

# No-determinismo

Clase 03

IIC2223 / IIC2224

Prof. Cristian Riveros

# Outline

Unión de autómatas (clase anterior)

Autómatas no-deterministas

Relevancia del concepto

Comparación con DFA

# Outline

Unión de autómatas (clase anterior)

Autómatas no-deterministas

Relevancia del concepto

Comparación con DFA

# Complemento, intersección y unión de autómatas

## Definiciones

Dado dos lenguajes  $L, L' \subseteq \Sigma^*$  se define:

$$L^C = \{w \in \Sigma^* \mid w \notin L\}$$

$$L \cap L' = \{w \in \Sigma^* \mid w \in L \wedge w \in L'\}$$

$$L \cup L' = \{w \in \Sigma^* \mid w \in L \vee w \in L'\}$$

Dado dos autómatas  $\mathcal{A}$  y  $\mathcal{A}'$ :

1. ¿Existe un autómata  $\mathcal{B}$  tal que  $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{A})^C$ ? ✓
2. ¿Existe un autómata  $\mathcal{B}$  tal que  $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{A}')$ ? ✓
3. ¿Existe un autómata  $\mathcal{B}$  tal que  $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{A}) \cup \mathcal{L}(\mathcal{A}')$ ?

¿existe un autómata  $\mathcal{B}$  tal que  $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{A}) \cup \mathcal{L}(\mathcal{A}')$ ?

Sabemos que:

$$\mathcal{L}(\mathcal{A}) \cup \mathcal{L}(\mathcal{A}') = (\mathcal{L}(\mathcal{A})^c \cap \mathcal{L}(\mathcal{A}')^c)^c$$

Para calcular el autómata que acepta el lenguaje  $\mathcal{L}(\mathcal{A}) \cup \mathcal{L}(\mathcal{A}')$ :

1. Complementamos  $\mathcal{A}$  y  $\mathcal{A}'$ .
2. Intersectamos  $\mathcal{A}_\neg$  y  $\mathcal{A}'_\neg$ .
3. Complementamos  $\mathcal{A}_\neg \times \mathcal{A}'_\neg$ .

# Outline

Unión de autómatas (clase anterior)

**Autómatas no-deterministas**

Relevancia del concepto

Comparación con DFA

¿qué significa “no-determinismo”?

*“Indeterminism is the concept that events (certain events, or events of certain types) are not caused **deterministically** (cf. causality) by prior events. It is the opposite of determinism and related to chance. It is highly relevant to the philosophical problem of **free will**.”*

Wikipedia (2022)

# ¿por qué nuestros autómatas son deterministas?

## Definición

Un autómata finito **determinista** (DFA) es una estructura:

$$\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$$

- $Q$  es un conjunto finito de estados.
- $\Sigma$  es el alfabeto de input.
- $\delta : Q \times \Sigma \rightarrow Q$  es la **función de transición**.
- $q_0 \in Q$  es el estado inicial.
- $F \subseteq Q$  es el conjunto de estados finales (o aceptación).

¿qué sería un autómata finito **no-determinista**?



# Autómata finito no-determinista

## Definición

Un autómata finito **no-determinista** (NFA) es una estructura:

$$\mathcal{A} = (Q, \Sigma, \Delta, I, F)$$

- $Q$  es un conjunto finito de estados.
- $\Sigma$  es el alfabeto de input.
- $F \subseteq Q$  es el conjunto de estados finales (o aceptación).

+

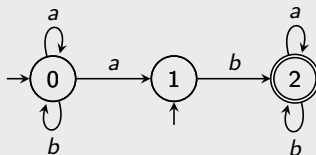
- $\Delta \subseteq Q \times \Sigma \times Q$  es la **relación de transición**.
- $I \subseteq Q$  es un **conjunto de estados iniciales**.

# Autómata finito no-determinista

## Ejemplo

- $Q = \{0, 1, 2\}$
- $\Sigma = \{a, b\}$
- $\Delta \subseteq Q \times \Sigma \times Q$  se define como:

$$\begin{aligned}\Delta = \{ & (0, a, 0), \\ & (0, a, 1), \\ & (0, b, 0), \\ & (1, b, 2), \\ & (2, a, 2), \\ & (2, b, 2) \}\end{aligned}$$



- $I = \{0, 1\}$
- $F = \{2\}$

# ¿cómo ejecuto un autómata no-determinista?

Sea:

- Un autómata finito no-determinista  $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ .
- El input  $w = a_1 a_2 \dots a_n \in \Sigma^*$ .

Una **ejecución** (o run)  $\rho$  de  $\mathcal{A}$  sobre  $w$  es una secuencia:

$$\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} p_n$$

- $p_0 \in I$
- para todo  $i \in \{0, \dots, n-1\}$ ,  $(p_i, a_{i+1}, p_{i+1}) \in \Delta$ .

Una ejecución  $\rho$  de  $\mathcal{A}$  sobre  $w$  es de **aceptación** si:

$$p_n \in F.$$

Desde ahora hablaremos de **las ejecuciones** de  $\mathcal{A}$  sobre  $w$

# Lenguaje aceptado por un autómata no-determinista

Sea un autómata  $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$  y  $w \in \Sigma^*$ .

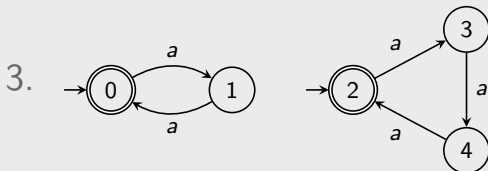
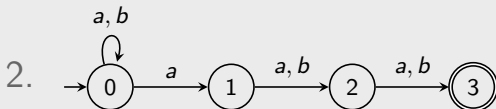
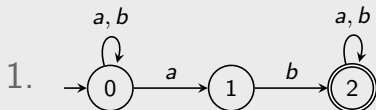
## Definiciones

- $\mathcal{A}$  **acepta**  $w$  si **existe** una ejecución de  $\mathcal{A}$  sobre  $w$  que es de aceptación.
- $\mathcal{A}$  **rechaza**  $w$  si **todas** las ejec. de  $\mathcal{A}$  sobre  $w$  **NO** son de aceptación.
- El **lenguaje aceptado** por  $\mathcal{A}$  se define como:

$$\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^* \mid \mathcal{A} \text{ acepta } w\}$$

# Lenguaje aceptado por un autómata no-determinista

¿qué lenguaje acepta cada autómata no-determinista?



# Interpretación de la relación de transición

Un autómata finito **no-determinista** (NFA) es una estructura:

$$\mathcal{A} = (Q, \Sigma, \Delta, I, F)$$

1.  $\Delta \subseteq Q \times \Sigma \times Q$  es la **relación de transición**.

*" $(q, a, p) \in \Delta$  entonces existe una transición desde **q** a **p** al leer **a**."*

2.  $I \subseteq Q$  es un **conjunto de estados iniciales**.

*" $p \in I$  entonces **p** es un posible estado inicial del autómata."*

# Interpretación de la relación de transición

Un autómata finito **no-determinista** (NFA) es una estructura:

$$\mathcal{A} = (Q, \Sigma, \Delta, I, F)$$

Otra definición equivalente:

1'.  $\Delta : Q \times \Sigma \rightarrow 2^Q$  es una **función de transición**.

*" $q \in \Delta(p, a)$  entonces*

*$q$  es un posible estado que puedo llegar desde  $p$  al leer  $a$ ."*

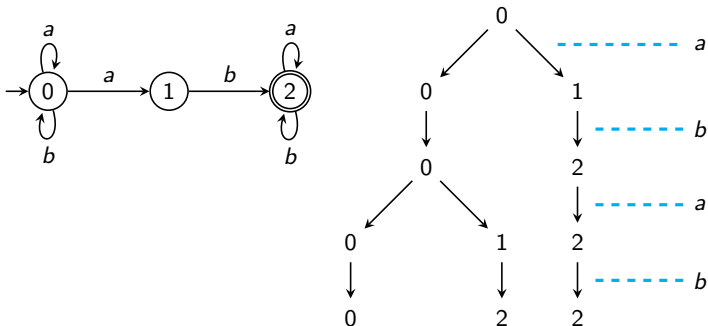
Esta última definición es la que posiblemente verán en libros

# Interpretación del no-determinismo

El **no-determinismo** puede ser visto como:

## 1. Paralelización infinita.

- Cada ejecución es un thread distinto.



## 2. "Guessing and Verifying" (adivinar y verificar).



# Interpretación del no-determinismo

El no-determinismo **NO debe ser visto** como:

- explícitamente como el **indeterminismo** o “libre albedrío”.
  - Para un input, un NFA siempre produce el mismo resultado.
- comportamiento **aleatorio** del autómata.

# Outline

Unión de autómatas (clase anterior)

Autómatas no-deterministas

**Relevancia del concepto**

Comparación con DFA

# ¿qué tan importante es el no-determinismo en ciencia de la computación?

Propuesto en el paper:

*"Finite Automata and Their Decision Problem" (1959)*



**Michael O. Rabin**

- Israel
- 92 años



**Dana Scott**

- EE UU
- 91 años

# ¿qué tan importante es el no-determinismo en ciencia de la computación?

Ambos ganadores del **Turing Award** (Novel en Computación) en 1979:

*"For their joint paper 'Finite Automata and Their Decision Problem' which introduced the idea of **nondeterministic machines**, which has proved to be an enormously valuable concept. Their (Scott and Rabin) classic paper has been a continuous source of inspiration for **subsequent work** in this field."*

ACM

¿cuál ha sido este legado o "subsequent work" en el área?

¿cuál es la pregunta abierta  
más importante en ciencia de la computación?

$$P \stackrel{?}{=} NP$$

*"If the solution to a problem is easy to check for correctness,  
must the problem be easy to solve?"*

Wikipedia (2024)

# ¿cuál es la pregunta abierta más importante en ciencia de la computación?

Uno de los 7 “Millennium Prize Problems”:

1. Yang–Mills and Mass Gap
2. Riemann Hypothesis
3. **P vs NP Problem**
4. Navier–Stokes Equation
5. Hodge Conjecture
6. ~~Poincaré Conjecture~~
7. Birch and Swinnerton-Dyer Conjecture

propuestos por *The Clay Mathematics Institute* (2000)

¿cuál es la pregunta abierta  
más importante en ciencia de la computación?

$$P \stackrel{?}{=} NP$$

US\$ 1 millón a quien lo resuelva.

*"Aside from being an important problem in computational theory, a proof either way would have **profound implications** for mathematics, cryptography, algorithm research, artificial intelligence, game theory, multimedia processing, philosophy, economics and many other fields. "*

Wikipedia (2024)

# Outline

Unión de autómatas (clase anterior)

Autómatas no-deterministas

Relevancia del concepto

**Comparación con DFA**



# ¿qué tan poderoso es el no-determinismo en autómatas?

$$\text{DFA} \subseteq \text{NFA}$$

Dado un autómata finito determinista  $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$  podemos construir un autómata finito no-determinista:

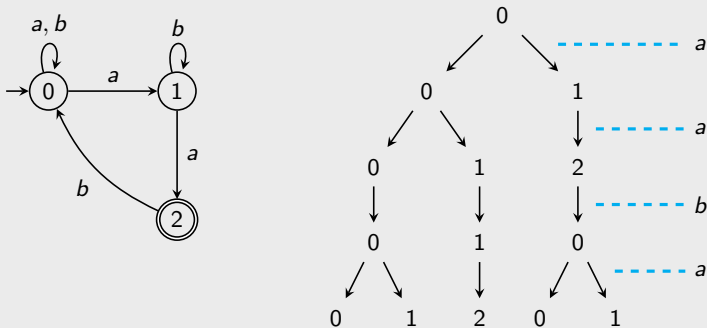
$$\mathcal{A}' = (Q, \Sigma, \Delta, I, F)$$

- $(p, a, q) \in \Delta$  si, y solo si,  $\delta(p, a) = q$ .
- $I = \{q_0\}$ .
- $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}')$ .

¿ DFA  $\not\subseteq$  NFA ?

## ¿qué tan poderoso es el no-determinismo en autómatas?

## Ejemplo



¿puede un autómata determinista almacenar **todas** las ejecuciones?

# ¿qué tan poderoso es el no-determinismo en autómatas?

## Teorema

Para todo autómata finito no-determinista  $\mathcal{A}$ , existe un autómata determinista  $\mathcal{A}'$  tal que:

$$\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}')$$

En otras palabras, DFA  $\equiv$  NFA.

Ambos modelos computan lo mismo

Veremos la demostración  
la próxima clase ...

# Cierre de clase

En esta clase vimos:

- Modelo de autómata no-determinista.
- Importancia del concepto de no-determinismo.
- Comparación con autómatas deterministas.

**Próxima clase:** determinización de autómatas y algoritmos.