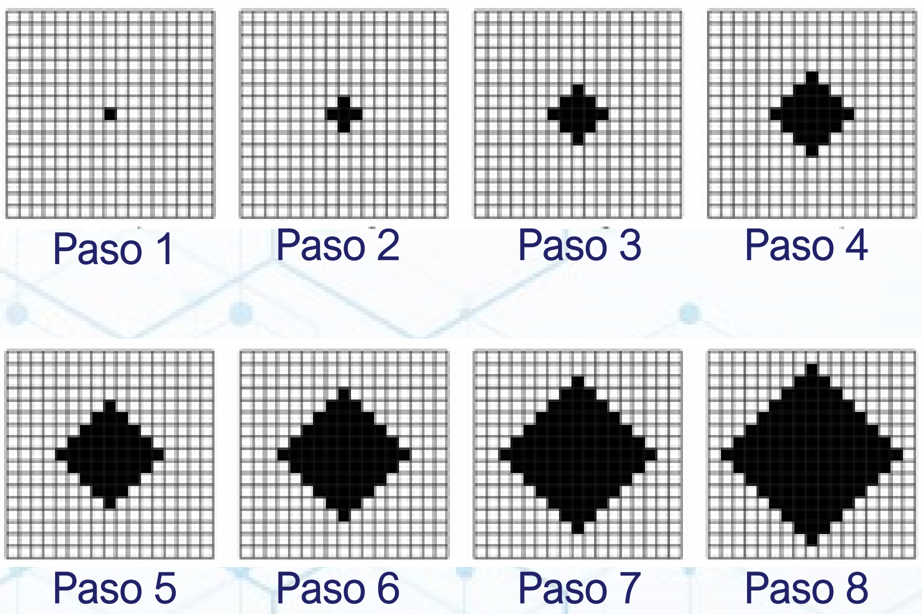
Algoritmos Bioinspirados

Unidad IV : Autómatas celulares

Programa: Ingeniería en Inteligencia Artificial

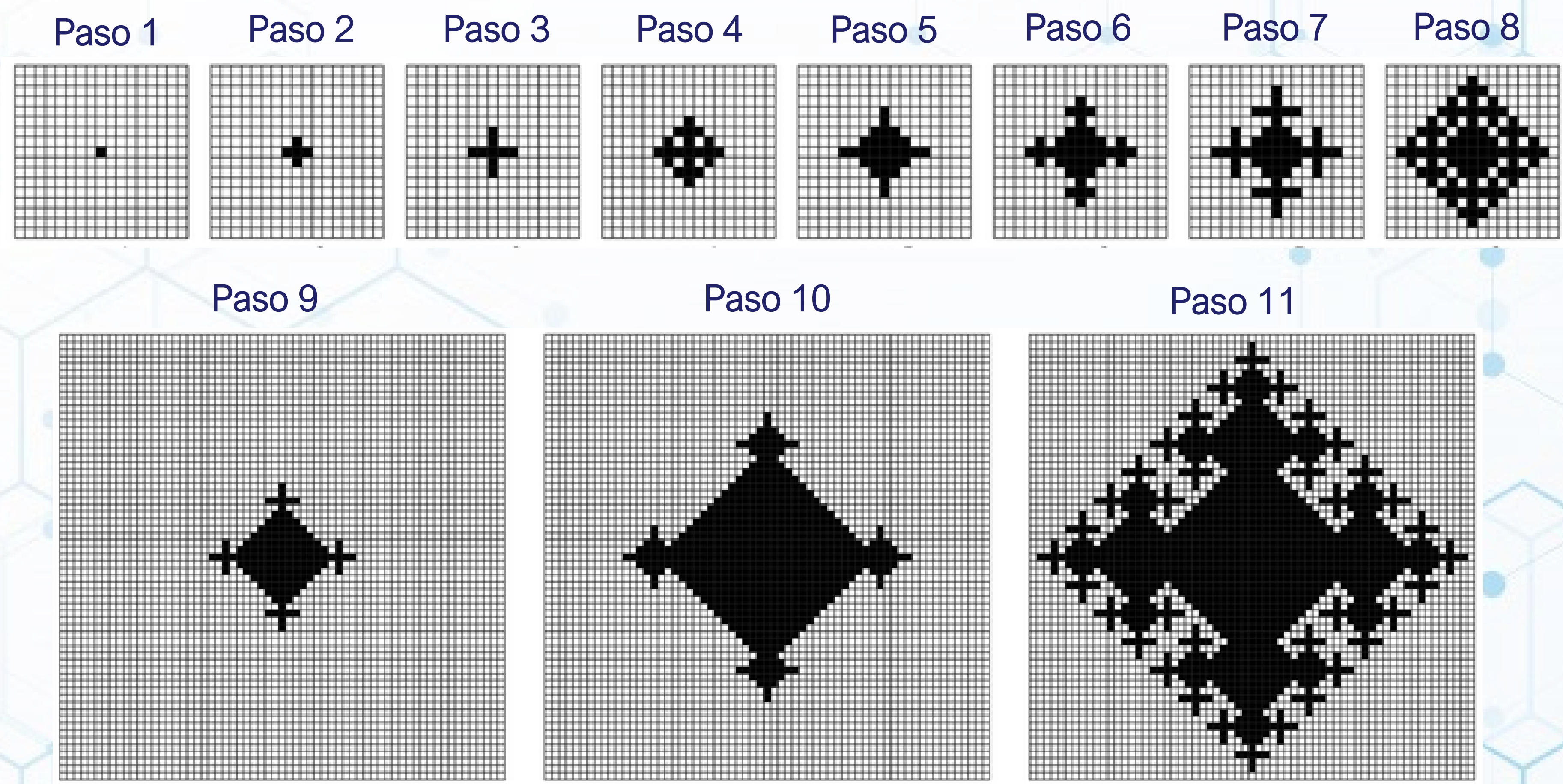
Definición (informal)

Un autómata celular (AC) es una colección de celdas *coloreadas* en una cuadrícula de forma específica que evoluciona a través de una serie de pasos de tiempo discretos de acuerdo con un conjunto de reglas basadas en los estados de las celdas vecinas. Luego, las reglas se aplican iterativamente durante tantos pasos de tiempo como se desee.



La imagen muestra que pasa al aplicar una simple regla, que indica que una célula (celda) debe volverse negra si cualquiera de sus 4 vecinos es negra en el paso previo.

La imagen muestra que pasa al aplicar una regla que indica que una célula (celda) debe volverse negra si 1 o todas sus 4 vecinas son negra.



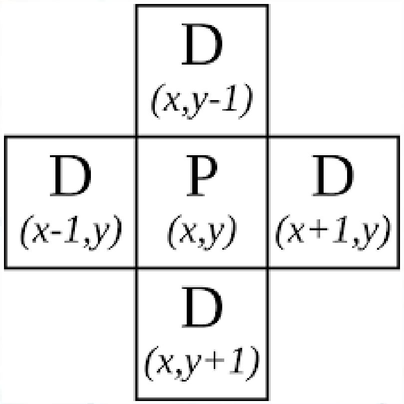
John Von Neumann

Stanislaw Ulam

El origen de los autómatas celulares se asocia a los intentos de John Von Neumann de crear máquinas que se autorrepliquen (*The general and logical theory of automa*, 1948”).

Theory of Self-Reproducing Automata, 1966, completado por Arthur W. Burks posterior a la muerte de Von Neumann.

El modelo propuesto fue una máquina de Turing embebida en una lattice celular 2-dimensional con 29 estados para cada célula y una vecindad de 5 células que requería aproximadamente 200,000 células. **Nunca fue implementada.**



Vecindario Von Neumann

En el autómata celular de von Neumann, las máquinas de estado finito (o celdas) están dispuestas en una cuadrícula cartesiana bidimensional e interactúan con las cuatro celdas circundantes.

Como el autómata celular de Von Neumann fue el primer ejemplo en utilizar este arreglo, se lo conoce como el vecindario de Von Neumann (*Von Neumann neighbourhood*)

En organismos multicelulares, casi todas las células contienen el mismo material genético y, sin embargo, la morfología y la función de dos células pueden ser sorprendentemente diferentes

Esta diferencia se explica por el hecho de que el estado de cada célula depende no sólo de su material genético sino también del estado de la célula cuando se generó y de las influencias que actuaron sobre la célula a partir de ese momento.

Un sistema multicelular es un ejemplo de un sistema que se compone de muchas copias de una unidad fundamental (célula) cuya interacción produce un comportamiento global que no es simplemente una versión ampliada del comportamiento de una unidad aislada.

Definición

Un autómata celular (AC) es un sistema dinámico discreto síncrono que consta de una matriz infinita de células capaz de simular procesos complejos [1].

- Cada celda tiene un estado de un conjunto de estados finitos.
- Las celdas cambian sus estados de acuerdo con reglas de actualización en función de los estados anteriores de la celda y sus vecinos (interacción local).
- Todos los estados usan la misma regla de actualización y la actualización ocurre simultáneamente en todas las celdas (homogéneos).
- La actualización se repite una y otra vez en pasos de tiempo discretos, lo que conduce a una evolución temporal del sistema

- Se usa la inspiración proporcionada por los tejidos celulares biológicos para definir los elementos de un sistema celular abstracto.
- El complejo estado interno de una célula biológica se reduce a variables de estado que toman valores en un conjunto de estado razonablemente simple.
- Las complejas interacciones que rigen la dinámica de las células biológicas se abstraen mediante una función o regla matemática que especifica cómo debe actualizarse la variable de estado, teniendo en cuenta las interacciones de una célula con sus vecinas, a partir de una configuración inicial del espacio celular.

4 elementos fundamentales componen un AC:

- Malla
- Estados
- Vecindario
- Funciones transición

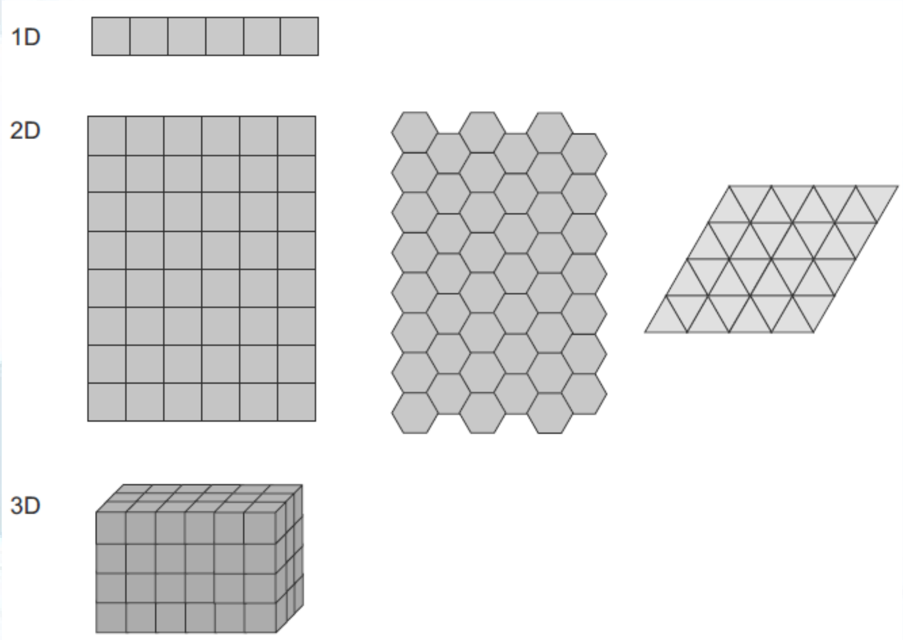
Malla

La colección de células en el sistema se llama el espacio celular. En general, es una malla (retícula, *lattice*) regular de celdas d-dimensional.

- A un nivel abstracto, la malla suele considerarse infinita, en la práctica cualquier implementación real trata con un espacio finito.
- Cada celda, llamada en la literatura célula, puede ser representada como un autómata de estado finitos (AF).
- Un espacio celular d-dimensional se representa por $\mathbb{Z}^d$. Los elementos de $\mathbb{Z}^d$ son las células.

- La mayoría de las aplicaciones se basan en una rejilla cuadrada, pero también se han utilizado rejillas triangulares y hexagonales e incluso rejillas irregulares.

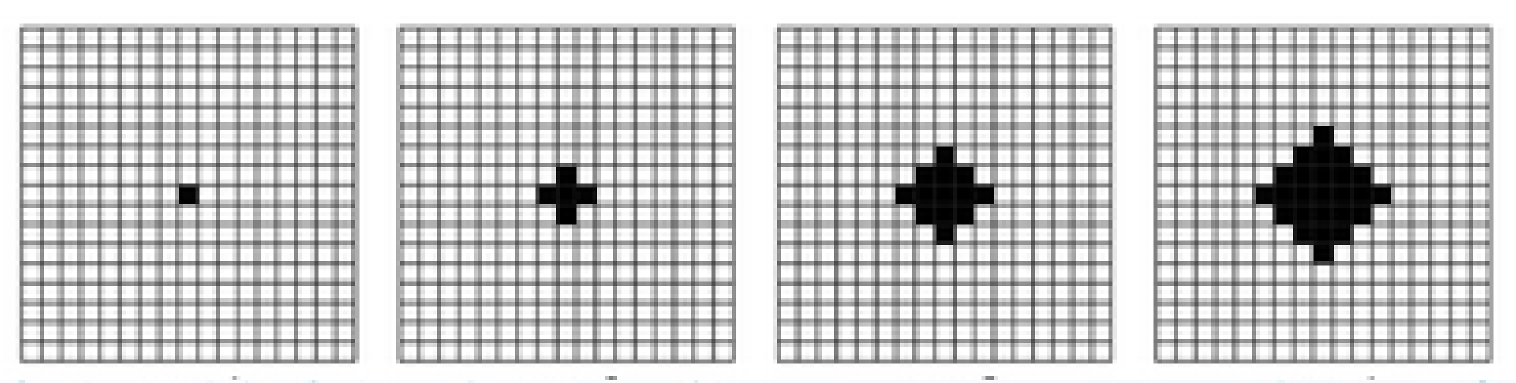
- Los espacios celulares de más de tres dimensiones rara vez se consideran en implementaciones reales porque el número total de celdas a lo largo de cada dimensión crece exponencialmente con el número de dimensiones.



Variable de tiempo: Las células cambian sus estados sincrónicamente en pasos de tiempo discretos

Los cambios en los estados de las células se realizan pasos de tiempo $t = 0,1,2,\ldots$

El paso $t = 0$ se reserva para la configuración inicial del sistema.



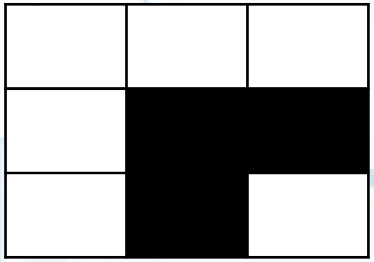
Paso 1 Paso 2 Paso 3 Paso 4

Estados

Es un espacio local de valores que define todos los posibles estados de cada célula (AF).

- Cada célula puede existir en k diferentes estado (de forma no simultanea) donde k es un valor entero igual o mayor que 2.
- Uno de estos estados tiene un estatus especial llamado **estado inactivo** (*quiescent state*)

0	0	0
0	1	1
0	1	0



El estado mas simple (2 estados) podría se denotado por 0 (estado inactivo) y 1, de forma gráfica se podrían representar por blanco y negro.

Estado 0 (blanco/inactiva) → **muerta**

Estado 1 (negro) → **viva**

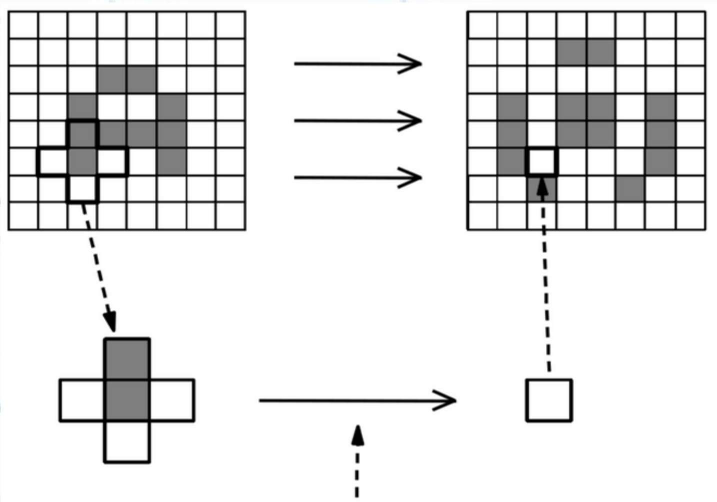
El estado de cada célula en un tiempo $t+1$ se encuentra bien definido por su propio estado y el de sus vecinos en un tiempo t.

El estado de cada célula es un estado local, mientras que el estado de todas las células en la malla es el estado global o configuración.

Una configuración debe entenderse como una descripción instantánea (*snapshot*) de todos los estados de las células del sistema algún momento del tiempo.

Estado en un tiempo t Estado en un tiempo $t+1$

Transición



Una configuración de un AC d-dimensional con un conjunto de estado S es una función:

$$c: \mathbb{Z}^d \rightarrow S$$

Vecindad

La vecindad de una celda es el conjunto de celdas (incluida la celda misma) cuyo estado puede influir directamente en el estado futuro de la celda.

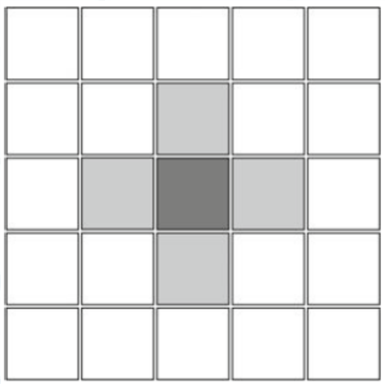
x_0 es la célula que va a ser actualizada.

$\mathcal{N} = (x_1, x_2, \ldots, x_m)$ donde $\mathcal{N} = (|x_i - x_0|) \leq r \;\; \forall\, i = 1,2, \ldots, m$

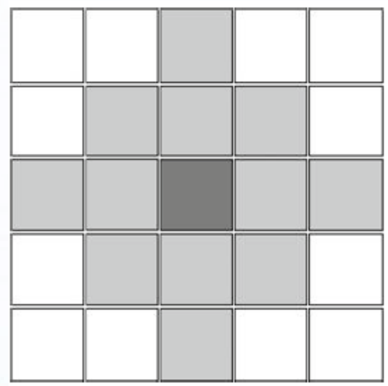
r es llamado el radio de la vecindad $\mathcal{N}$.

La forma y el tamaño del vecindario pueden ser arbitrarios. Sin embargo, la vecindad normalmente se compone de un pequeño número de celdas adyacentes.

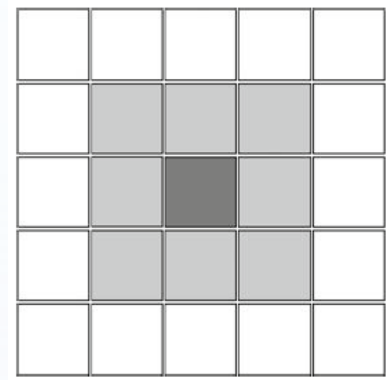
Típicamente se usan vecindarios homogéneos, es decir que todas las células del sistema tienen el mismo tipo de vecindad,



Vecinda Von Neumann
$(\boldsymbol{r = 1})$



Vecinda Von Neumann
$(\boldsymbol{r = 2})$



Vecinda de Moore

$$\mathcal{N}_N^1(i,j) = \{x_{k,l} \mid |i-k| + |j-l| \leq 1\} = (x_{i,j}, x_{i-1,j}, x_{i,j-1}, x_{i+1,j}, x_{i,j+1})$$

$$\mathcal{N}_M^1(i,j) = \{x_{k,l} \mid |i-k| \leq 1, |j-l| \leq 1\} = (x_{i,j}, x_{i-1,j}, x_{i-1,j-1}, x_{i-1,j+1}, x_{i,j-1}, x_{i,j+1}, x_{i+1,j}. x_{i+1,j-1}. x_{i+1,j+1})$$

Función o Regla de Transición

Describe el cambio de cada celda actualizada de su valor de estado actual a uno nuevo, operando sobre la vecindad.

$$s_i(t+1) = \varphi\big(s_j(t) \,\big|\, j \in \mathcal{N}_i\big)$$

En los autómatas celulares, todas las celdas usan la misma regla y la regla se aplica en todas las celdas simultáneamente, esto causa un cambio global en la configuración. El cambio de la configuración c a la configuración c'se le llama función de transición global.

Tipícamente la función G es aplicada repetidas veces lo que produce una evolución en el tiempo.

$$c \mapsto G(c) \mapsto G^2(c) \mapsto G^3(c) \mapsto \ldots$$

Esta secuencia de configuraciones es conocida como orbita

$$orb(c) = c, G(c), G^2(c), G^3(c), \ldots$$

El tiempo se refiere al número de aplicaciones de G realizadas. Cada aplicación de G toma un paso de tiempo, por lo que $G^t(c)$ es la configuración en el tiempo t, para todo $t = 0,1,2, \ldots$

En resumen: para definir un AC se deben especificar

i. La dimensión $d \in \mathbb{Z}^+$.

ii. El conjunto de estados finitos S.

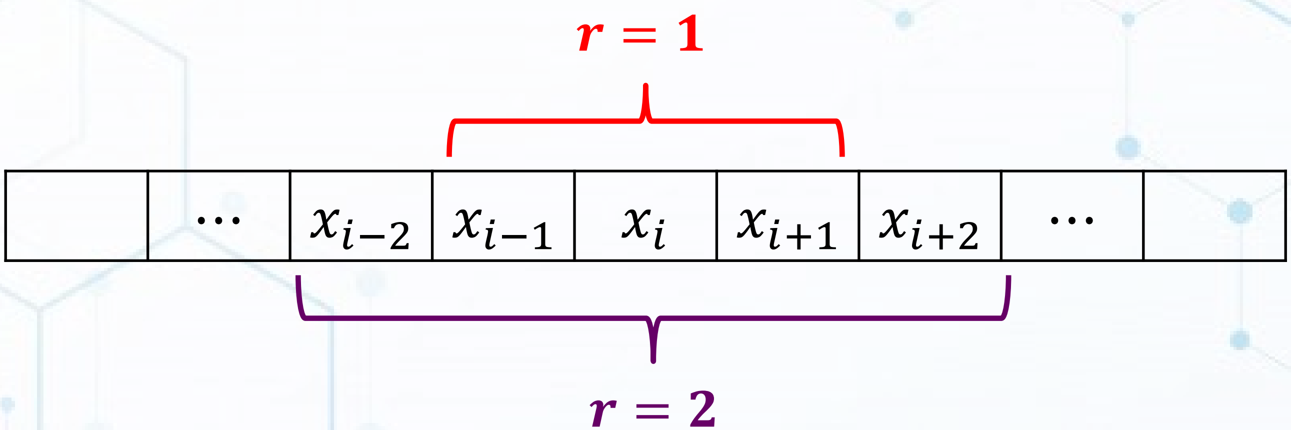
iii. El vector de vecindad $\mathcal{N} = (x_1, x_2, \ldots, x_m)$

iv. La función de transición $s_i(t+1) = \varphi\big(s_j(t) \,\big|\, j \in \mathcal{N}_i\big)$

Por tanto se puede definir un AC como una 4-tupla $CA = (d, S, \mathcal{N}, \varphi)$

Malla

Para AC unidimensionales la malla es una fila formada por celdas adyacentes.



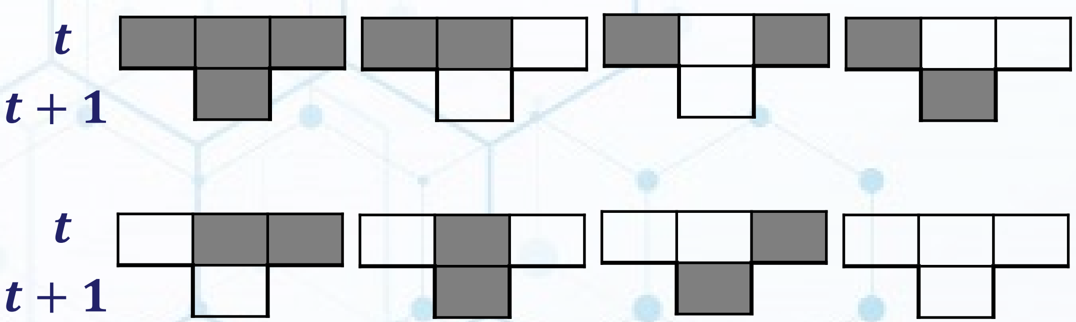
La forma de este espacio celular unidimensional deriva en que se define el vecindario con radio r como r células a la derecha de la celda a actualizar y r células a su izquierda. La vecindad contiene $2r + 1$ células.

La función o reglas de transición pueden ser expresadas de varias maneras:

Notación de función

$$f(111)=0,\quad f(110)=1,\quad f(101)=1,\quad f(100)=0$$
$$f(011)=1,\quad f(010)=1,\quad f(001)=1,\quad f(000)=0$$

Diagrama

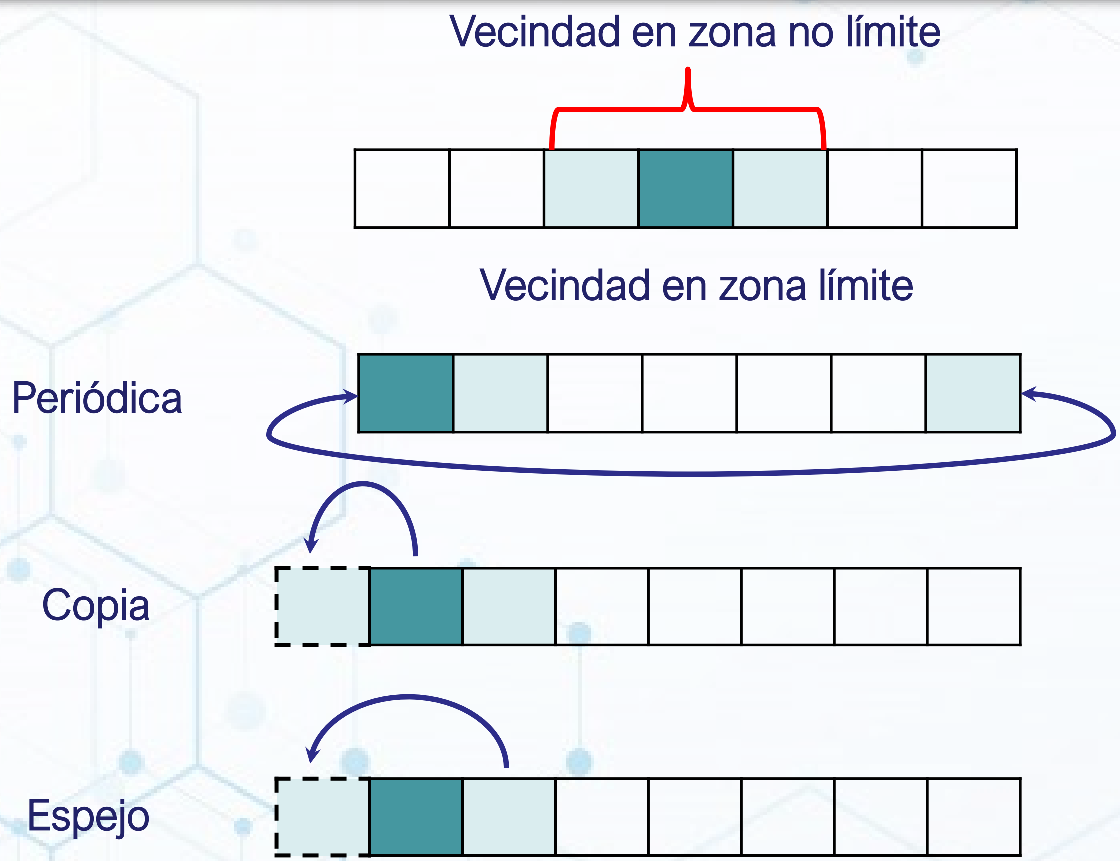


Tabla

$c_i(t+1)$	$c_{i-1}(t)$	$c_{i1}(t)$	$c_{i+1}(t)$
1	1	1	1
0	1	1	0
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	0
1	0	0	1
0	0	0	0

Condiciones de frontera

Si el espacio celular tiene un límite, las celdas en el límite pueden carecer de algunas de las celdas requeridas para formar la vecindad definida.



Desde inicio de los años 80 el investigador **Stephen Wolfram** ha realizado numerosos estudios sobre el comportamiento de los AC, con base principalmente en los AC unidimensionales con 2 o 3 estados.

El AC mas sencillo $r = 1$, $k = 2$, $S = \{0,1\}$ y $\mathcal{N} = \{x_{i-1}, x_i, x_{i-2}\}$, denominado por Wolfram como **AC elemental**. Para este, existen $2^3 = 8$ posibles combinaciones en el vecindario de una celda, que se pueden mapear a una salida de 0 o 1, por lo tanto, existen $2^8 = 256$ posibles reglas de transición

La formula general para el posible número de reglas que se pueden aplicar a un AC es k^{k^n} donde k es el número de estados y n es el tamaño de la vecindad.

Ejemplos de número de reglas para un número de estados y vecinos dados

# estados	# vecinos	k^{k^n}	# reglas
2	2	2^{2^2}	16
2	3	2^{2^3}	256
2	5	2^{2^5}	4 294 967 296
2	10	$2^{2^{10}}$	$1.797 \cdot 10^{308}$
5	2	5^{5^2}	$2.98 \cdot 10^{17}$
5	3	5^{5^3}	$2.35 \cdot 10^{87}$
5	5	5^{5^5}	$1.91 \cdot 10^{2184}$
10	2	10^{10^2}	10^{100}
10	3	10^{10^3}	10^{1000}
10	5	10^{10^5}	10^{100000}

Las reglas de los AC elementales han sido ampliamente estudiadas por Wolfram proponiendo un esquema de nombres que se ha transformado en un estándar.

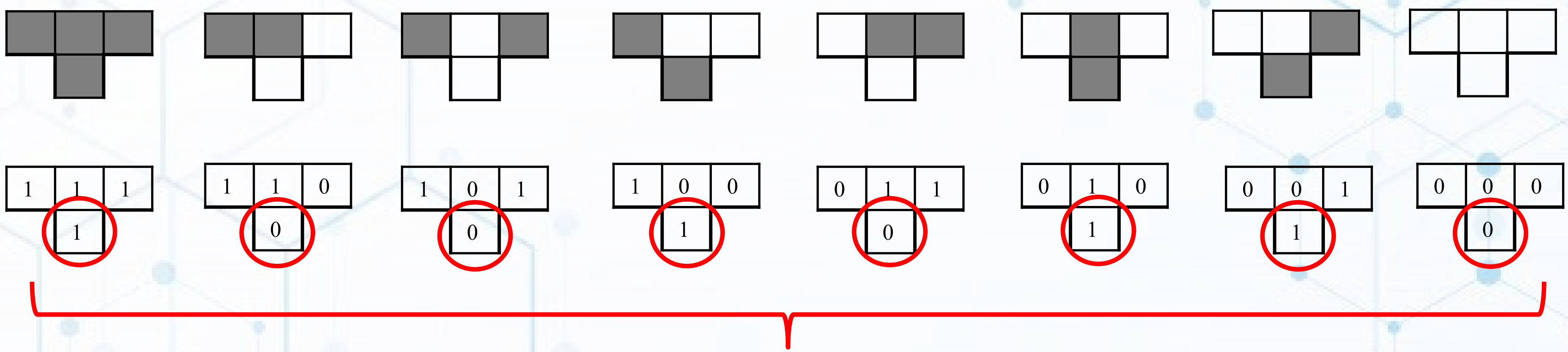
Cada regla elemental es especificada por una secuencia de 8 bits

$$f(111)\ f(110)\ f(101)\ f(100)\ f(011)\ f(010)\ f(001)\ f(000)$$

donde f es la regla local de actualización del CA. Esta secuencia binaria es la representación de un entero en el intervalo entre 0 – 255, llamado el número Wolfram for AC.

En este sistema de nombres cada regla se identifica por el número entero correspondiente a la cadena binaria.

$f(111) = 1$ $f(110) = 0$ $f(101) = 0$ $f(100) = 1$ $f(011) = 0$ $f(010) = 1$ $f(001) = 1$ $f(000) = 0$



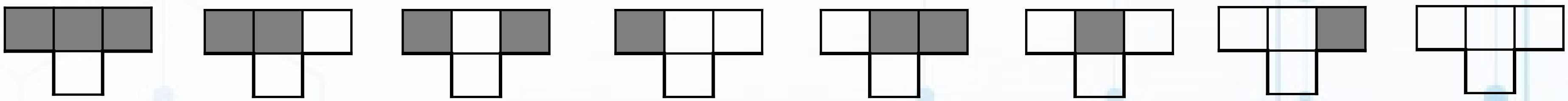
$$10010110 = 150$$

Regla 150

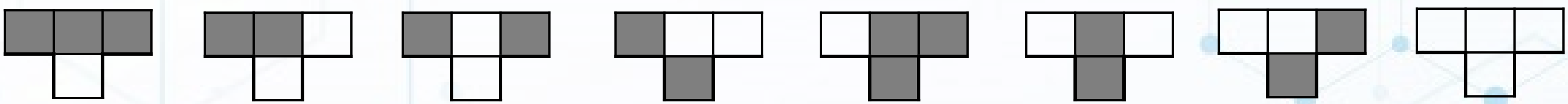
El esquema de numeración de Wolfram puede generalizarse fácilmente a conjuntos de estados y vecindarios más grandes.

Ejemplos:

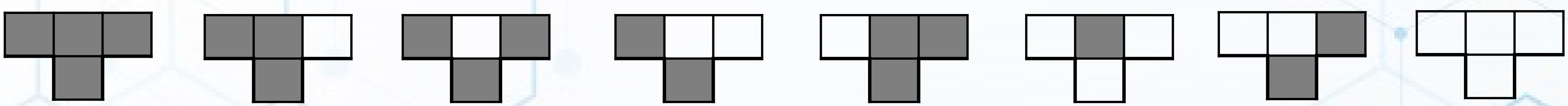
Regla ¿? 0



Regla ¿? 30



Regla ¿? 250

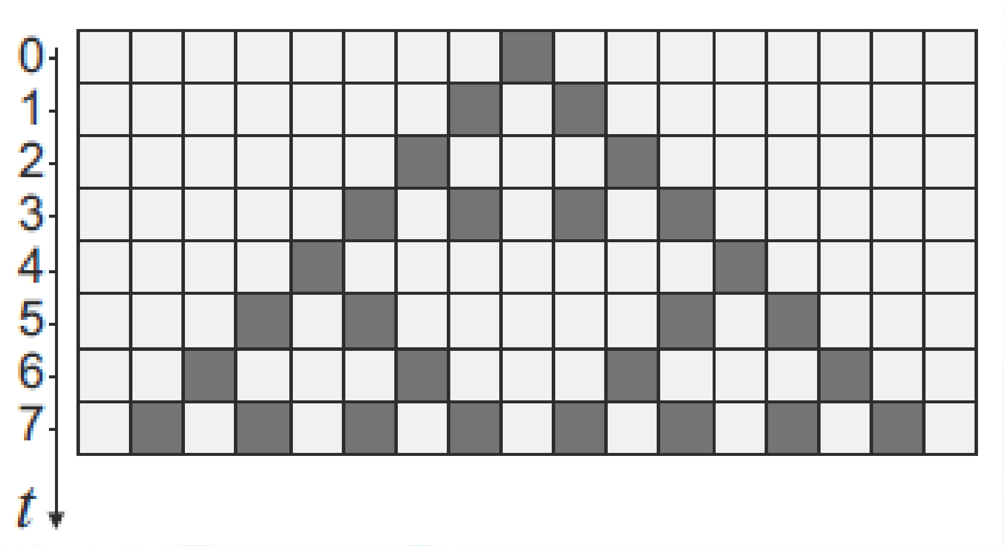


Con solo aumentar el radio de vecindario $r = 2$, entonces $n = 5$, por tanto $2^{2^5} = 4{,}294{,}967{,}296$ posibles reglas.

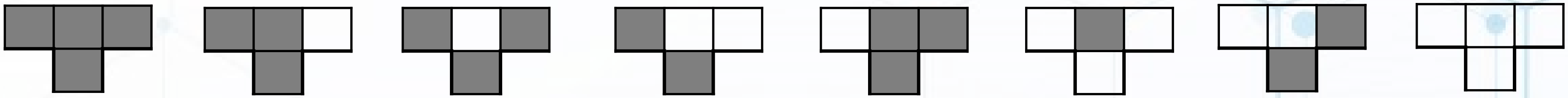
Y si aumentamos solo 1 estado $3^{3^3} = 7{,}625{,}597{,}484{,}987$ posibles reglas.

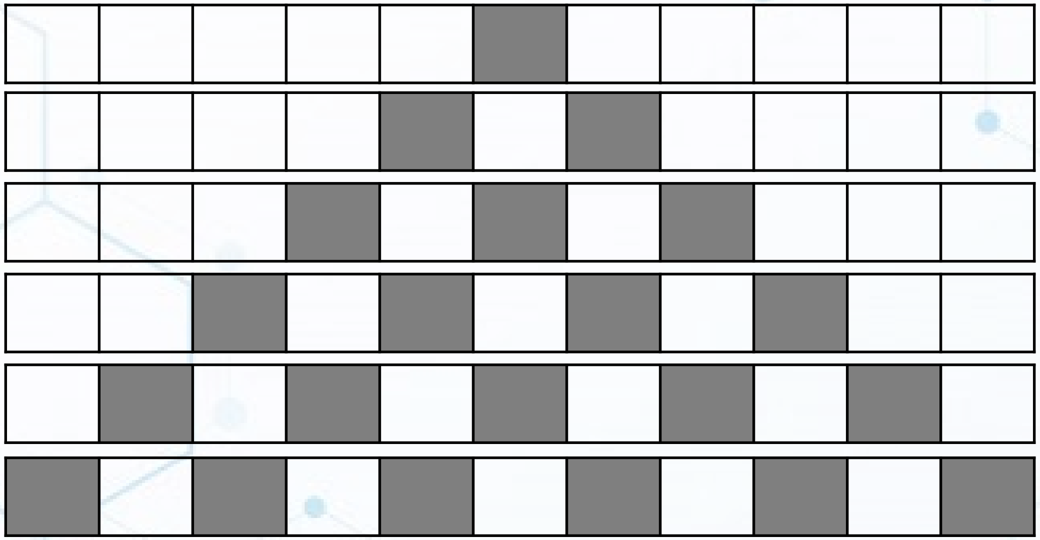
Diagrama espacio-temporales

El espacio celular en cada paso de tiempo se representa como una línea horizontal de cuadrados y la dirección vertical se utiliza para mostrar el despliegue en el tiempo de la configuración de estados del espacio celular.

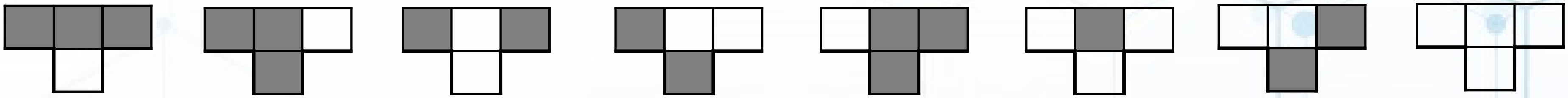


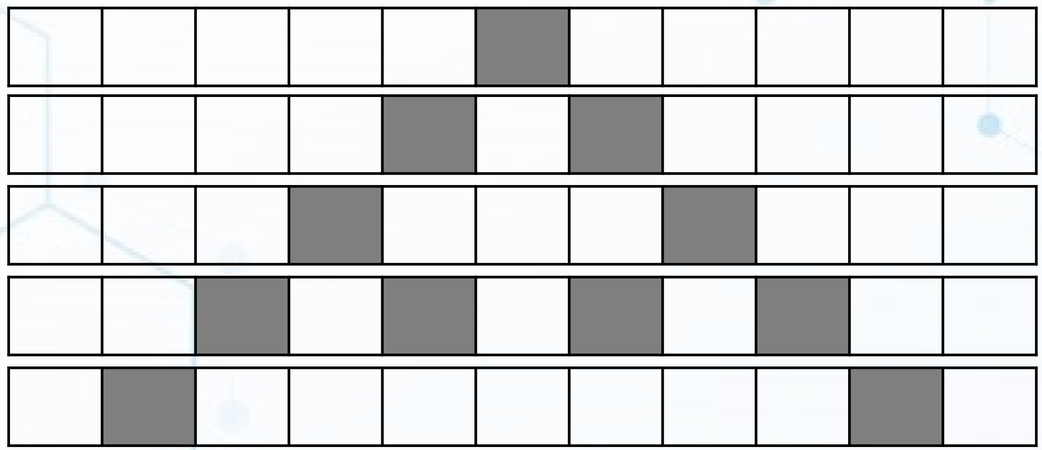
Ejemplo: generación parcial del diagrama espacio-temporal de la regla 250





Ejemplo: generación parcial del diagrama espacio-temporal de la regla 90

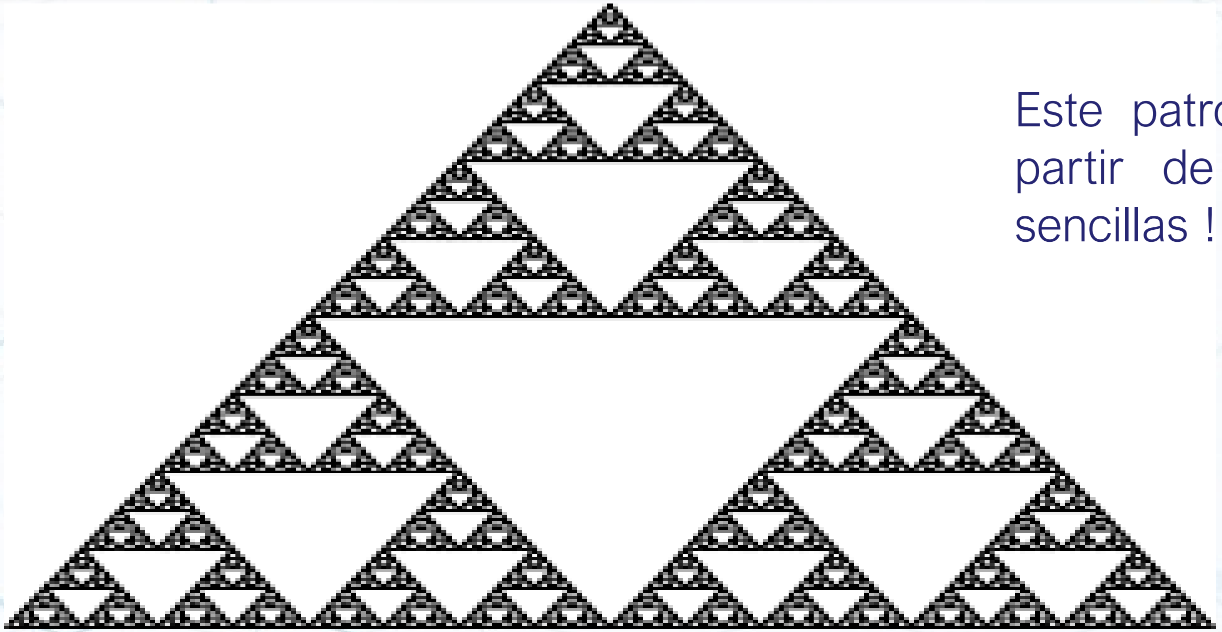




Cada vez que tenemos el patrón de cuadros intercalados, la región central se limpia.
A largo plazo esto tiene un efecto interesante.

Triángulo de Sierpinski

- Es un fractal que se puede construir a partir de cualquier triangulo.
- El patrón es autosimilar en diferentes escalas.
- Los triángulos más pequeños muestran la misma estructura que el triángulo principal.



Este patrón emerge a partir de reglas muy sencillas !!!

AC totalista

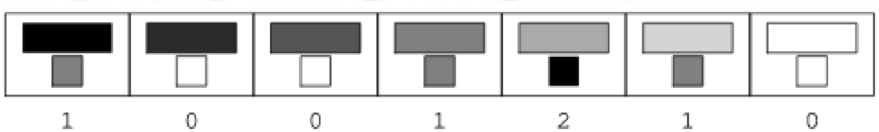
Dado que el universo de reglas de transición es en general tan amplio, es útil señalar algunas reglas que cumplen con algunas restricciones adicionales que las hacen más sencillas para asegurar que el AC posea alguna propiedad especial.

Regla totalista: suponiendo que los estados se representan como números, una regla CA se llama totalista si depende solo de la suma de los valores de los estados en la vecindad. Estos autómatas fueron presentados por Wolfram en 1983.

$$s_i(t+1) = \varphi\Big(\sum_{j\in N_i} s_j(t)\Big)$$

Para un AC totalista con k estados y n vecinos se tiene $n(k-2)$ posibles estados para la vecindad de n celdas y por tanto $k^{n(k-2)}$ posibles AC totalistas.

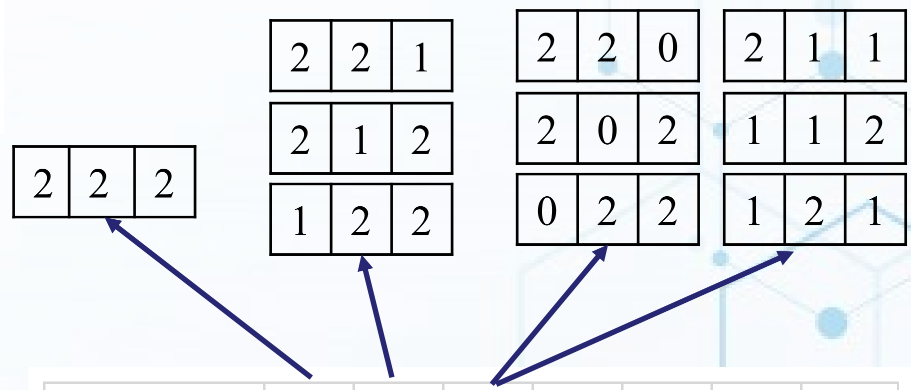
Cada uno de estos posibles AC se identifica con un número en base k denominado código (*code*). **Ejemplo:** automata 3 colores, radio 1, code 777.



$$1001210_3$$

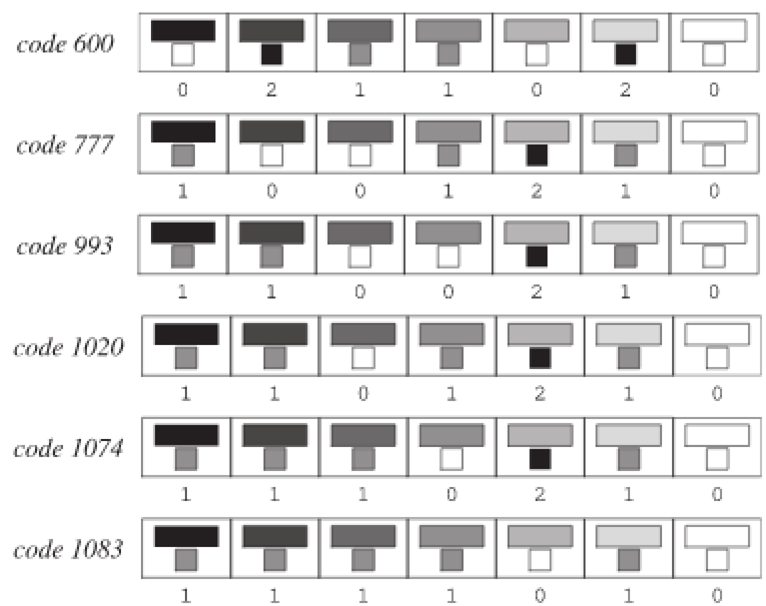
$$(1 \times 3^6) + (0 \times 3^5) + (0 \times 3^4) + (1 \times 3^3) + (2 \times 3^2) + (1 \times 3^1) + (0 \times 3^0)$$

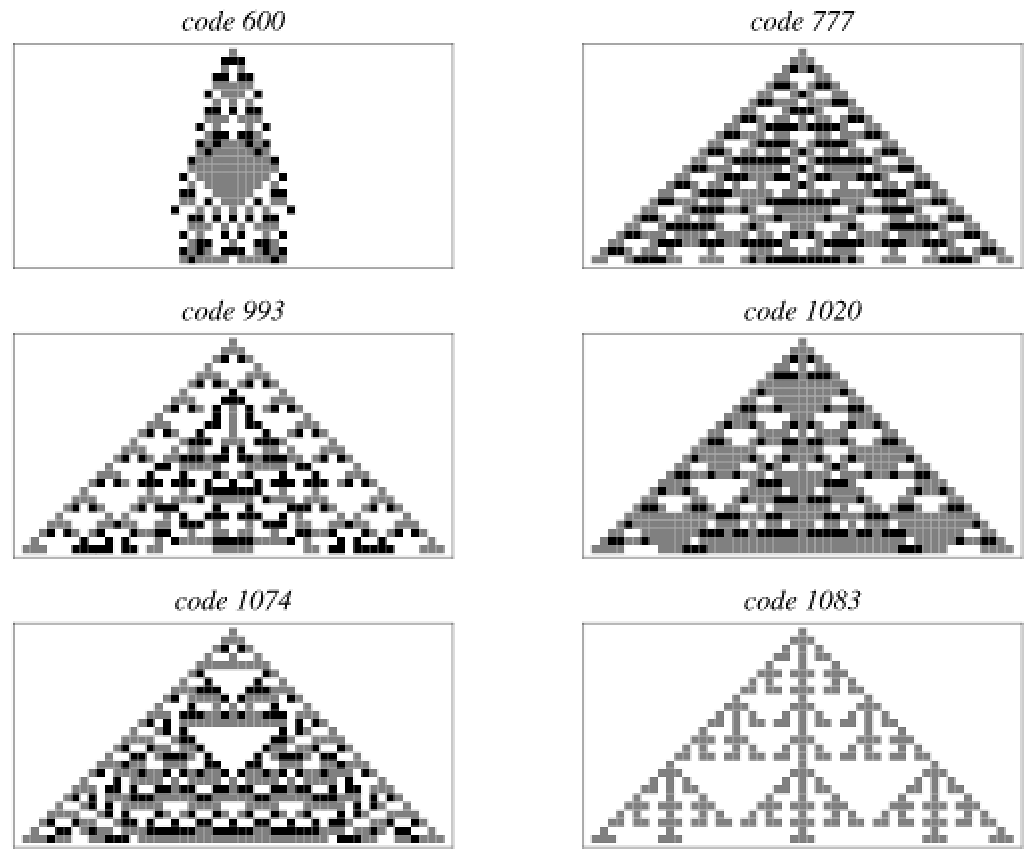
$$729 + 27 + 18 + 3 = 777$$



total	6	5	4	3	2	1	0
new value	1	0	0	1	2	1	0

Otros ejemplos:

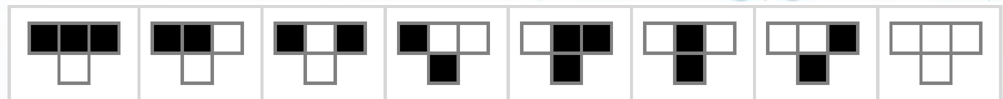




El conjugado de una regla f se define:

$$f_c(x,y,z) = 1 - f(1-x, 1-y, 1-z)$$

Ejemplo: conjugado de la regla 30



$$f_c(0,0,0) = 1 - f(1-0,1-0,1-0) = 1 - f(1,1,1) = 1 - 0 = 1$$
$$f_c(0,0,1) = 1 - f(1-0,1-0,1-1) = 1 - f(1,1,0) = 1 - 0 = 1$$
$$f_c(0,1,0) = 1 - f(1-0,1-1,1-0) = 1 - f(1,0,1) = 1 - 0 = 1$$
$$f_c(0,1,1) = 1 - f(1-0,1-1,1-1) = 1 - f(1,0,0) = 1 - 1 = 0$$
$$f_c(1,0,0) = 1 - f(1-1,1-0,1-0) = 1 - f(0,1,1) = 1 - 1 = 0$$
$$f_c(1,0,1) = 1 - f(1-1,1-0,1-1) = 1 - f(0,1,0) = 1 - 1 = 0$$
$$f_c(1,1,0) = 1 - f(1-1,1-1,1-0) = 1 - f(0,0,1) = 1 - 1 = 0$$
$$f_c(1,1,1) = 1 - f(1-1,1-1,1-1) = 1 - f(0,0,0) = 1 - 0 = 1$$

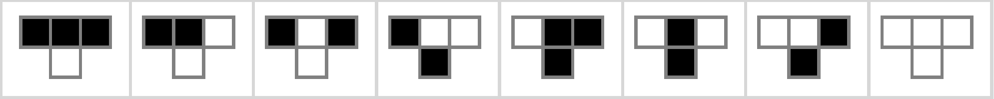
10000111

Regla 135

La reflexión de una regla f se define:

$$f_r(x,y,z) = f(z,y,x)$$

Ejemplo: conjugado de la regla 30



$$f_c(0,0,0) = f(0,0,0) = 0$$
$$f_c(0,0,1) = f(1,0,0) = 1$$
$$f_c(0,1,0) = f(0,1,0) = 1$$
$$f_c(0,1,1) = f(1,1,0) = 0$$
$$f_c(1,0,0) = f(0,0,1) = 1$$
$$f_c(1,0,1) = f(1,0,1) = 0$$
$$f_c(1,1,0) = f(0,1,1) = 1$$
$$f_c(1,1,1) = f(1,1,1) = 0$$

01010110

Regla 86

Las reglas que son invariantes ante la reflexión son llamadas simétricas. Ejemplo: regla 90.

Se dice que 2 AC son equivalentes si uno puede ser obtenido a partir del otro mediante la conjugación, reflexión o aplicando ambas de forma consecutiva.

Esto divide el espacio de las reglas en clases equivalentes. Escoger un solo representante por cada clase (la regla de menor número), se obtienen los autómatas celulares elementales esencialmente diferentes.

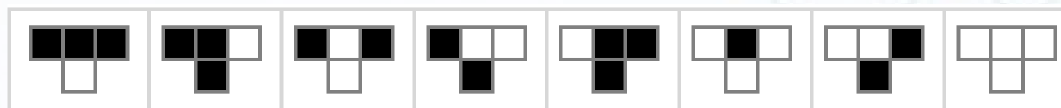
Teorema: existen 88 autómatas celulares elementales esencialmente diferentes.

4. Li, W. & Packard, N. (1990). The structure of the elementary cellular automata rule space. Complex Systems, vol. 4, pp. 281-297

Propiedad aditivita

Un autómata celular aditivo es un autómata celular cuya regla es compatible con una suma de estados. Es decir, que el estado de una célula en un tiempo $t+1$ depende de la suma de los estados de sus vecinas en el tiempo t.

Ejemplo: regla 90



$$c_i^{t+1} = \left(c_{i-1}^t + c_{i+1}^t\right) mod\ 2$$

$c_{i-1}^t = 1 \qquad c_{i+1}^t = 1$

$(1+1)\ mod\ 2 = 0$

$c_{i-1}^t = 1 \qquad c_{i+1}^t = 0$

$(1+0)\ mod\ 2 = 1$

$c_{i-1}^t = 0 \qquad c_{i+1}^t = 1$

$(0+1) mod\ 2 = 1$

$c_{i-1}^t = 0 \qquad c = 0$

$(0+0) mod\ 2 = 0$

En general, un AC de k estados es aditivo si su regla puede escribirse como la suma de enteros múltiplos de su vecindario $(mod\ k)$, donde los enteros pueden tomar valores de 0 hasta k.

$$c_i^{t+1} = \left(\alpha c_{i-1}^t + \beta c_i^t + \gamma c_{i+1}^t\right) mod\ k$$

donde las constantes $\alpha, \beta,$ y γ tomaran valores entre 0 y k. Por lo tanto, existen k^{2r+1} reglas aditivas entre las k^{k^r} posibles reglas con k estados y un vecindario con radio r.

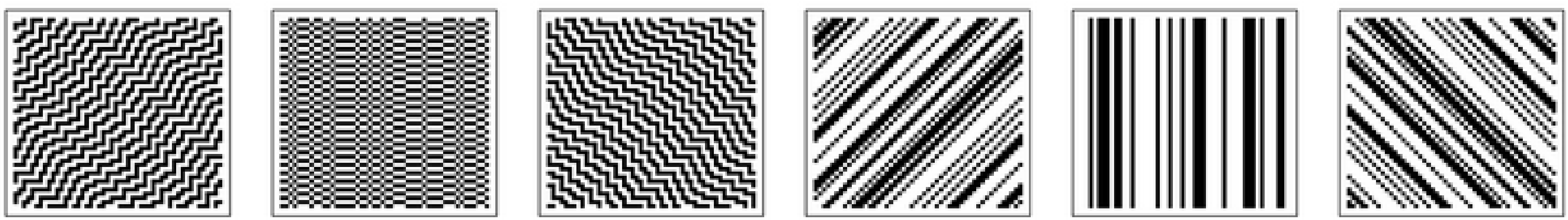
Es decir que para los autómatas celulares elementales solo hay 8 reglas aditivas. La siguiente tabla resume dichas reglas:

Regla	Suma (mod 2)
0	0
60	$c_{i-1} + c_i$
90	$c_{i-1} + c_{i+1}$
102	$c_i + c_{i+1}$
150	$c_{i-1} + c_i + c_{i+1}$
170	c_{i+1}
204	c_i
240	c_{i-1}

Reversibilidad

En un AC una configuración particular tiene descendientes únicos, pero no necesariamente tiene ancestros únicos. Un AC reversible es aquel en que cada configuración tiene un predecesor único

Dentro de los AC elementales solamente 6 son reversibles:



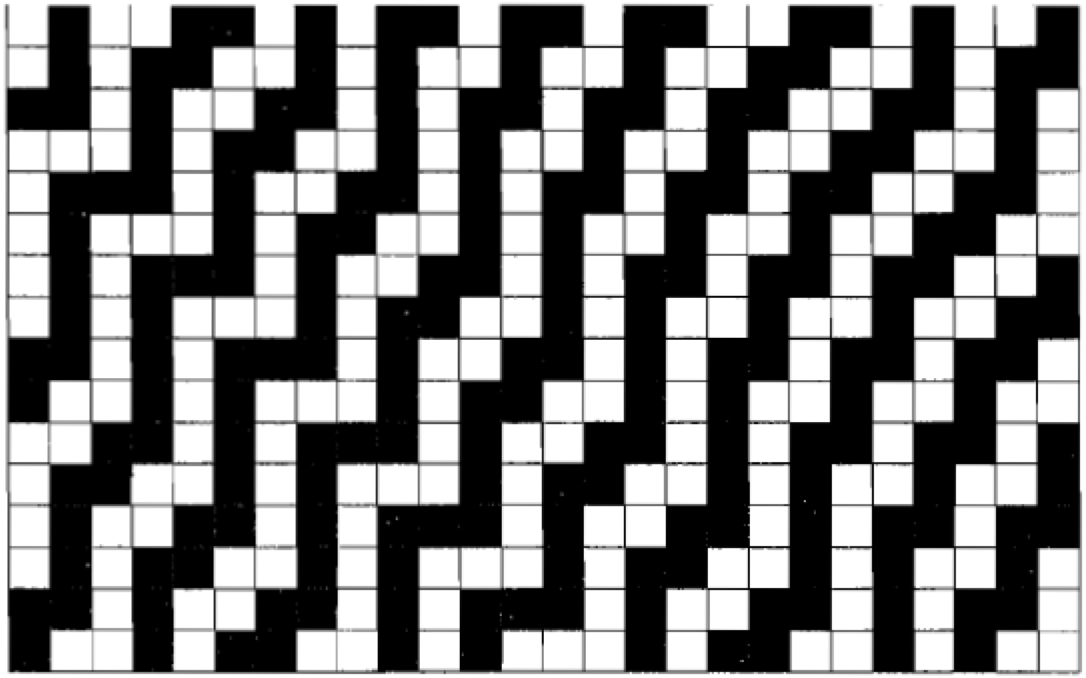
rule 15 | rule 51 | rule 85 | rule 170 | rule 204 | rule 240

Un ejemplo simplificado se muestra en la siguiente tabla donde cada letra (a, b, c, ….) representa una configuración de celdas en un determinado tiempo. Si tuvieras un autómata con 10 células serian $2^{10} = 1024$ posibles configuraciones

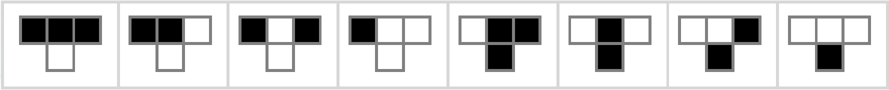
	a	b	c	d	e	f	g	h
1	b	c	d	e	f	g	h	g
2	c	d	e	f	g	h	g	h
3	d	e	f	g	h	g	h	g
4	e	f	g	h	g	h	g	h
5	f	g	h	g	h	g	h	g
6	g	h	g	h	g	h	g	h

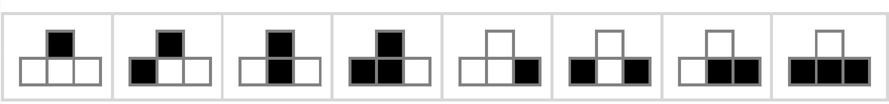
Podemos observar que el estado a nunca se alcanza, estos estados son conocidos como *Jardín del Edén.* Los AC reversibles no deben tener este tipo de estados

Regla 15



Se puede iniciar en cualquier línea de la configuración del autómata y determinar el estado anterior usando la regla rotada 180°.



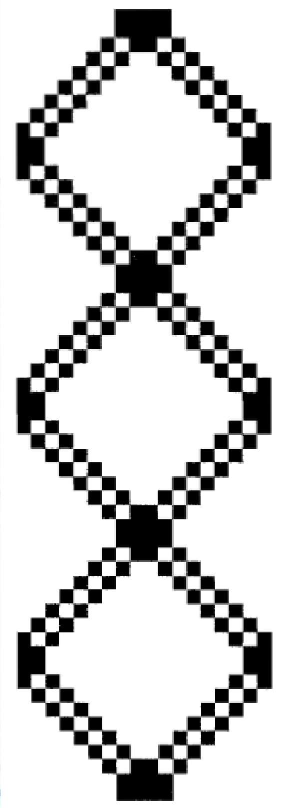


Es posible la creación de autómatas celulares reversibles [3], al incluir en la función de transición el estado de la celda a actualizar en el tiempo $(t - 1)$.

$$c_i(t+1) = f\big(c_{i-1}(t), c_i(t), c_{i+1}(t)\big) + c_i(t-1)$$

Para la cual existirá una función inversa única dada por:

$$c_i(t-1) = f\big(c_{i-1}(t), c_i(t), c_{i+1}(t)\big) + c_i(t+1)$$



El AC reversible basado en la regla 90 (rule 90R)

$$c_i(t+1) = c_{i-1}(t) + c_{i+1}(t) + c_i(t-1)\ mod\ 2$$

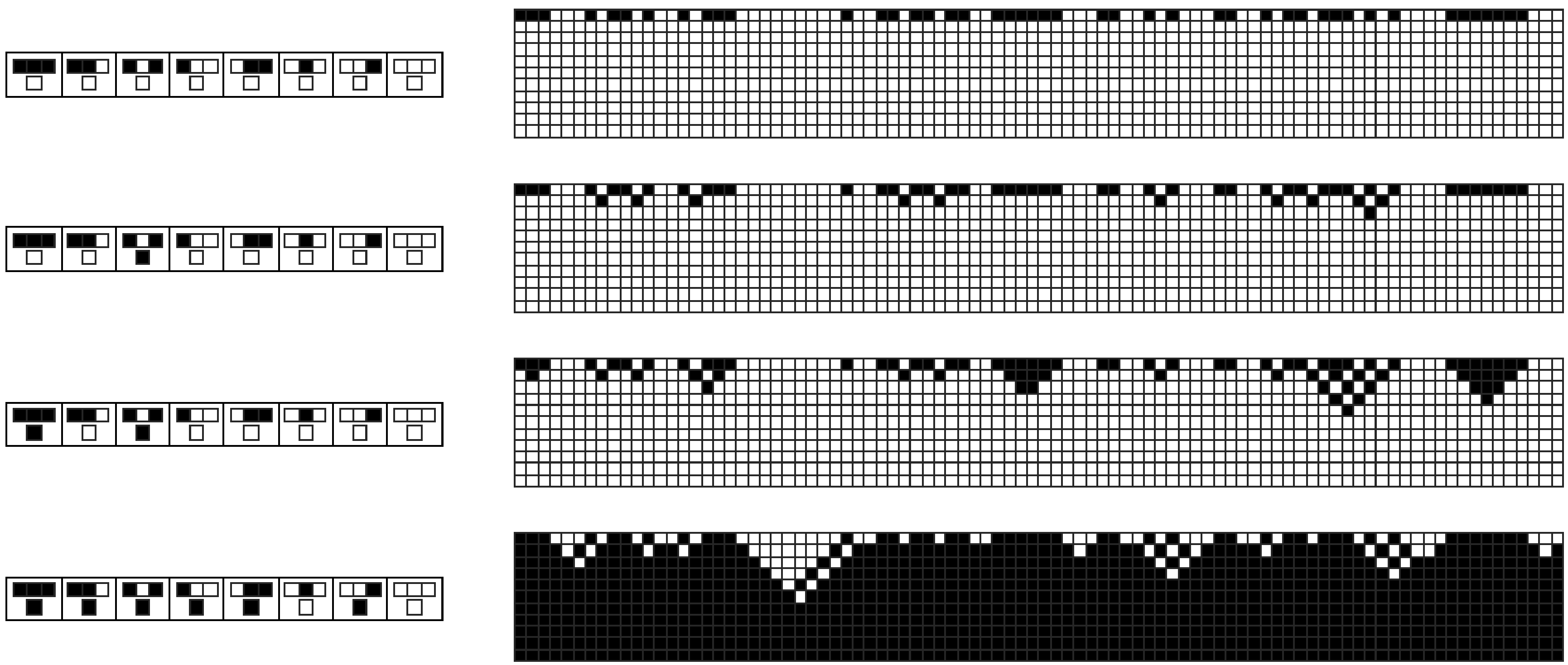
Es decir, para actualizar el valor de celda, se usa la celda del lado izquierdo, la celda del lado derecho y la celda atrás (arriba). La inversa esta dada por:

$$c_i(t-1) = c_{i-1}(t) + c_{i+1}(t) + c_i(t+1)\ mod\ 2$$

Clasificación

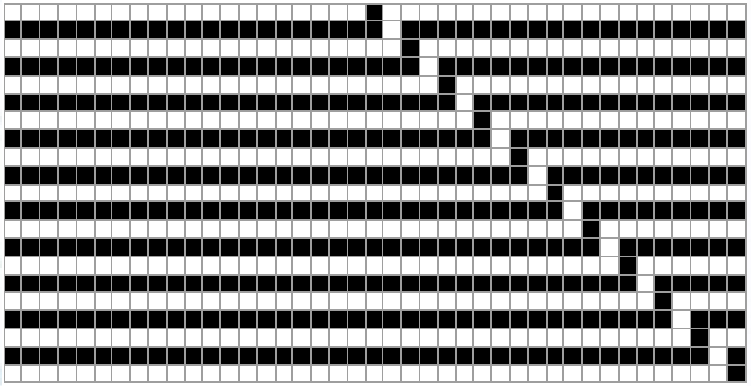
A inicios de los 80 Wolfram propuso el clasificar los AC de acuerdo con la complejidad de su comportamiento. 4 clases fueron definida, aunque algunos AC tienen una mezcla de diferentes clases

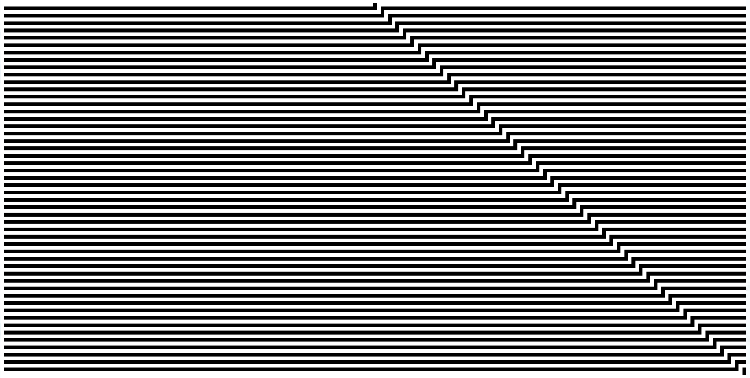
Clase 1: el comportamiento es muy simple y casi todas las condiciones iniciales conducen exactamente al mismo estado final uniforme.



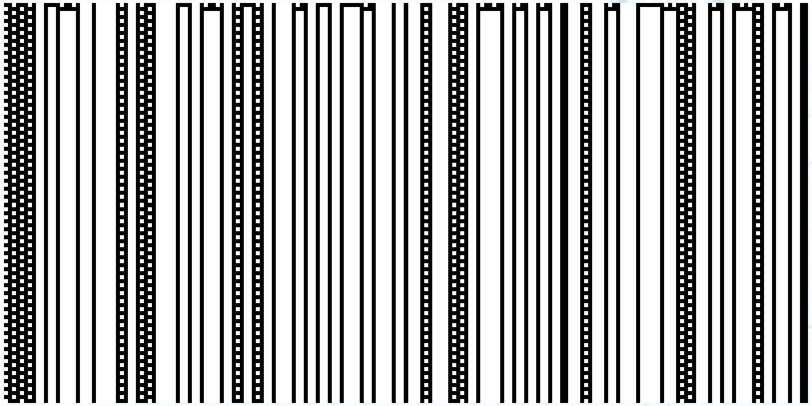
Clase 2: hay muchos estados finales diferentes posibles, pero todos ellos consisten solo en un cierto conjunto de estructuras simples que permanecen iguales para siempre o se repiten cada pocos pasos.

Regla 15


20 steps

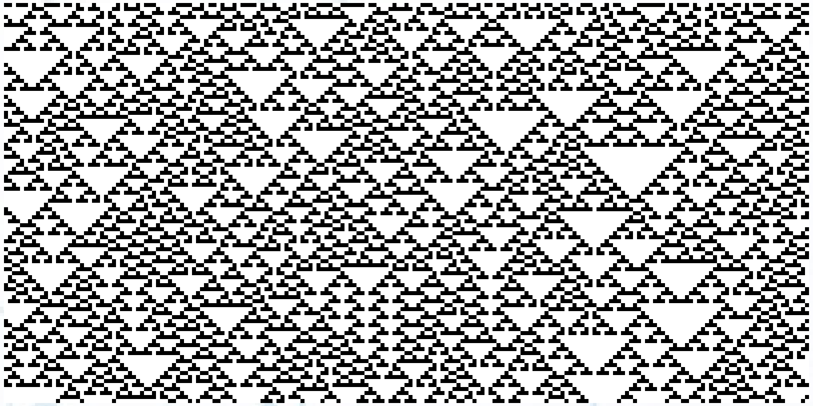

100 steps

Regla 108

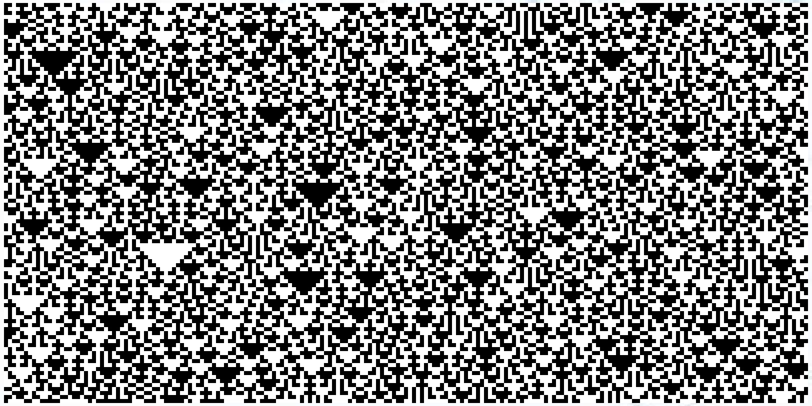


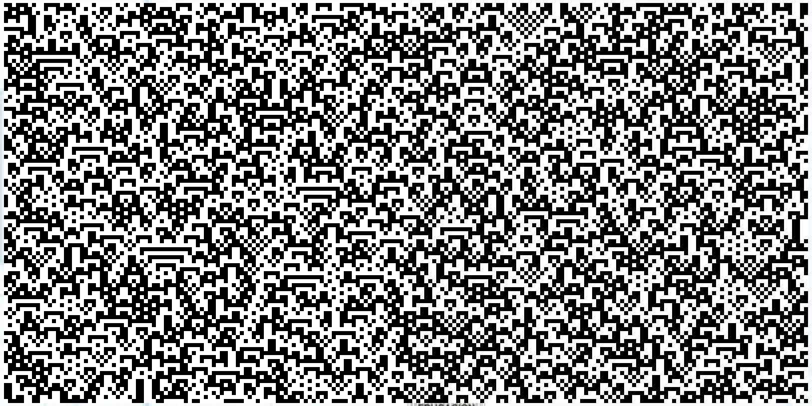
Clase 3: el comportamiento es más complicado y parece en muchos aspectos aleatorio, aunque los triángulos y otras estructuras a pequeña escala siempre se ven en algún nivel.

Regla 22



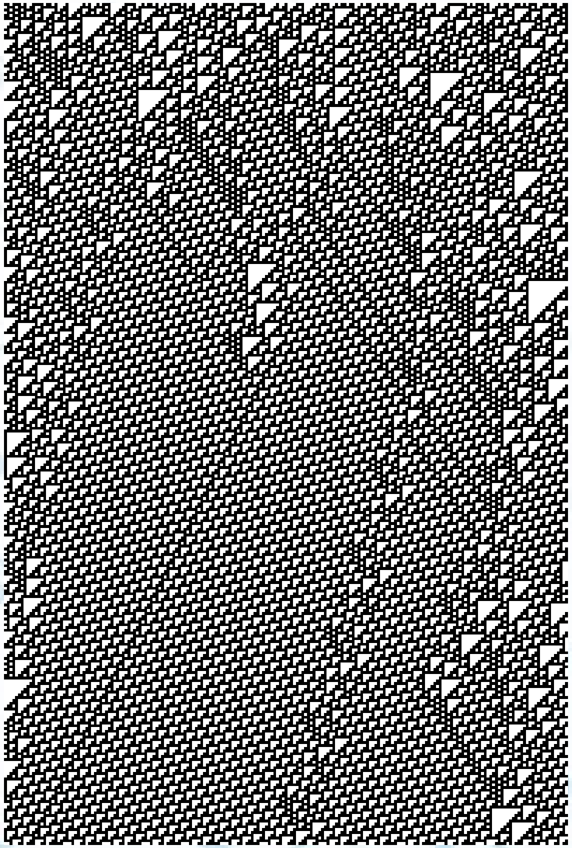
Regla 150



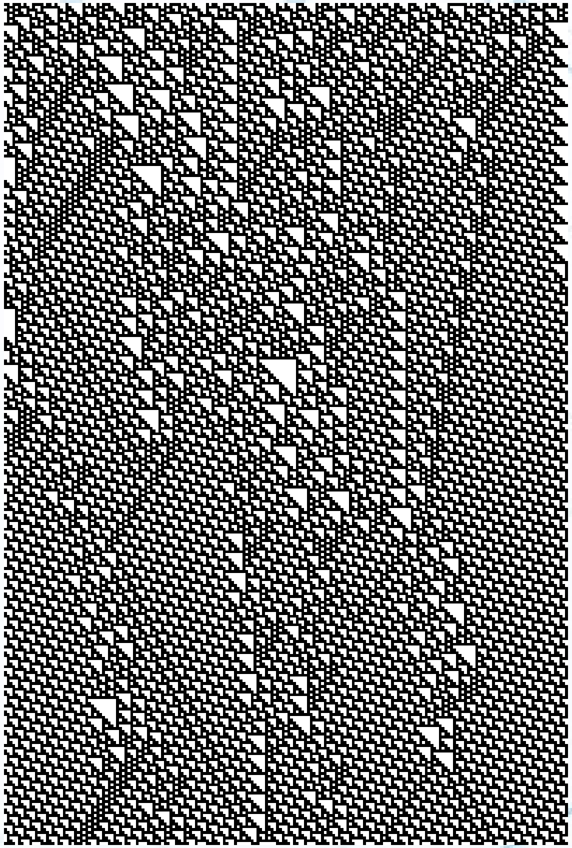


Regla 105

Clase 4: Implica una mezcla de orden y aleatoriedad: se producen estructuras localizadas que en sí mismas son bastante simples, pero estas estructuras se mueven e interactúan entre sí de formas muy complicadas.



Regla 110



Regla 124

Bibliografía

1. Rozenberg, G., Bäck,T., & Kok, J.N. (2012). Handbook of Natural Computing (1st. ed.). Springer Publishing Company, Incorporated.

2. Hadeler, K.P., & Müller, J. (2017). Cellular automata: analysis and applications. Springer.

3. Schiff, J.L. (2008). Cellular Automata: A Discrete View of the World. Wiley.

4. Li, W. & Packard, N. (1990). The structure of the elementary cellular automata rule space. Complex Systems, vol. 4, pp. 281-297.

5. Wolfram, S. (2002). A new kind of science. Wolfram Media.

¡ Gracias !

Thanks !

Obrigado

Xie xie ni

Domo arigatou

Спасибо

Merci

Grazie