# ACI650 - Modelos y Simulación

## Introducción a los Autómatas Celulares


Mario González

*Facultad de Ingeniería y Ciencias Ambientales*

Centro de Investigación, Estudios y Desarrollo de Ingeniería (CIEDI)


UNIVERSIDAD DE LAS AMÉRICAS
Laureate International Universities®

June 13, 2016

# Objetivos de Aprendizaje

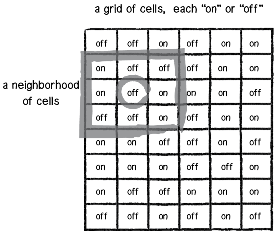
- Comprender los conceptos básicos y funcionamiento de los Autómatas Celulares.
- Identificar aplicaciones de los Autómatas Celulares.
- Implementar un Autómata Celular elemental en Python.

# Indice de la presentación

# Qué es un autómata celular (CA) I

Un autómata celular es un modelo de un sistema de objetos "celulares", con las siguientes características:

- Las células viven en una **rejilla** (de un número finito de dimensiones).
- Cada célula tiene un **estado**. El número de posibilidades de estados suele ser finito. El ejemplo más simple tiene las dos posibilidades de 1 y 0 (conocidos como on/off o vivo/muerto).
- Cada célula tiene un **vecindario**. Este se puede definir de muchas maneras, pero es típicamente una lista de células adyacentes.



# Qué es un autómata celular (CA) II

- El desarrollo de sistemas de autómatas celulares se suele atribuir a Stanisław Ulam y John von Neumann, investigadores en el Laboratorio Nacional de Los Álamos en Nuevo México en 1940.
- Ulam estaba estudiando el crecimiento de cristales y von Neumann estaba imaginando un mundo de robots auto-replicantes.
- Quizás la obra científica más significativa (y larga) acerca de autómatas celulares llegó en 2002, de Stephen Wolfram: A New Kind of Science.
- Wolfram remarca que los CA no son simples trucos, sino que son relevantes para el estudio de la biología, la química, la física y todas las ramas de la ciencia.

# Autómata celular elemental I

1. **Rejilla**. La rejilla más simple sería unidimensional: una línea de células.

2. **Estados**. El conjunto más simple de los estados (más allá de tener un solo estado) sería de dos estados: 0 o 1.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |

# Autómata celular elemental II

3. **Vecindario.** El vecindario más simple en una dimensión para cualquier célula dada sería la celda en sí y sus dos vecinos adyacentes: uno a la izquierda y una a la derecha.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |

Digamos que empezamos cada célula en un estado aleatorio y dos vecinos. ¿Qué hacemos con las células de los extremos?

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |

# Células de los extremos

Aquí hay tres soluciones posibles a este problema:

- Los bordes se mantienen constantes. Esta es quizás la solución más sencilla. Nunca nos molestamos en evaluar los bordes y siempre dejamos su valor de estado constante (0 ó 1).
- Envolver (cerrar) los bordes. Piense en la CA como una tira de papel y convertir esa tira de papel en un anillo. La celda en el borde izquierdo es un vecino de la celda de la derecha y viceversa. Esto puede crear la apariencia de una red infinita y es probablemente la solución más utilizada.
- Los bordes tienen diferentes vecindarios y reglas. Si quisiéramos, podríamos tratar las células del borde de otra manera y crear reglas para las células que tienen una vecindad de dos en vez de tres.

# Evolución de un autómata celular

Los CA viven durante un período de tiempo, que también se podría llamar una generación y, en nuestro caso, se refiere también a la cuenta de “frames” de una animación. Dado un CA en el tiempo 0 o generación 0:

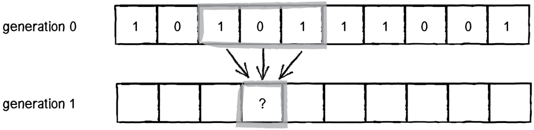
- ¿Cómo calculamos los estados de todas las células en la generación 1?
- Y la generación 2?
- Y así sucesivamente y así sucesivamente.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| generation 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| generation 1 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? |
| generation 2 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? |

# Evolución de un autómata celular

- Dada una celda individual CELL en el CA, el estado de CELL en cualquier momento dado t es:

$$\text{CELL}_i^t = f(\text{CELL}_j^{t-1}),\ j \in \text{vecindario}$$

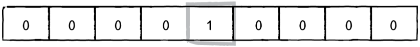



- Si tenemos 3 células cada una con estados 0 o 1. ¿Cuántas maneras posbiles hay de configurar los estados?

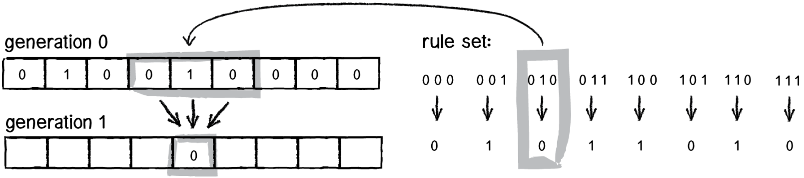
| 0 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1 0 0 | 1 0 1 | 1 1 0 | 1 1 1 |
|---|---|---|---|---|---|---|---|
| ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ | ↓ |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |

# Conjuto de reglas de evolución de un CA

- El modelo estándar de Wolfram empieza la generación 0 con todas las células con estado 0, a excepción de la celda del medio, que debe tener estado de 1.



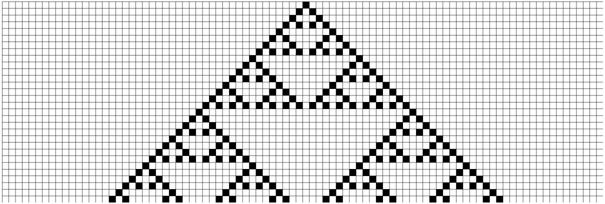

- Siguiendo el conjunto de reglas presentado anteriormente tenemos:



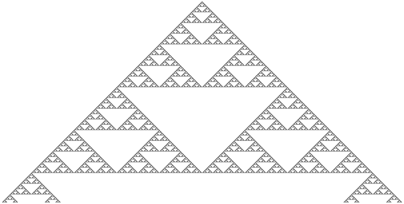

- Intenten aplicar el conjunto de reglas a las demás celdas.

# Generaciones de un CA

- Vamos más allá de sólo una generación (0 es blanco, 1 es negro), y apilamos, cada nueva generación aparece debajo de la anterior.

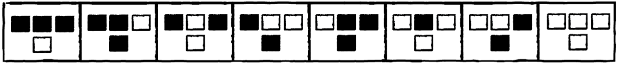


- La forma de arriba es el Triángulo de Sierpinski.



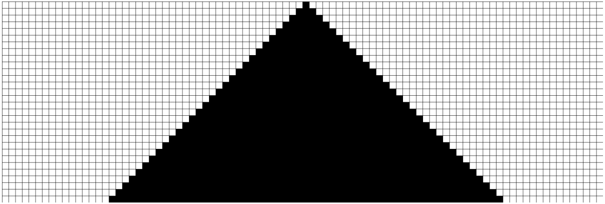
# Reglas de evolución

- Nuestra regla de evolución es una lista de 8-bits.
- En términos de una CA elemental de Wolfram, hay 256 posibles conjuntos de reglas.
- Regla 90: convierta la secuencia 01011010 a decimal y obtendrá 90.



0 1 0 1 1 0 1 0

- Regla 222: 11011110



# Clasificación de Wolfram I

- **Clase 1: Uniformidad.** Clase 1 CAs terminan, después de un cierto número de generaciones, con cada celda constante. La Regla 222 (abajo) es un CA clase 1.

- **Clase 2: Repetición.** Al igual que un CA clase 1, CAs clase 2 se mantienen estables, pero los estados celulares no son constantes. Más bien, oscilan en un patrón regular de 0 a 1 a 0 a 1 y así sucesivamente. Ejemplo la regla 190 (abajo).

# Clasificación de Wolfram II

- **Clase 3: Aleatoriedad.** Clase 3 CAs parecen aleatorios y no tienen un patrón fácilmente discernible. La regla 30 (abajo) se utiliza como un generador de números aleatorios en el software Mathematica de Wolfram.

- **Clase 4: Complejidad.** Clase 4 CAs pueden ser considerados como una mezcla de la clase 2 y clase 3. Uno puede encontrar, patrones oscilantes repetitivos dentro del CA, pero dónde y cuándo aparecen estos patrones es impredecible y aparentemente al azar. Clase 4 CAs exhiben propiedades de sistemas complejos. Ejemplo Regla 110 (abajo).

# Aplicaciones I

- **Criptografía.** Los autómatas celulares han sido propuestos como llaves públicas en criptografía. Por ejemplo, la función de evolución de una CA finito tiene una inversa que (se cree) es difícil de encontrar. Conociendo la regla, cualquiera puede calcular fácilmente los estados futuros, pero parece ser muy difícil calcular los estados anteriores.
- **Procesamiento de Imágenes.** Many image-processing algorithms operate on CA-like rules. Blurring an image is creating a new pixel out of the average of a neighborhood of pixels.
  - Simulación de tinta dispersandose en papel o la ondulación del agua sobre una imagen.

# Aplicaciones II

- **CA para el modelado físico y sistemas biológico.** Los autómatas celulares son una alternativa a las ecuaciones diferenciales en el modelado de las leyes de la física. Modelos de CA se usan para sistemas físicos con énfasis en sistemas de espines, modelos para diversas formas de regular el crecimiento en dendritas y crecimiento aleatorio están basados en la CA de dos dimensiones, modelos para la formación de patrones en los sistemas de reacción difusión. En qúimica el proceso de solidificación con el análisis especial de la transformación de fase de la sustancia de líquido a sólido, la formación de aleaciones, etc.
- **Bioinformática.** CA para la predicción de codificación de proteínas.
- **Otros.** Modelos de flujo de tráfico microscópico. Flujo de tráfico en una sola línea o autopista. Aniquilación balística por el cual partículas y antipartículas en movimiento se aniquilan entre sí cuando chocan (Rule 184).

# Implementación

Implemente un Autómata Celular Elemental en Python.

# Fuentes y Recursos

- Daniel Shiffman, The Nature of Code. Chapter 7. Cellular Automata.
- Golly application. Conway's Game of Life and other cellular automata.
- Leandro Nunes de Castro. Fundamentals of natural computing: an overview.
- Stephen Wolfram's New Kind of Science.