Universidad Nacional Autónoma de México

Facultad de Ciencias

Modelación y simulación computacional basada en Agentes

Práctica 1: "Implementación y análisis de autómatas celulares"

Martínez Hidalgo Paola Mildred

2025-2

Parte 1. Preguntas "Implementación y análisis de autómatas celulares"

- **a) ¿Qué es la modelación basada en agentes?**

  Es un modelo computacional el cual nos ayuda a simular sistemas complejos, acciones e interacciones de agentes autónomos. Un agente lo podemos ver como un sistema computacional que es situado en un ambiente y tiene la capacidad de acción autónoma en su ambiente con el fin de alcanzar los objetivos diseñados.

- **b) ¿Qué es el enfoque** $bottom - up$**?**

  El enfoque $botton - up$ (de abajo hacia arriba) es una estrategia de diseño y análisis el cual se caracteriza por comenzar con componentes y detalles específicos, y luego construir gradualmente un sistema más grande, es decir, se toman decisiones desde los niveles más bajos hacia arriba. Por ejemplo, en la programación los desarrolladores comienzan escribiendo y probando componentes de bajo nivel, como funciones o módulos individuales. Luego, estos componentes se ensamblan para formar partes más grandes del sistema, y así sucesivamente, hasta que se complete todo el proyecto.

- **c) ¿Cuándo se usa el concepto de autoorganización?**

  Este concepto se usa cuando un sistema complejo desarrolla comportamientos ordenados sin una autoridad central y sin ayuda o control externo. Este concepto lo podemos encontrar en biología, neurociencia, ciencias sociales y economía, en computación y sistemas artificiales, etc.

  Uno de los ejemplos más comunes es la autoorganización en las hormigas. Las hormigas dejan rastros de feromonas al buscar comida. Otras hormigas siguen los caminos con más feromonas, reforzando ciertas rutas sin necesidad de un líder.

- **d) ¿Qué es una propiedad emergente?**

  Una propiedad emergente es un atributo o característica el cual se presenta en sistemas complejos, consiste en el funcionamiento de un sistema que no puede explicarse ni predecirse completamente a partir de las propiedades de sus componentes individuales.

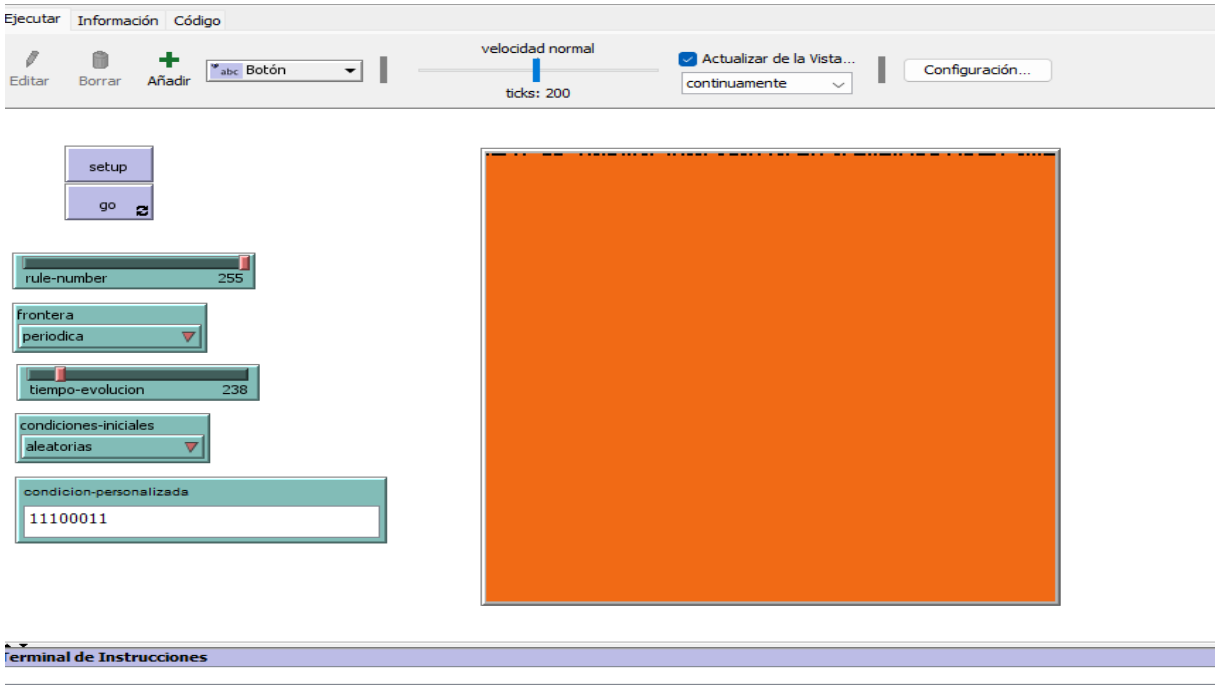
Parte 2. Ejercicios

- **1. Exploración. Ejecute algunas de las 256 reglas del autómata celular a partir de la implementación de su programa con condiciones de frontera periódicas (cilindro), mínimo 100 tiempos de evolución y condiciones iniciales aleatorias. Identifique el tipo de dinámica que presenta y a que clase pertenece según las cuatro categorías de la Clasificación de Wolfram. Muestre 1 representante de cada clase, adjunte la "captura de pantalla" y expliqué por qué consideran que pertenece a tal clase. Después de explorar todas o algunas de las 256 reglas con el programa que implementaste ¿Cuál crees que sea la clase más frecuente?**

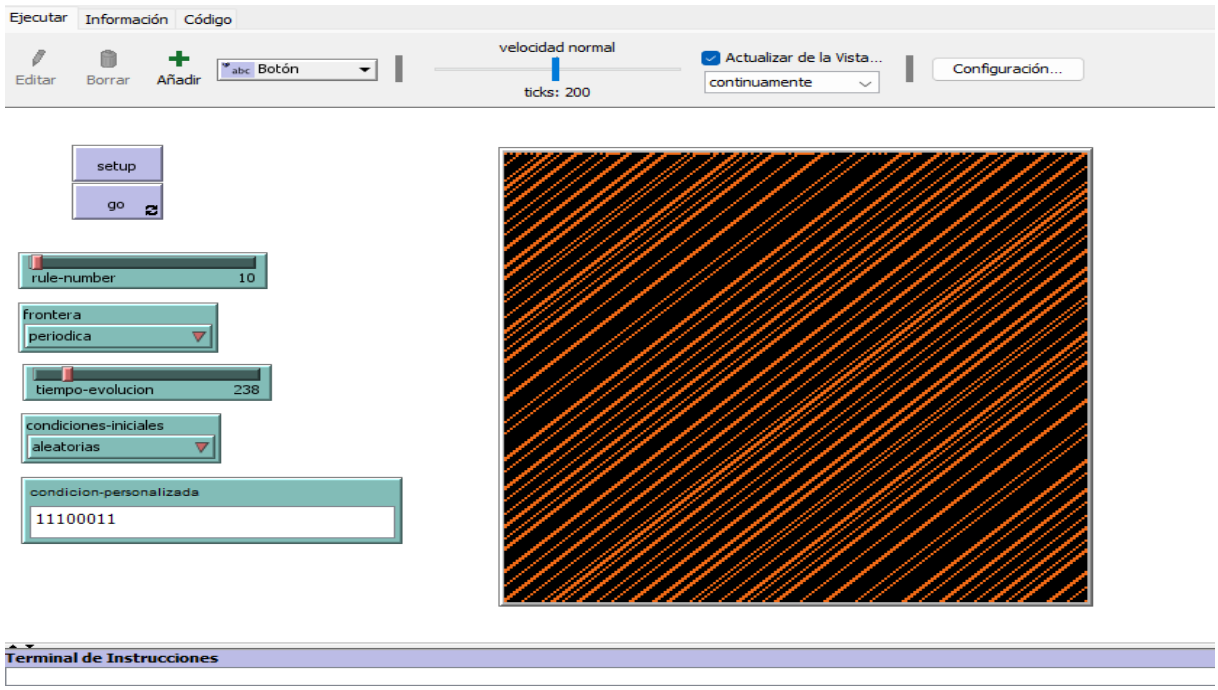
  NOTA* tomar en cuenta que en este caso estamos tomando de la regla 0 a 255

Capturas de pantalla

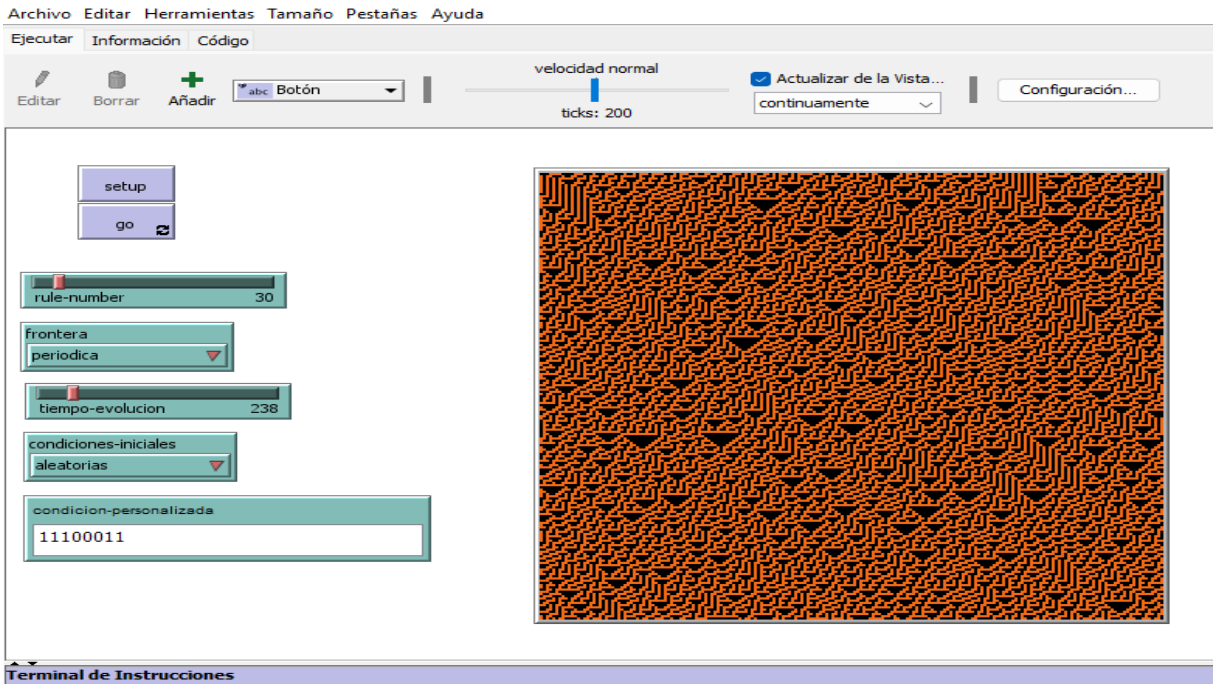
Clase 1 Comportamiento uniforme: La evolución del sistema lleva a un estado homogéneo, sin estructuras espaciales ni temporales de ningún tipo. En este caso tenemos la regla 255, en ella podemos observar que todo el mundo se pone de un solo color, en este caso naranja.



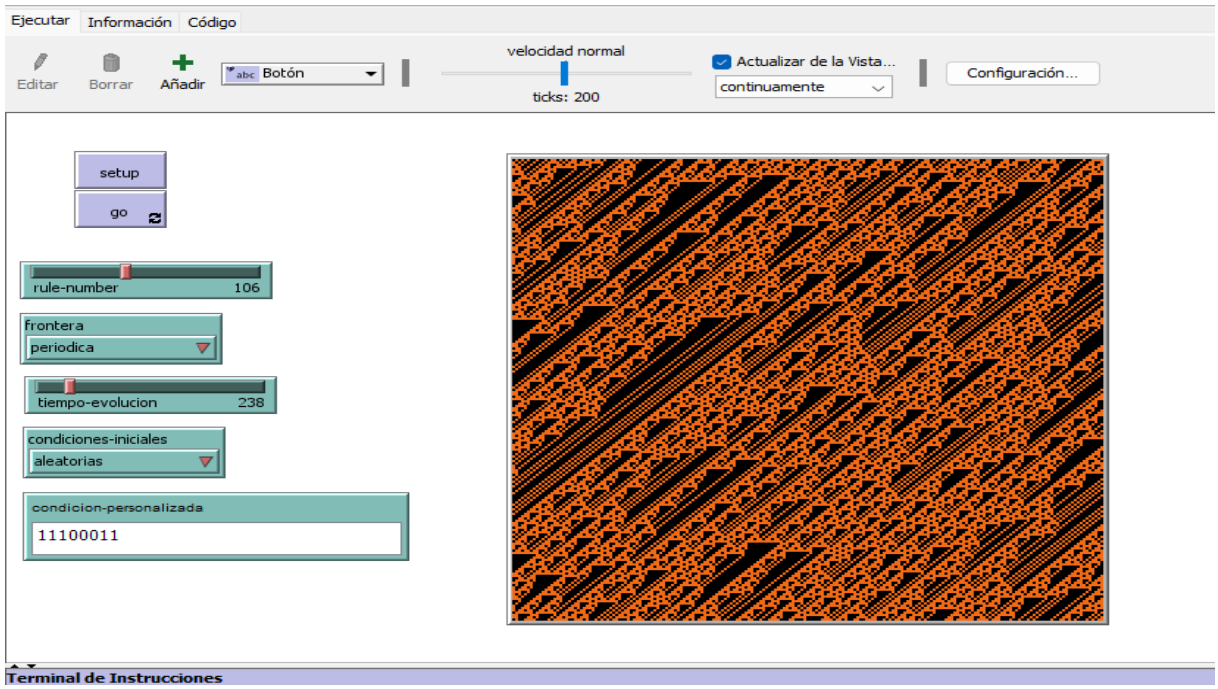
Clase 2 Comportamiento periódico: La evolución del sistema da lugar a estructuras separadas de tipo estable o periódico. En este caso tenemos la regla 10, en ella podemos notar patrones que se repiten en el tiempo, en este caso líneas.



Clase 3 Comportamiento caótico: La evolución da lugar a patrones caóticos, surgen estructuras fractales y temporalmente hay ciclos de longitud muy grande. En este caso tenemos la regla 30, en ella podemos observar patrones que no se repiten y parecen aleatorios, con estructuras fractales o caóticas.



Clase 4 Comportamiento complejo: La evolución genera estructuras complejas localizadas, esta clase es una combinación de las tres clases anteriores. En este caso tenemos la regla 106, en ella se observan patrones que no son completamente ordenados ni completamente caóticos, con estructuras que se mueven y colisionan.



Pregunta

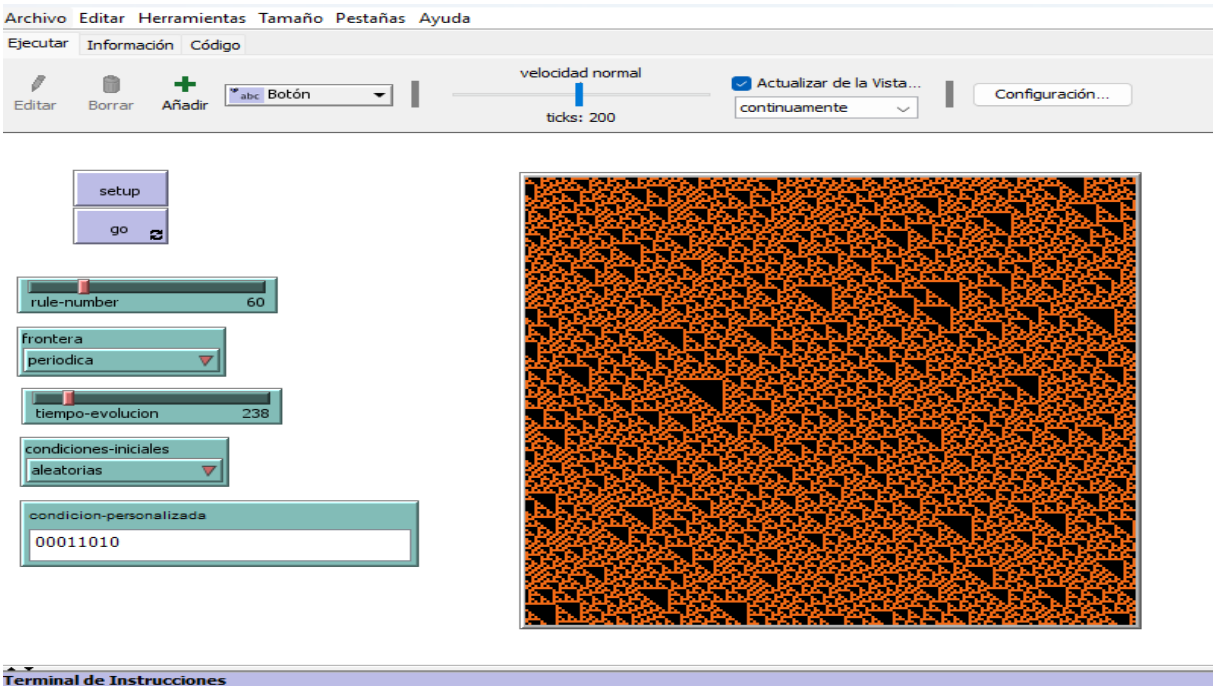
¿Cuál crees que sea la clase más frecuente?

Yo diría que la clase 3 son las más frecuentes, ya que muchas reglas generan comportamientos complejos y aparentemente aleatorios.

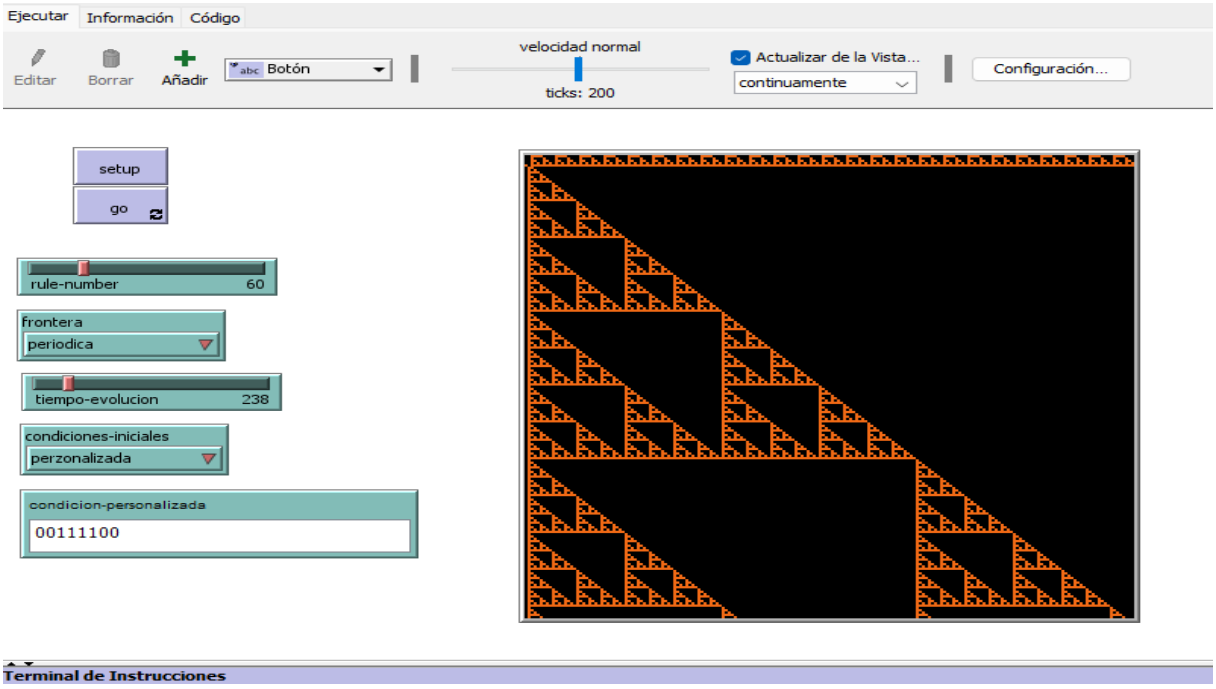
- **2. Sensibilidad a las condiciones iniciales. Encuentre un autómata de clase III, perturbe la condición inicial aleatoria por un bit, evolucione y trate de ver las diferencias entre el autómata perturbado y el normal. Hint: pueden tomar las diferencias entre los estados de las celdas y pintarlos en la posición correspondiente. ¿La sensibilidad a las condiciones iniciales es una propiedad suficiente para catalogar la dinámica como caótica? Explique. Adjunte las imágenes del ACE perturbado, sin perturbar y la diferencia de estados**

Imágenes

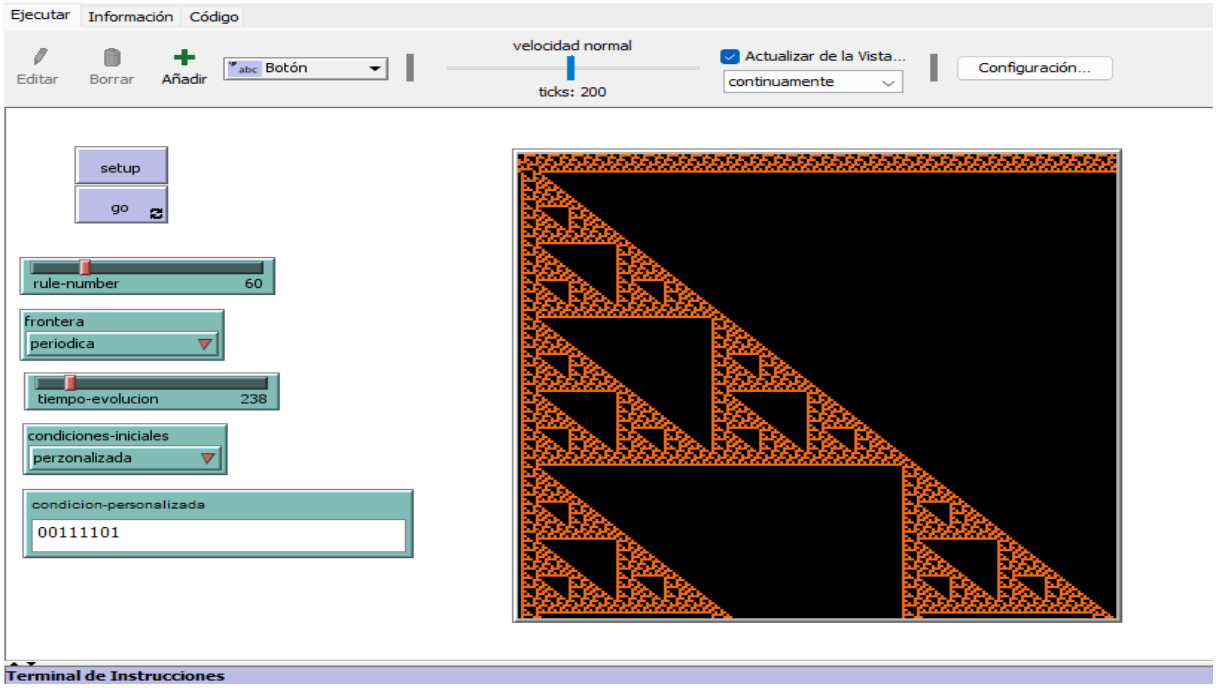
Para esto elegimos la regla 60, al tener esta regla aún sin ninguna perturbación tenemos lo siguiente:



Después en condición inicial seleccionaremos personalizada, en este caso en número binario de 60 es 00111100, lo colocamos en la entrada y obtenemos lo siguiente:



Ahora lo que haremos será que cambiaremos un bit del número binario, en este caso el último por lo que quedará así 00111101



Pregunta

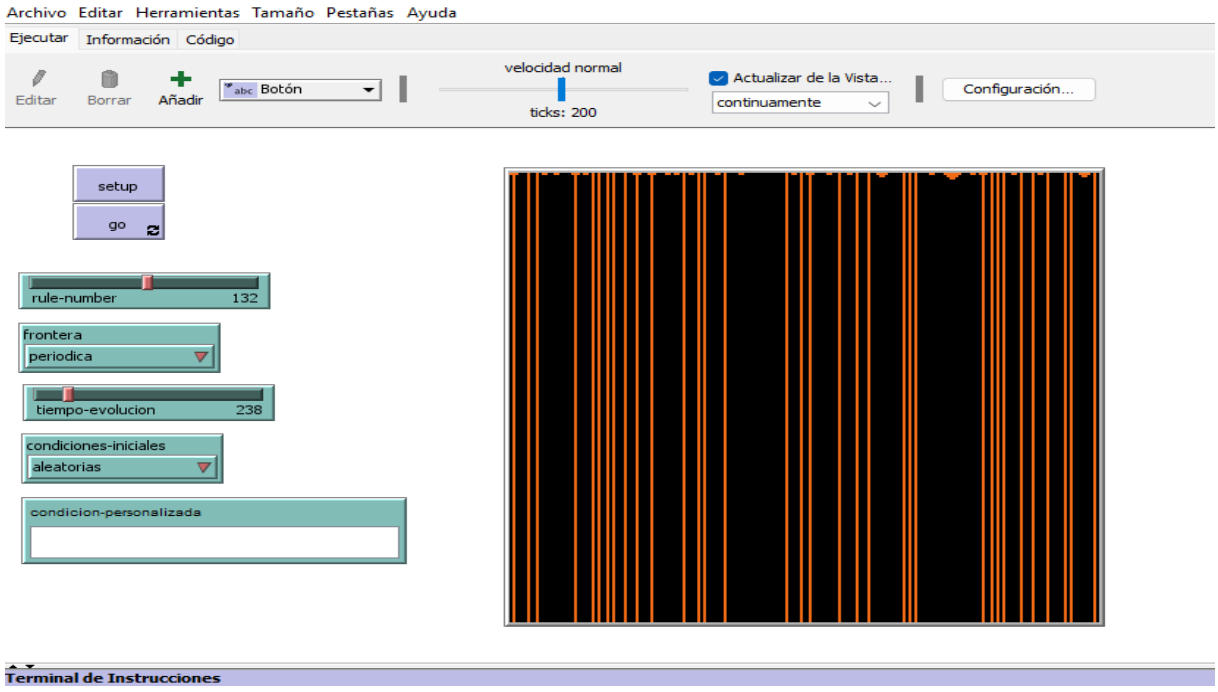
¿La sensibilidad a las condiciones iniciales es una propiedad suficiente para catalogar la dinámica como caótica?

La sensibilidad a las condiciones iniciales es una propiedad suficiente para catalogar la dinámica como caótica en el contexto de los autómatas celulares. Esto se debe a que un pequeño cambio en la condición inicial provoca grandes diferencias en la evolución del sistema, lo que es característico de los sistemas caóticos. En este caso lo podemos observar, ya que al cambiar un sólo bit, podemos observar cambios significativos.

- **3. ¿Qué poder de cómputo tiene la regla 132? Explique y adjunte captura de pantalla. ¿Qué propiedades tiene la regla 30? Explique y adjunte capturas de la simulación**

Regla 132:

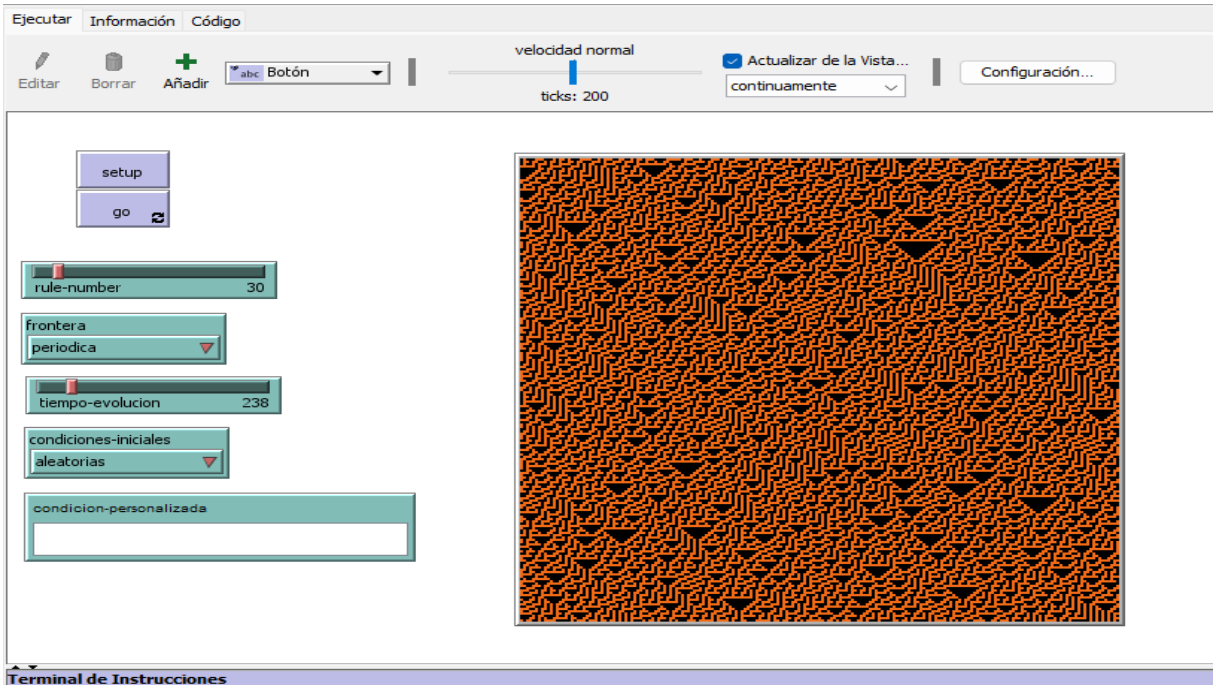
Lo primero que podemos notar de la regla 132 es que pertenece a la clase 2 (periódico), en cuanto a su poder de cómputo no es Turing completo, esto debido a que su comportamiento es demasiado simple para realizar cálculos universales. Genera patrones repetitivos y no muestra la complejidad necesaria para la computación universal.



Regla 30:

Lo primero que podemos observar es que pertenece a la clase 3, es decir, tiene un comportamiento caótico, genera patrones aparentemente aleatorios. Entre su propiedades tenemos:

- Sensibilidad a las condiciones iniciales: Pequeños cambios en las condiciones iniciales pueden llevar a patrones completamente diferentes.
- Los patrones generados tienen una estructura fractal.
- No es Turing completa, es decir, no puede realizar cálculos universales.
- Es un autómata celular que genera un comportamiento caótico y se usa como generador de números pseudoaleatorios.



- **4. En las siguientes imágenes se muestra la evolución de la regla 22 con condición inicial simple (una celda negra en el centro, Figura A) y condición inicial aleatoria (Figura B). Explique lo que sucede con la dinámica.**

Figura A

Lo primero que podemos observar es que pertenece a la clase 2. Por otra parte la condición inicial simple (Figura A), tiende a generar patrones que se expanden desde la celda inicial, pero de manera controlada, también podemos notar que se forman patrones simétricos y que las estructuras son repetitivas.

Figura B

En la segunda imagen podemos notar que muestra un comportamiento más complejo y menos predecible. Las celdas encendidas interactúan entre sí, generando patrones que pueden ser más irregulares, bajo estas condiciones pertenece a la clase 3, esto debido a que se observa un patrón menos estructurado, con regiones de orden y desorden que interactúan entre sí.

Comparación entre ambas

Lo que podemos observar de ambas es que dependiendo de las condiciones iniciales, la Regla 22 puede mostrar comportamientos que se acercan a la Clase 3, aunque su comportamiento principal es de Clase 2, con esto podemos ver que las condiciones iniciales pueden afectar significativamente la dinámica de un autómata celular, incluso para reglas que no son completamente caóticas.

- **5. Si el autómata estuviera definido en un alfabeto de 3 estados considerando sus primeros vecinos, ¿cuántas posibles reglas hay?**

  Si tuviéramos tres estados, por ejemplo $0, 1$ y 2, los cuales representan la celda actual, vecino derecho y vecino izquierdo. Para sacar las posibles reglas hacemos lo siguiente:

  Cada vecino puede tener tres estados posibles, por lo que el tamaño de la vecindad sería de

$$3 \times 3 \times 3 = 27$$

Ahora para sacar las posibles reglas tenemos

$$3^{27} = 7,625,597,484,987$$

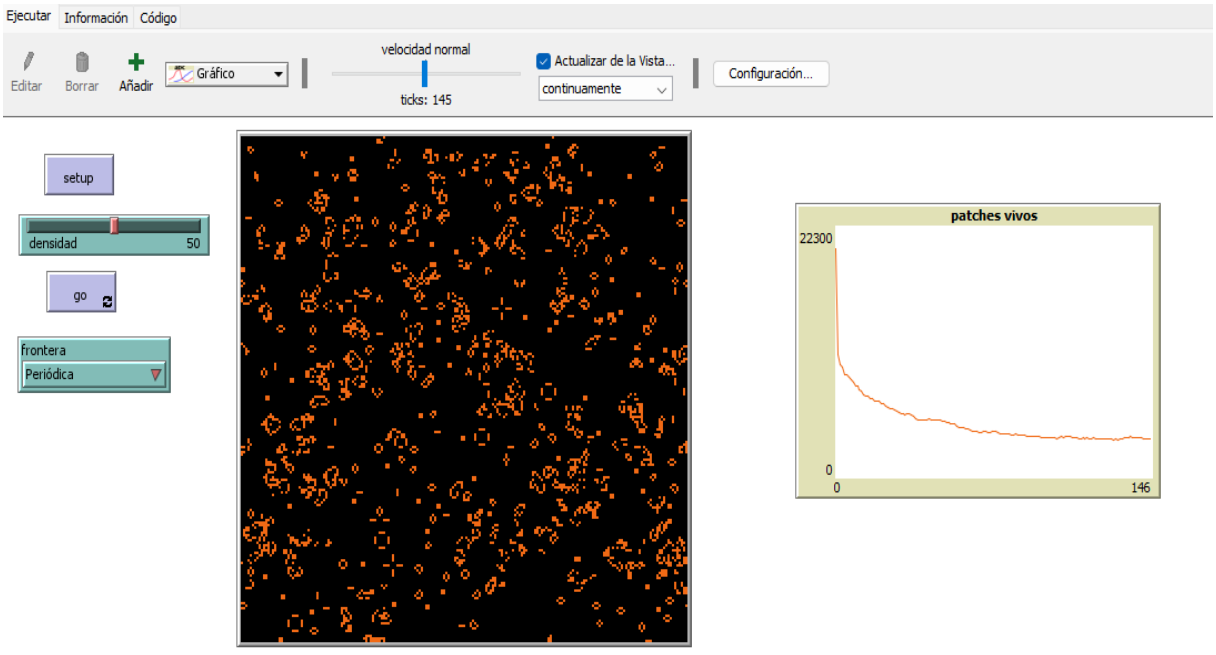
Con lo anterior podemos notar que al agregar un sólo estado el número de reglas aumenta mucho a comparación de las 256 reglas que tenemos con dos estados.

Parte 3. LIFE: EL JUEGO DE LA VIDA

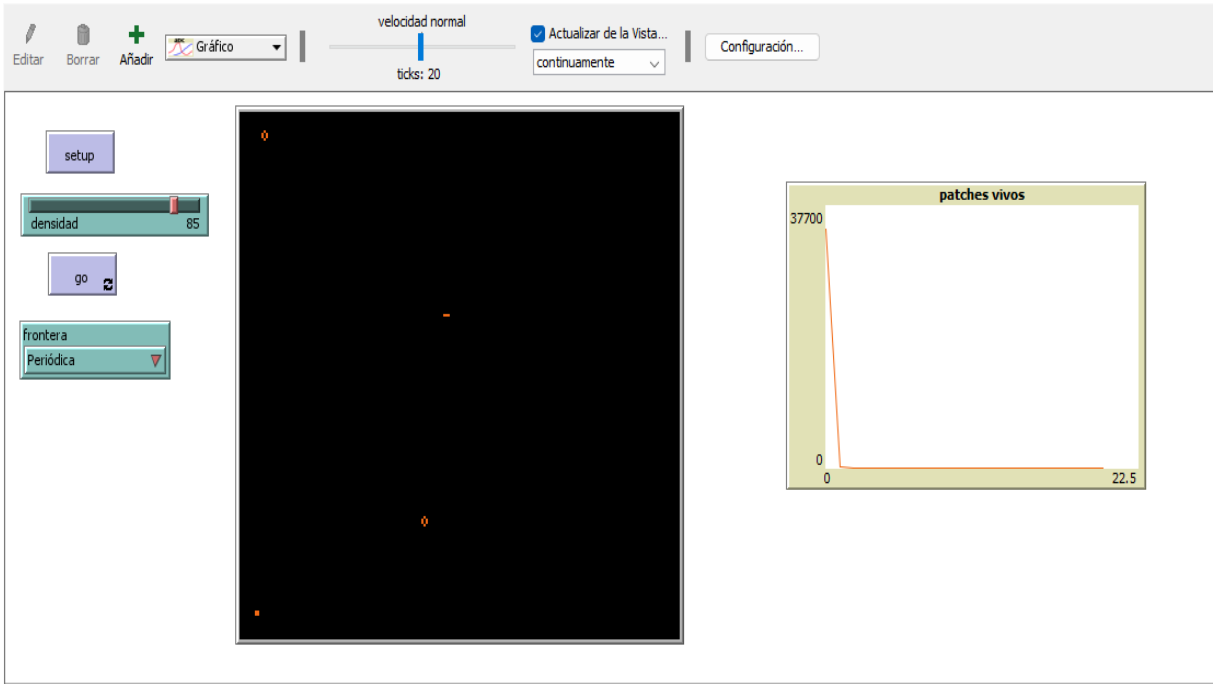
- **1. En NetLogo construya una gráfica de conteo de celdas vivas a través del tiempo. Ejecute varias veces su programa con una retícula de 100x100 celdas y condición aleatoria con 50 % de celdas ocupadas, ¿en cuánto tiempo (ticks) se .estabilizó"la dinámica?, ¿para cualquier configuración inicial aleatoria pasará lo mismo? Explique y adjunte una captura de pantalla del "view" de agentes y la gráfica de conteo (0.5 pt).Realice el mismo experimento aumentando la densidad de la condición aleatoria a 85 %, ¿sucede lo mismo? Explique.**

  - ¿En cuánto tiempo (*ticks*) se "*estabilizó*" la dinámica?
    Cuando la condición aleatoria es con 50 % de celdas ocupadas, el núm de ticks en que se estabiliza es aproximadamente de 20 a 50, a partir de ahí se puede observar la gráfica constante. Tomemos en cuenta que esto dependerá de la configuración inicial.
  - ¿Para cualquier configuración inicial aleatoria pasará lo mismo?
    No necesariamente, tomemos en cuenta que El Juego de la Vida es un sistema complejo y su comportamiento depende en gran medida de la configuración inicial. Algunas configuraciones pueden estabilizarse rápidamente, mientras que otras pueden tardar más.
  - ¿Qué sucede con una densidad inicial del 85 %?
    Lo que podemos observar es que se generan estructuras muy diferentes, en este caso vemos que el número de ticks en el que se estabiliza es casi inmediato, esto se puede deber a que la densidad es algo alta y el mayor número de celdas están ocupadas.

Condición aleatoria con 50 % de celdas ocupadas.



Condición aleatoria con 85 % de celdas ocupadas.



- **2. Dados los siguientes elementos implemente la compuerta NOT con el diagrama de la Figura D). ¿Qué implicaciones tiene controlar ciertas estructuras en función del tiempo?**

  En este caso al usar la operación lógica NOT, y usar Gosper's Gun, Eater y Gliders, al controlar el comportamiento en función del tiempo podemos observar cómo interactúan y se desarrollan a lo largo del tiempo, esto puede resultar útil para entender el comportamiento y los patrones de la simulación.Por ejemplo, ver qué pasa cuando dos de las estructuras se cruzan.

  Por otra parte gestionar el tiempo permite observar fenómenos emergentes y puede llevar a la creación de sistemas computacionales más complejos basados en las interacciones de estructuras simples.

- **3. ¿El juego de la vida es reversible? Explique**

  La respuesta es no, esto lo podemos saber ya que en El Juego de la Vida no es posible recuperar un estado anterior a partir de un estado dado. Por ejemplo, dos configuraciones diferentes en t pueden llevar a la misma configuración en $t + 1$. Esto significa que no hay una manera única de "*revertir*" el sistema para recuperar el estado anterior, dicho de otra forma, varias configuraciones diferentes pueden evolucionar hacia un mismo estado en el siguiente paso, lo que genera colisiones irreversibles.

Bibliografía

Casero, A. (2024, March 15). ¿Qué es la programación bottom up? | KeepCoding Bootcamps. KeepCoding Bootcamps. `https://keepcoding.io/blog/que-es-la-programacion-bottom-up/#Enfoque_bottom_up`

Definición de propiedad emergente (teoría de sistemas). (2023, June 19). Alegsa.com.ar. `https://www.alegsa.com.ar/Dic/propiedad_emergente.php#h0`