - Equivalência entre AFNs e AFDs
 - AFDs aceitam/reconhecem **exatamente** as linguagens regulares
 - Algoritmo de construção de um AFD a partir de um AFN

Referências

- [SIL25] – SILVA, M.V.G. Autômatos, Computabilidade e Complexidade Computacional , 2025.