INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

EVOLUTIONARY COMPUTING

PRÁCTICA 7: AUTOMATAS CELULARES

ALUMNO: ORTEGA VICTORIANO IVAN

PROF.: JORGE LUIS ROSAS TRIGUEROS

FECHA DE REALIZACIÓN DE LA PRÁCTICA: 24/04/2018

FECHA DE ENTREGA DEL REPORTE: 01/05/2018

MARCO TEÓRICO.

Para los matemáticos es fundamental el utilizar y desarrollar herramientas, que expliquen los fenómenos que nos rodean. Esto se logra, generalmente, a través de modelos matemáticos que den respuesta a dichos fenómenos. Así, se puede señalar el gran avance adquirido en el estudio del caos [1] y de los sistemas dinámicos [2], en este último destaca el de los autómatas celulares. Por otro lado, la computación puede verse como la transformación de la información, donde al inicio de este proceso siempre hay condiciones iniciales. Sin embargo, hay procesos de cómputo donde nuevas entradas de información pueden darse durante el proceso mismo. Esta información nueva a veces determina el resultado del proceso, lo que implica un enfoque distinto para estudiar la computación, donde el sistema sea capaz de cambiar de comportamiento ante cualquier perturbación, incorporando información nueva durante el proceso. Para auxiliar a ambos enfoques, es de mucha ayuda el estudio y simulación de sistemas dinámicos, evitando las desventajas existentes en la matemática clásica para expresar la complejidad de estos sistemas. Por ello se recurre a un método de modelización conocido como autómatas celulares. Ciertos autómatas celulares son universales, es decir son capaces de representar cualquier algoritmo. Estos son maquinas abstractas capaces de construir nuevos autómatas que a su vez pueden generar otros. En otras palabras, son capaces de procesar cualquier cosa computable. Ahora ¿hay cosas incomputables? Esta es una pregunta difícil, y por ello los autómatas universales son las maquinas abstractas - concepto general de computadora más potentes que se conocen. [3]

Los autómatas celulares(AC) surgen en la década de 1940 con John Von Neumann, que intentaba modelar una máquina que fuera capaz de autoreplicarse, llegando así a un modelo matemático de dicha maquina con reglas complicadas sobre una red rectangular. Inicialmente fueron interpretados como conjunto de células que crecían, se reproducían y morían a medida que pasaba el tiempo. A esta similitud con el crecimiento de las células se le debe su nombre. Un autómata celular es un modelo matemático para un sistema dinámico, compuesto por un conjunto de celdas o células que adquieren distintos estados o valores. Estos estados son alterados de un instante a otro en unidades de tiempo discreto, es decir, que se puede cuantificar con valores enteros a intervalos regulares. De esta manera, este conjunto de células logra una evolución según una determinada expresión matemática, que es sensible a los estados de las células vecinas, la cual se le conoce como regla de transición local. El aspecto que más caracteriza a los AC es su capacidad de lograr una serie de propiedades que surgen de la propia dinámica local a través del paso del tiempo y no desde un inicio, aplicándose a todo el sistema en general. Por lo tanto, no es fácil analizar las propiedades globales de un AC desde su comienzo, complejo por naturaleza, a no ser por vía de la simulación, partiendo de un estado o configuración inicial de células y cambiando en cada instante los estados de todas ellas de forma síncrona. [3]

ELEMENTOS DE UN AUTÓMATA CELULAR

La definición de un AC requiere mencionar sus elementos básicos:

- **Arreglo Regular.** Ya sea un plano de 2 dimensiones o un espacio n-dimensional, este es el espacio de evoluciones, y cada división homogénea de arreglo es llamada célula.
- **Conjunto de Estados.** Es finito y cada elemento o célula del arreglo toma un valor de este conjunto de estados. También se denomina alfabeto. Puede ser expresado en valores o colores.
- **Configuración Inicial.** Consiste en asignar un estado a cada una de las células del espacio de evolución inicial del sistema
- **Vecindades.** Define el conjunto contiguo de células y posición relativa respecto a cada una de ellas. A cada vecindad diferente corresponde un elemento del conjunto de estados
- **Función Local.** Es la regla de evolución que determina el comportamiento del AC. Se conforma de una célula central y sus vecindades. Define como debe cambiar de estado cada célula dependiendo de los estados anteriores de sus vecindades. Puede ser una expresión algebraica o un grupo de ecuaciones. [2]

MATERIAL Y EQUIPO.

- Equipo de Cómputo
- Python3
- Golly

DESARROLLO DE LA PRÁCTICA.

El primer experimento consistió en utilizar la regla ya definida del juego de la vida en Golly, y describir los siguientes comportamientos:

- Una vida estática con 7 o más células vivas.
- Un oscilador con 6 o más células vivas.
- Un deslizador con 10 o más células vivas.

Los resultados obtenidos de este experimento se muestran en las figuras 1, 2 y 3 respectivamente.

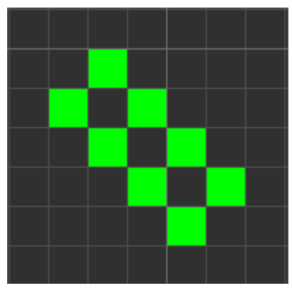

Figura 1. Vida estática con 8 células vivas.

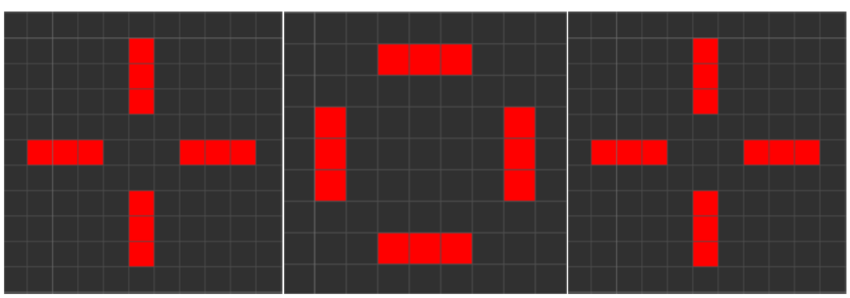

Figura 2. Oscilador con 9 células vivas.

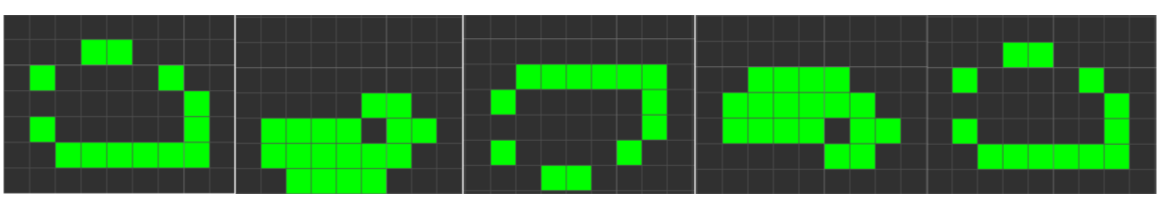

Figura 3. Deslizador con 13 y 18 células vivas.

El segundo experimento consistió en elegir 5 patrones ya definidos en Golly y explicar su comportamiento.

El primer patrón que escogí fue “Infinity Hotel” que se muestra en la figura 4:

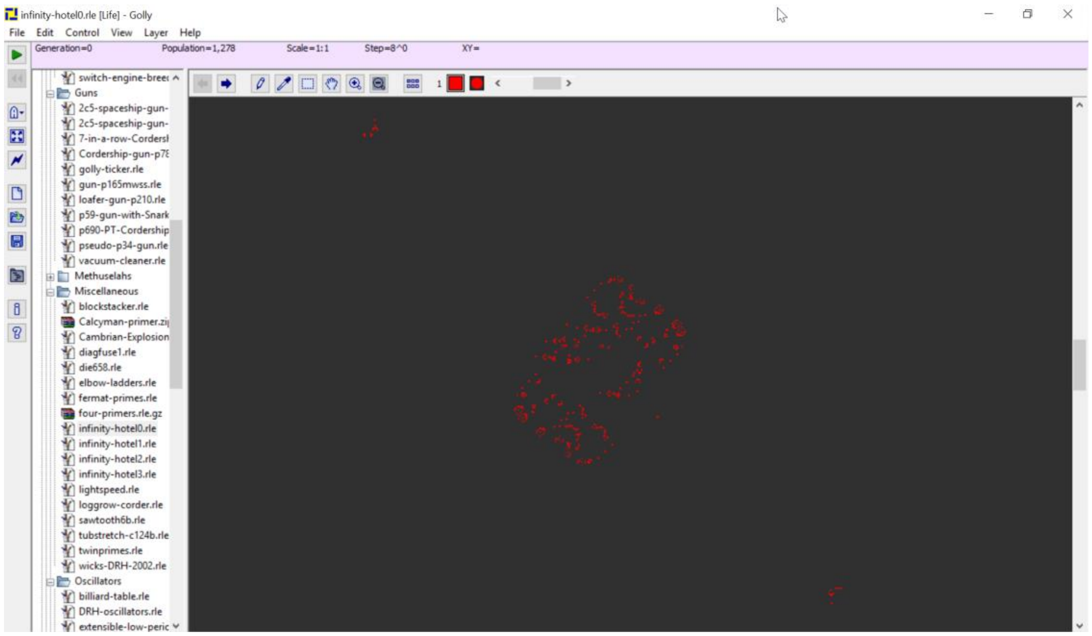

Figura 4. Infinity Hotel Generación 0.

Para poder entender mejor como funcionaba este patrón, tuve que dejarlo correr por muchas generaciones, ya que al inicio era un poco complicado entender que hacía. Después de analizarlo, me di cuenta de que los extremos inferior izquierdo y superior derecho (que en un principio están contenidos en el centro), siempre se estaban alejando, que, desde mi punto de vista, estos son los hoteles. Y aunque estos se iban alejando cada vez más, surgían nuevos elementos que “visitaban a los hoteles”. Es por ello a lo que se debe su nombre, ya que se va alejando sin detenerse, tendiendo al infinito.

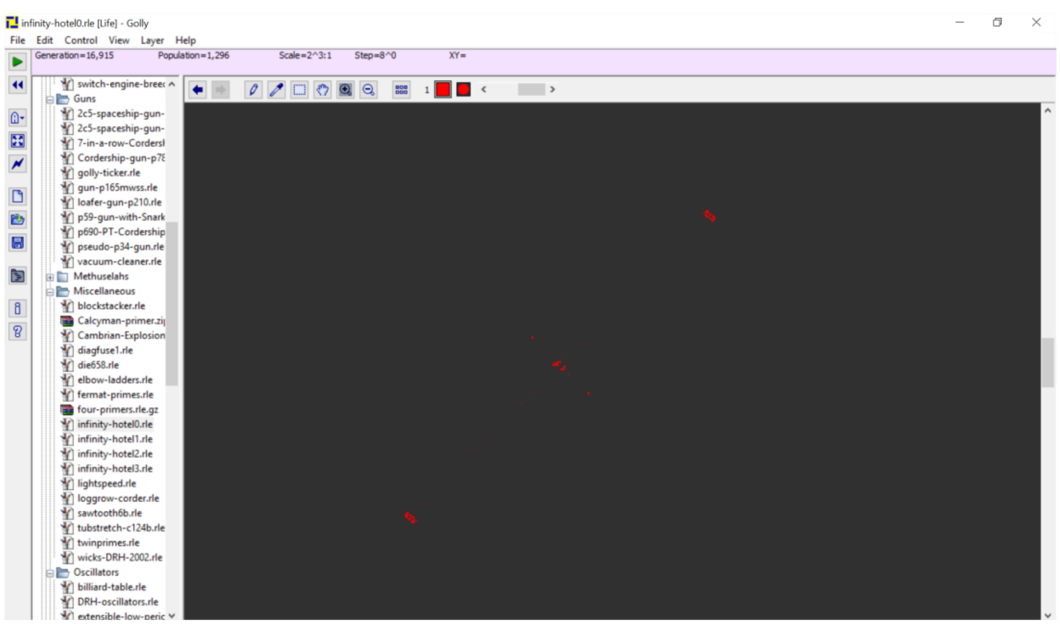

Figura 5. Infinity Hotel después de 16,915 generaciones.

El segundo patrón que escogí fue “DiagFuse” (figura 6), en este patrón, se parte con 6 puffers al parecer, los cuales conforme aumenta el número de generaciones, se van desplazando, pero en diagonal, es fácil darse cuenta con ver el rastro que deja el puffer.

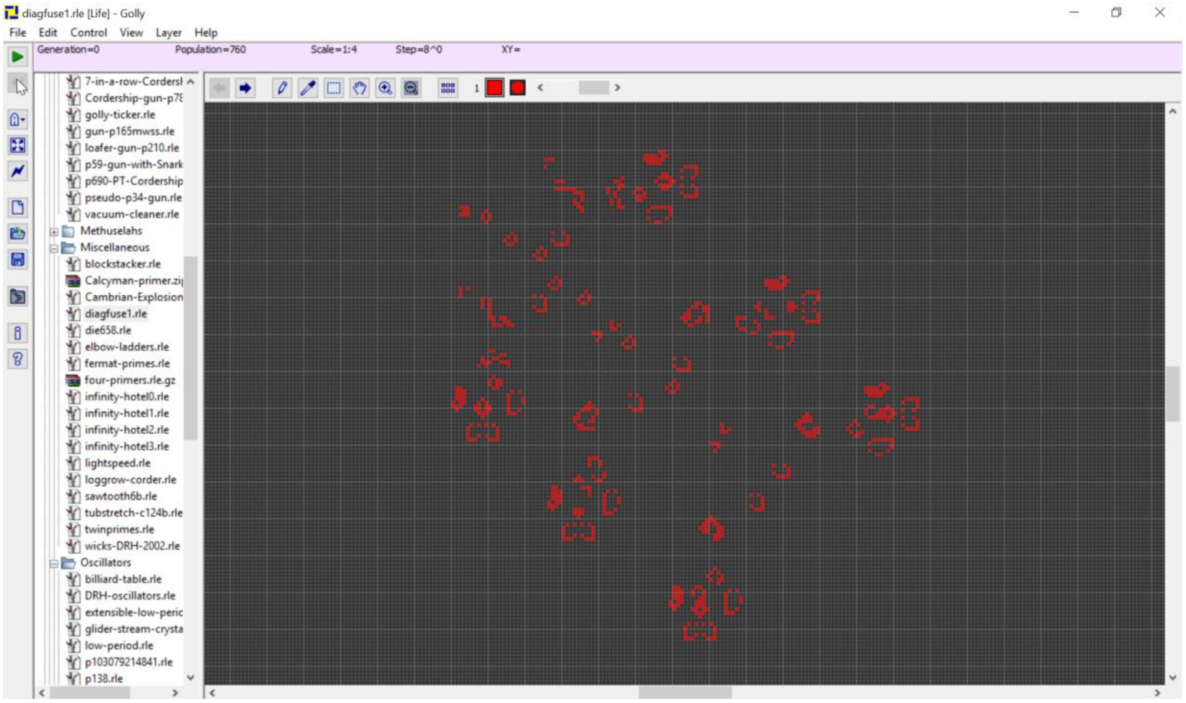

Figura 6. DiagFuse Generación 0.

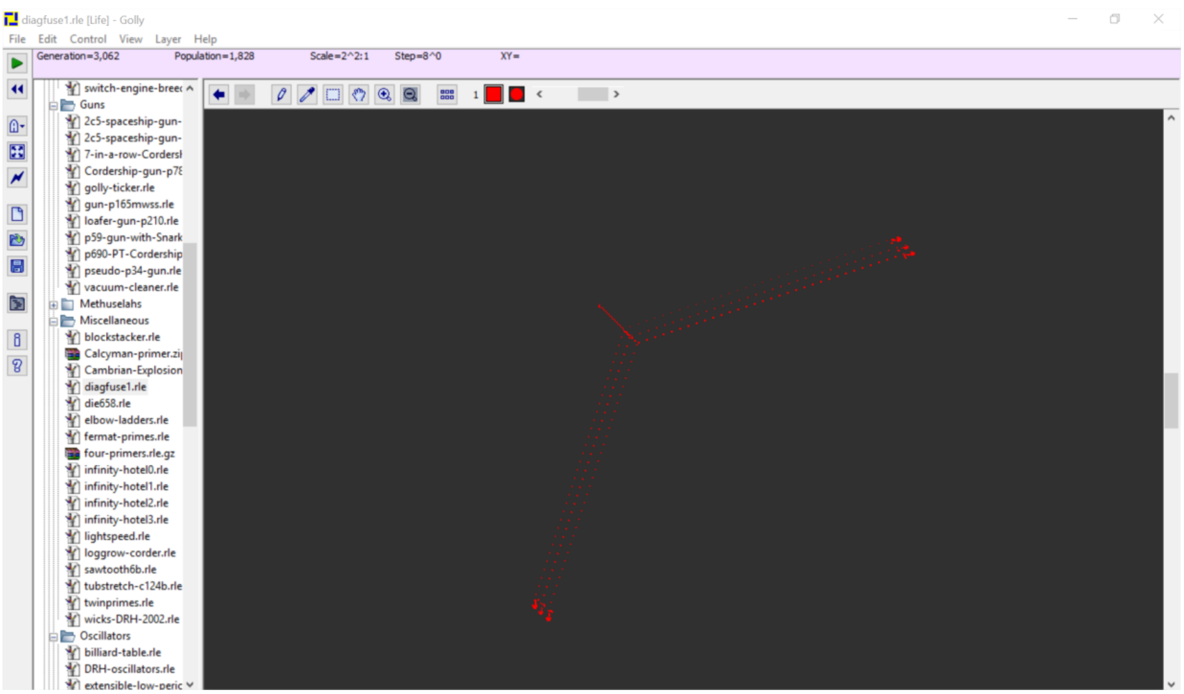

Figura 7. DiagFuse Generación 3062.

El tercer patrón que escogí fue “Die658”. Este fue más sencillo de entender, incluso, creo que desde el nombre se sabe de que trata. En la generación 0, se muestra el siguiente patrón:

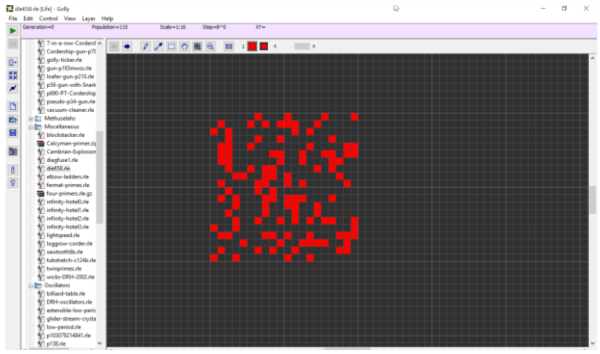

Figura 8. Die658 Generación 0.

El comportamiento de este patrón consiste en que después de 658 generaciones, todas las células habrán muerto.

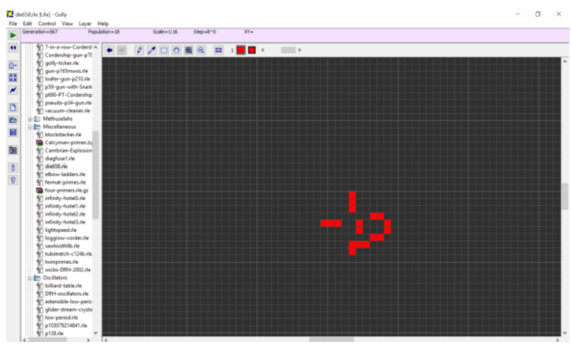

Figura 9. Die658 Generación 567.

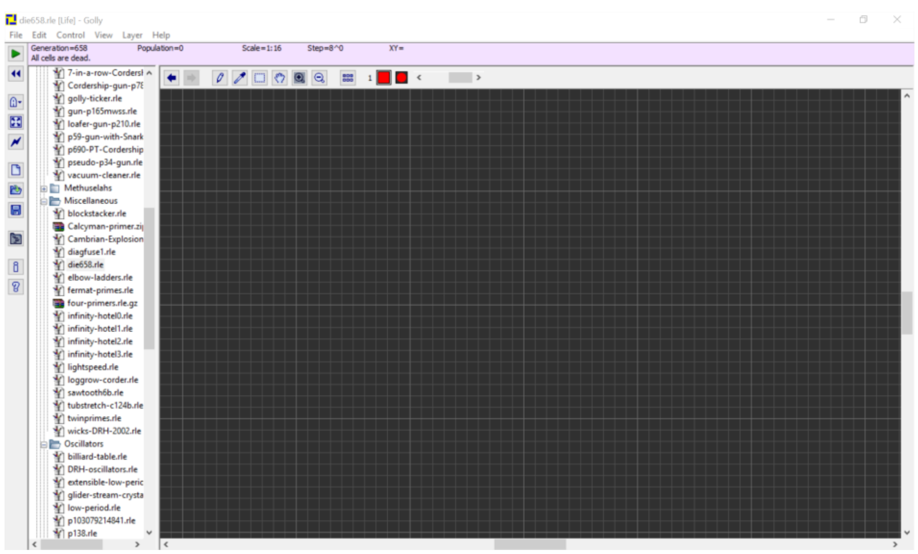

Figura 10. Die658 Generación 658 (todas las células muertas).

El cuarto patrón que escogí fue "Eaters-Misc" (figura 11), en este patrón al inicio se tienen varias vidas estáticas y al lado de cada una de estas, otro tipo de elementos. Lo que sucede en este patrón es que la vida estática simula como si la vida estática se alimentara del otro elemente que está a su lado, cada una de estas termina de "alimentarse" en generaciones distintas, y al final del experimento, solo quedan las vidas estáticas. El resultado se muestra en la figura 12.

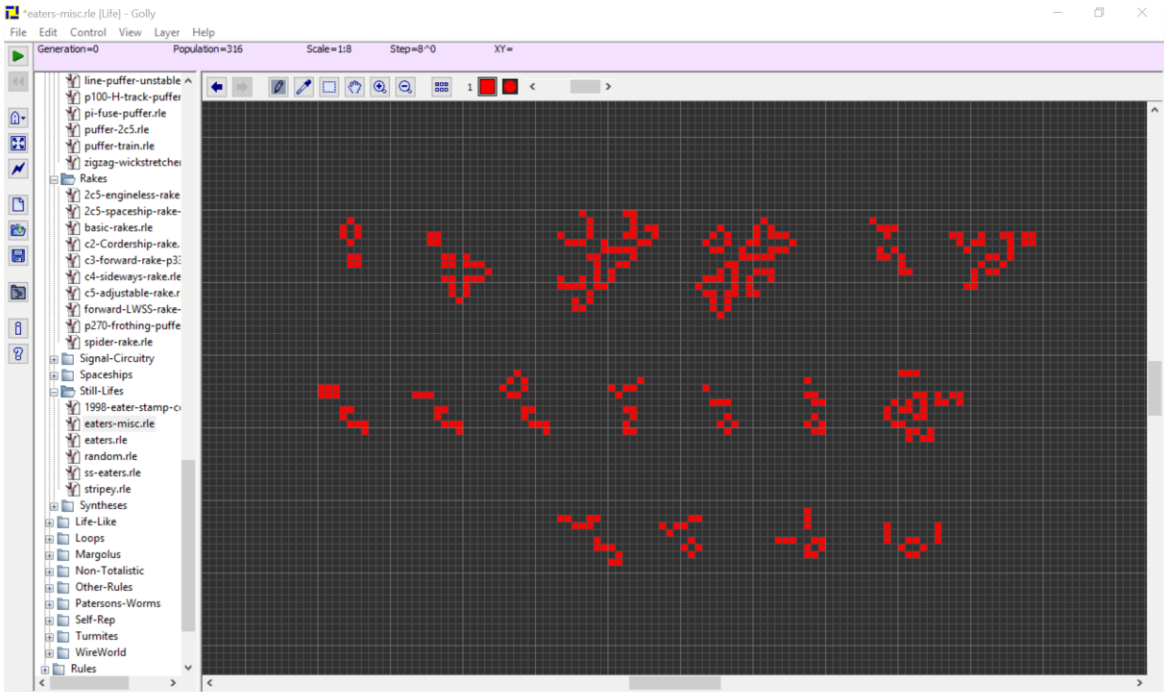

Figura 11. Eaters-Misc Generación 0.

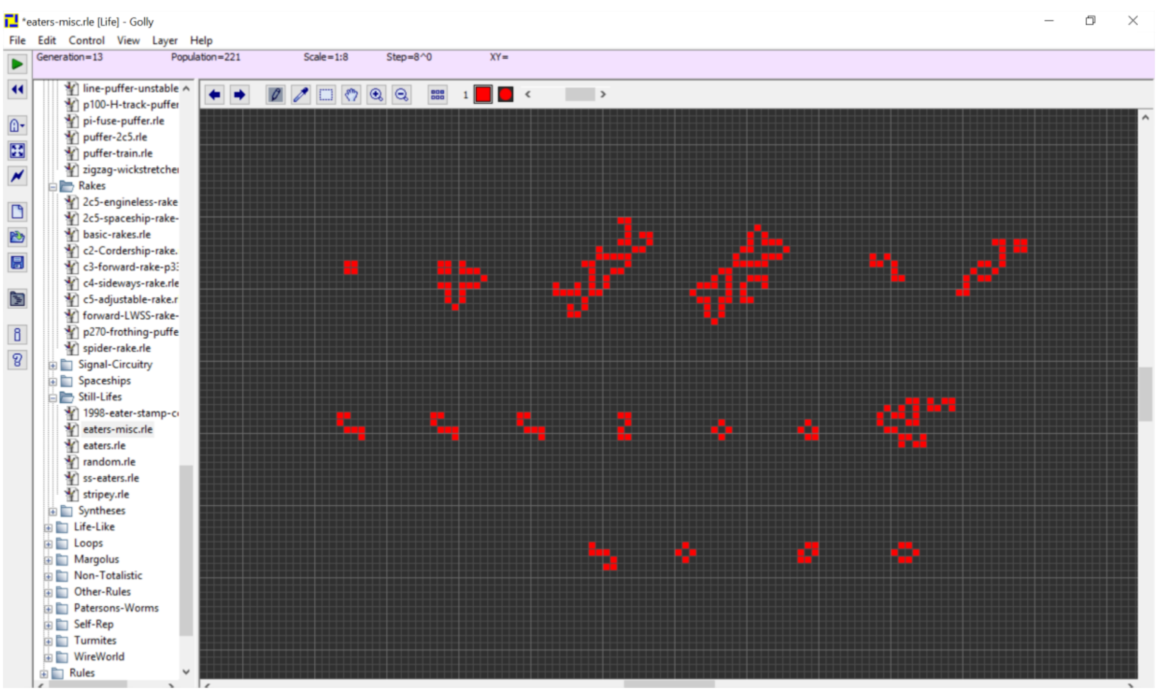

Figura 12. Eaters-Misc Generación 13 (todas las vidas estáticas terminan de "alimentarse").

El último patrón que tomé fue “Oscillator-Synthesis” (figura 13), el cual al inicio cuenta al parecer únicamente con deslizadores y una especie de frontera de células tipo rejilla (la cual desaparece en la primera generación). Conforme aumenta el número de generaciones, estos gliders al encontrarse, dan origen a osciladores, llegando a un punto (al parecer, en la generación 102), en el que todos los elementos son osciladores como se muestra en la figura 14.

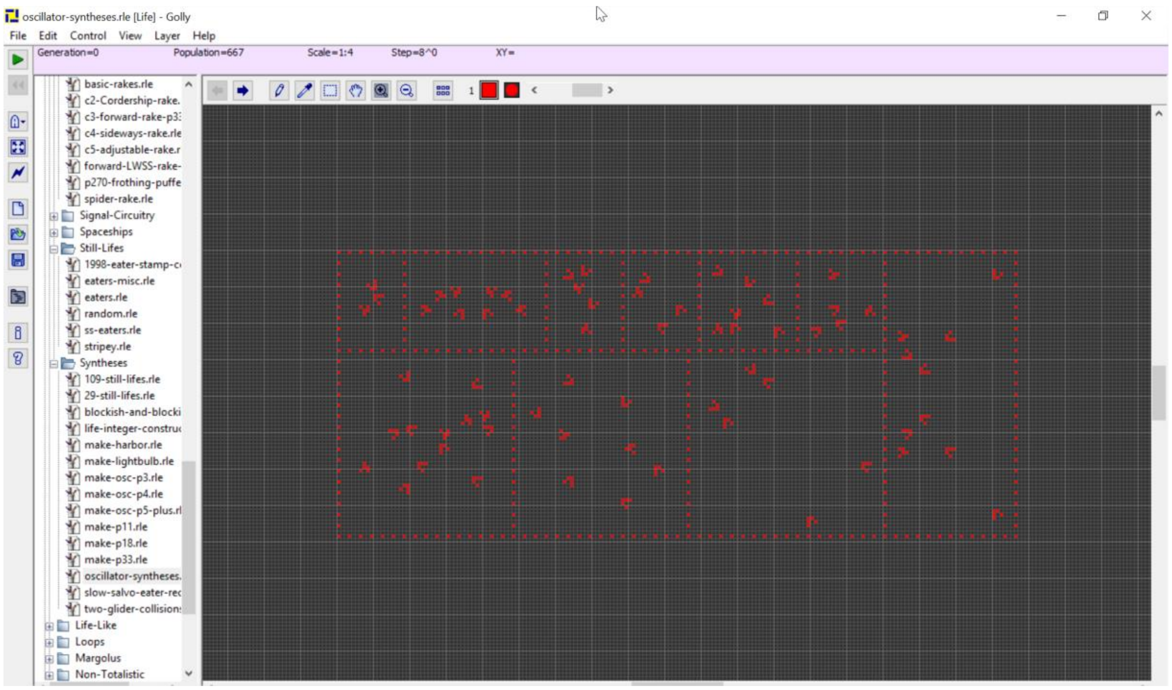

Figura 13. Oscillator-Synthesis Generación 0.

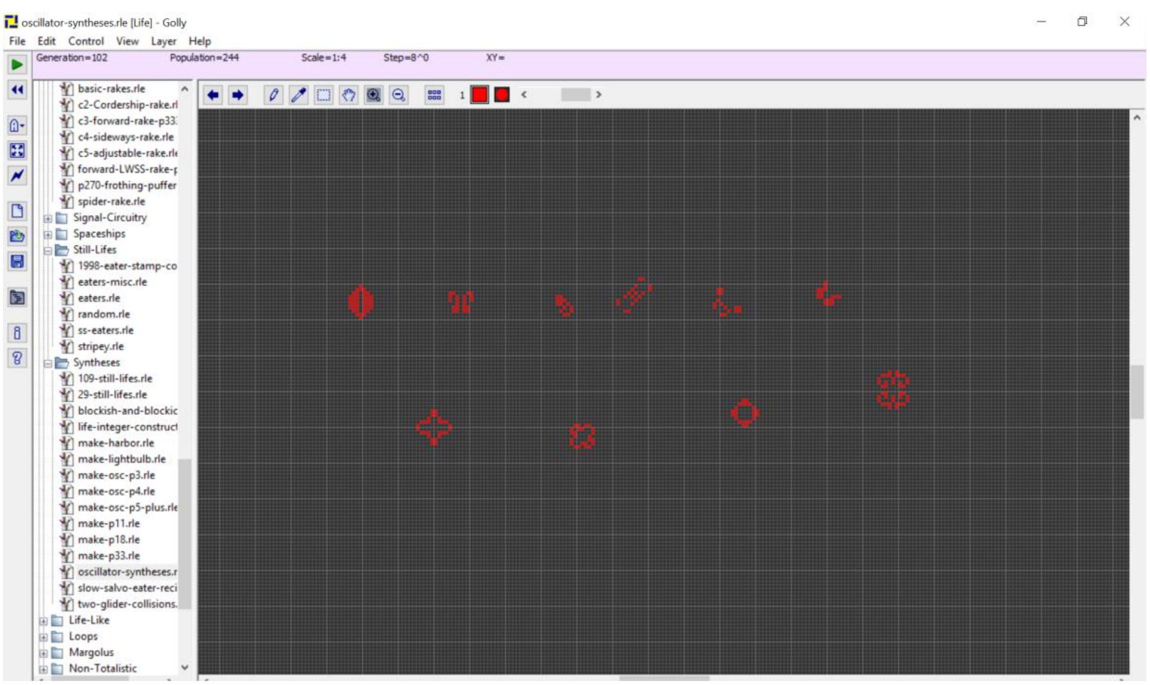

Figura 14. Oscillator-Synthesis Generación 102.

El siguiente experimento consistió en encontrar un deslizador, una vida estática y un oscilador dada la regla Bx/Syz, donde x, y, z son los últimos 3 dígitos de nuestra boleta respectivamente. Para mi caso fue B2/S82. Los resultados se muestran en las figuras 15, 16 y 17.

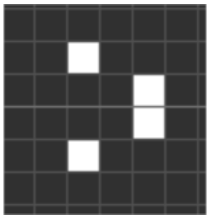

Figura 15. Deslizador con la regla B2/S82

Figura 16. Vida estática con la regla B2/S82:S3

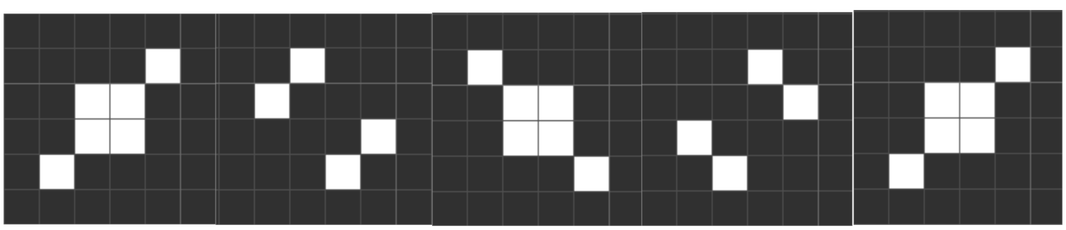

Figura 17. Oscilador con la regla B2/S82

La razón del por qué no se encontró una vida estática en una cuadrícula sin fronteras se dará en la conclusión.

El siguiente experimento fue establecer la regla W22 que se muestra en la figura 18.

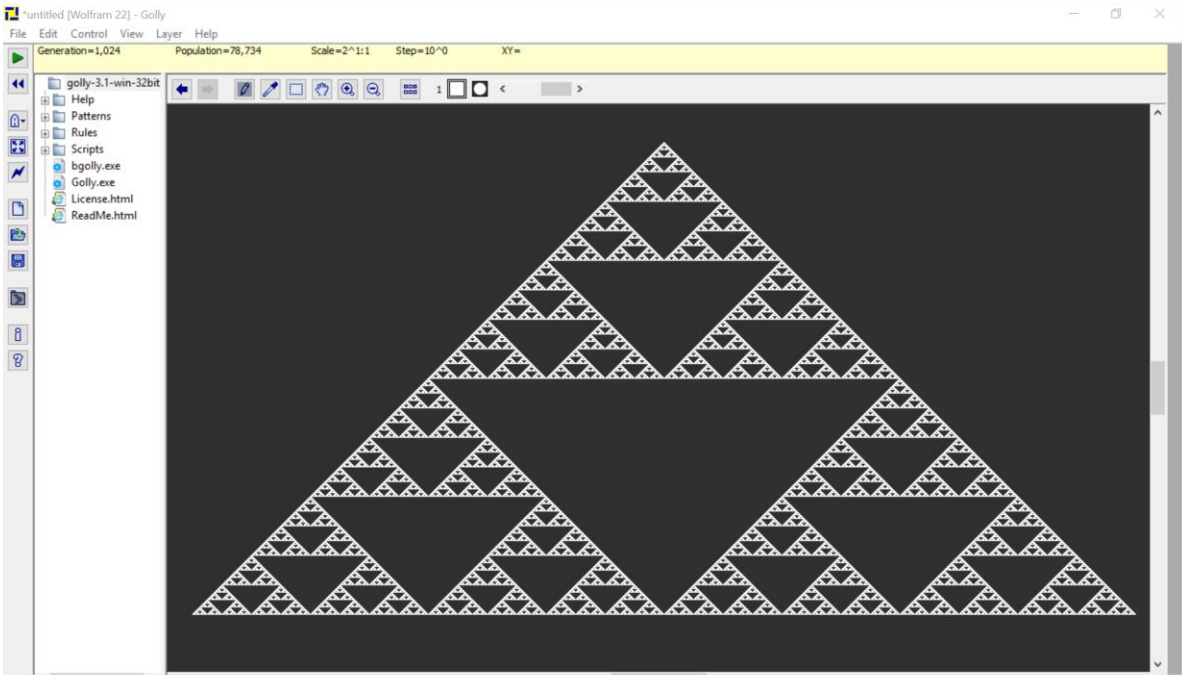

Figura 18. W22 Generación 1024.

El último experimento fue encontrar una regla que formara un patrón semejante al tablero de ajedrez. Para este caso se utilizó la regla 250. Como se muestra en la figura 19.

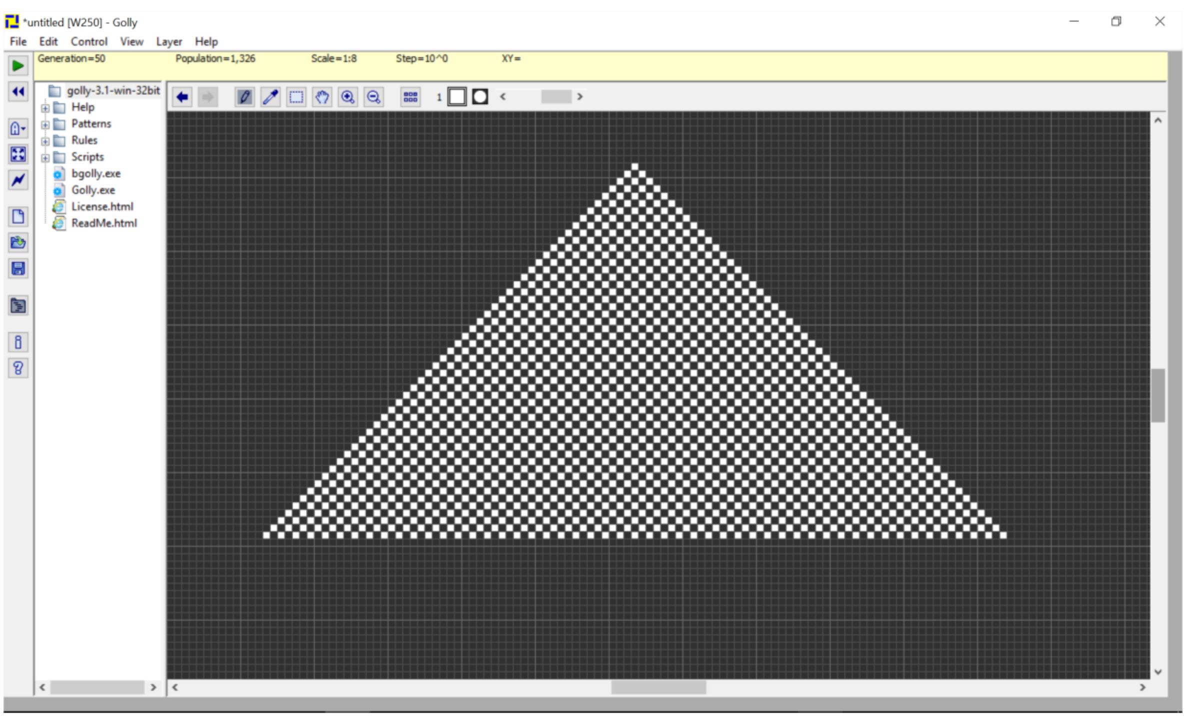

Figura 19. W250 generación50.

CONCLUSIONES.

Si bien, el comportamiento de los autómatas celulares puede ser muy interesante, hay casos en los que resulta demasiado caótico y analizarlo no es una tarea fácil. Como en mi caso, al tratar de buscar una vida estática en la regla que definí, resultaba casi imposible, lo que me dio una mayor complicación, fue que sobrevivían las células con 2 u 8 vecinos vivos, pero a su vez, nacía una nueva célula si la célula no viva tenía dos vecinos vivos. Eso resultó en que era muy difícil encontrar un patrón donde todas las células tuvieran 8 vecinos vivos, por lo que, para ello, utilicé un sufijo en la definición de la regla para definir una rejilla limitada de 3x3 y de tipo esfera. Y la vida estática resultante, fue llenar toda la rejilla, así, todas las células tendían 8 vecinos vivos y por ende sobreviven, ya que, debido al tipo de rejilla, si una célula se encuentra en un extremo, sus vecinos son los que la rodeen en ese extremo, y los que colindarían con ella si uniéramos los otros extremos con ella. Dando lugar así a una esfera.

Las aplicaciones de este tipo de autómatas me resultan fascinantes, desde simular colisiones, hasta la simulación de epidemias, virus, entre otras más. Sin duda, el simulas sistemas biológicos o que se encuentren en la naturaleza, muchos años atrás resultaba casi imposible, pero con este tipo de técnicas, cada vez nos acercamos más a poder simularlos y representarlos en una forma más completa mediante sistemas computacionales y digitales.

Referencias

[1] E. N. Lorenz, La esencia del caos, Barcelona: Pensamiento, 1995.

[2] R. Schmitz, Use of chaotic dynamical systems in cryptography, J. Franklin Instritute, 2001.

[3] D. A. Reyes Gómez, «CINVESTAV,» 25 Agosto 2011. [En línea]. Available: http://delta.cs.cinvestav.mx/~mcintosh/cellularautomata/Summer_Research_files/Arti_Ver_Inv_2011_DARG.pdf.