



---

**72.25 - Simulación de Sistemas**  
**Trabajo Práctico Nro. 2**  
**Autómatas Celulares - “Off-Lattice”**

---

*Fecha de entrega: 1/09/2023*  
*Grupo 2*

*Autores:*  
Chao, Florencia - 60755  
Cornidez, Milagros - 61432  
De Luca, Juan Manuel - 60103

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Sistema Real . . . . .	1
1.2. Modelo matemático . . . . .	1
<b>2. Implementación</b>	<b>2</b>
2.1. Estructura del proyecto . . . . .	2
2.2. Algoritmo . . . . .	3
<b>3. Simulaciones</b>	<b>3</b>
<b>4. Resultados</b>	<b>4</b>
4.1. Variación de la densidad . . . . .	4
4.2. Variación de la amplitud del ruido . . . . .	5
4.3. Variación de diferentes parámetros . . . . .	7
<b>5. Conclusiones</b>	<b>11</b>

## 1. Introducción

Este trabajo consistió en implementar el algoritmo “Off-Lattice” con el objetivo de analizar el comportamiento de bandadas de agentes autopropulsados. En este algoritmo, cada partícula se representa como un punto y se mueve de manera continua en una grilla de tamaño  $L$ , a velocidad constante. En cada paso temporal las partículas ajustan su dirección de movimiento promediando las direcciones de sus vecinas, lo que lleva a un comportamiento conjunto unificado en una dirección específica.

Para introducir variabilidad en este comportamiento, se incorpora un factor de ruido que induce cierta inestabilidad en la coordinación de direcciones de los agentes. La magnitud de este ruido y la densidad de partículas en la grilla de simulación influyen en la estabilidad de la coordinación direccional.

El informe subsiguiente tiene como finalidad detallar y analizar la implementación del modelo “Off-Lattice”, que se enfoca en representar el comportamiento de sistemas de partículas autopropulsadas.

A lo largo del informe se describe en profundidad el modelo propuesto, su implementación en el contexto “Off-Lattice”, las simulaciones realizadas para experimentar con diferentes parámetros y situaciones y, finalmente se presenta un análisis de los resultados obtenidos, respaldado por diversos gráficos.

### 1.1. Sistema Real

El modelo intenta imitar cómo las partículas interactúan con sus vecinas, en ocasiones dando lugar a fenómenos de agrupamiento que pueden ser observados en sistemas biológicos reales, como el vuelo en formación de aves, el enjambre de insectos, los cardúmenes de peces, entre otros.

### 1.2. Modelo matemático

El modelo se encuentra basado en dos ecuaciones, tanto para la evolución de la dirección como para la evolución de la posición de las partículas.

La forma para la actualización de la posición, se utiliza la ecuación (1):

$$x_i(t+1) = x_i(t) + v_i(t)\Delta t \quad (1)$$

Mientras que, para la actualización de la dirección, se utiliza la ecuación (2)

$$\theta(t+1) = \langle \theta(t) \rangle_r + \Delta\theta \quad (2)$$

Cabe considerar que  $\Delta\theta$  es un ruido uniforme, tal que  $\Delta\theta \in [-\eta/2; \eta/2]$ , y que  $\langle \theta(t) \rangle_r$  es el promedio de los ángulos, incluyendo la partícula  $r$ . Para realizar dicho promedio, se utiliza la ecuación (3)

$$\langle \theta(t) \rangle_r = \arctan[\langle \sin(\theta(t)) \rangle_r / \langle \cos(\theta(t)) \rangle_r] \quad (3)$$

## 2. Implementación

### 2.1. Estructura del proyecto

Para la realización del trabajo, se utilizó el lenguaje de programación *Java*. Para llevar a cabo el modelo, se generaron 4 clases: *Particle*, *Cell*, *Grid* y *Off-Lattice*. La clase *Particle* cuenta con toda la lógica que necesita una partícula individual. Es decir, su posición, velocidad y ángulo. Por otro lado, la clase *Cell* cuenta con una lista de las partículas que se encuentran en dicha celda. Luego, la clase *Grid* posee toda la lógica de la grilla y como encontrar los vecinos para cada celda. Por último, la clase *Off-Lattice* que se encarga de la lectura y la generación de los archivos, y se encarga de la lógica del modelo matemático en el que se encuentra basado el sistema desarrollado. En la Figura 1 se observa un diagrama UML de la organización de dichas clases, los métodos, y las variables que contiene cada una:

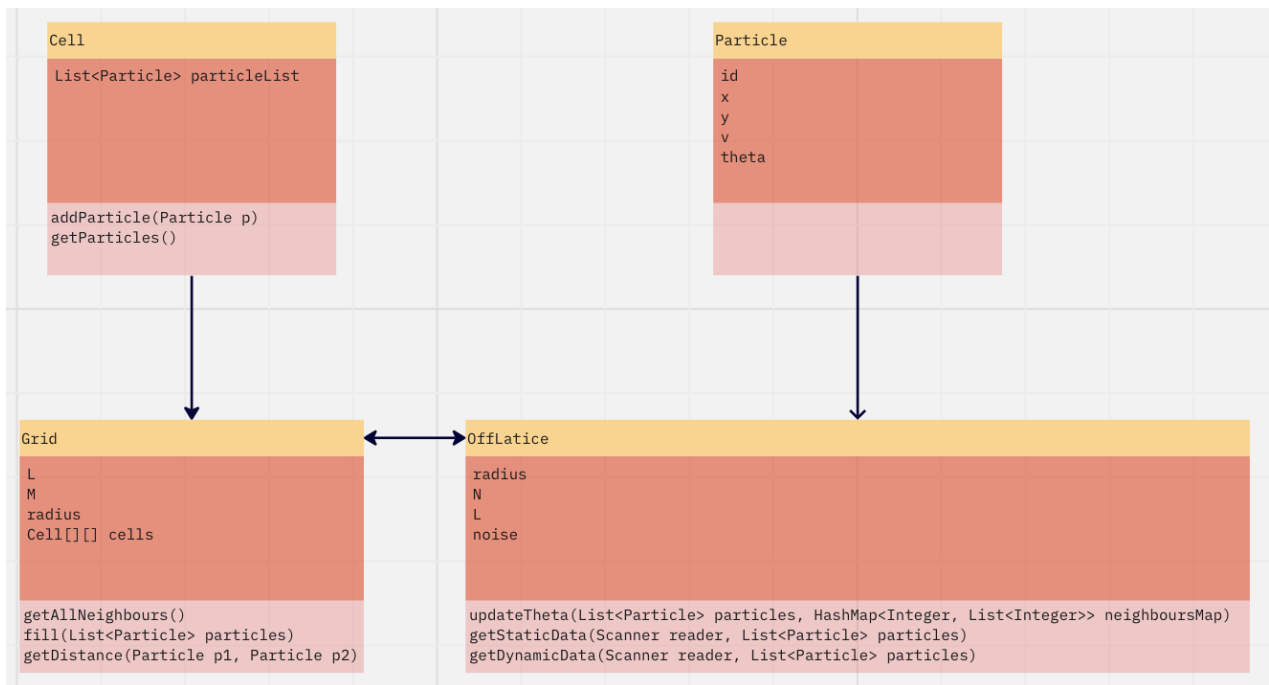


Figura 1: Diagrama UML del proyecto

## 2.2. Algoritmo

El comportamiento del sistema puede describirse, en pseudocódigo, por el algoritmo 1:

---

**Algoritmo 1** Algoritmo de comportamiento del sistema
 

---

```

1: grilla ← CrearGrilla()
2: for iteracion = 1, 2, ... do
3:   grilla.llenar()
4:   grilla.obtenerVecinos()
5:   calcularNuevaPosicion()
6:   calcularNuevaDireccion()
7:   actualizarPosiciones()
8:   actualizarDirecciones()
9: end for

```

---

## 3. Simulaciones

Para llevar a cabo las simulaciones, es necesario definir el parámetro de orden ( $v_a$ ). Dicho parámetro tiende a cero para total desorden y a 1 para partículas “polarizadas”, y queda definido por la ecuación (4):

$$v_a = \frac{1}{Nv} \left| \sum_{i=1}^N v_i \right| \quad (4)$$

Al momento de realizar las simulaciones, se utilizaron las siguientes condiciones iniciales:  $v = 0,03$  y  $r = 1$ , lo que quiere decir que la velocidad y el radio de las partículas se mantuvieron constantes a lo largo de las simulaciones. Asimismo, cabe destacar, que en el instante  $t_0$  se establecen de manera aleatoria las coordenadas iniciales de las partículas y que todas las simulaciones fueron realizadas con 1000 iteraciones.

Por otro lado, es válido mencionar que al haber establecido  $v_a$  y  $r$  como condiciones fijas a lo largo de las simulaciones, se buscó variar el ruido  $\eta$  para analizar su efecto sobre el orden  $v_a$ . Para realizar dicho análisis, se decidió tomar una densidad constante  $\rho = 20$ ,  $N=500$  y  $L = 5$ , mientras que el ruido fue variado entre los valores 0 y 5.

En otra simulación, se buscó variar la densidad  $\rho$  para analizar su efecto sobre el orden  $v_a$ . Para realizar el análisis, se decidió variar  $L$  entre 5 y 25; dejando fijo el ruido en 0.5 y  $N = 500$ .

Finalmente, se decidió analizar el comportamiento del sistema ante una baja densidad  $\rho$  y variar el ruido  $\eta$ . Para llevar a cabo dicho análisis, se decidió variar el ruido entre 0 y 5, y tomar una densidad constante  $\rho = 1,25$ ,  $N=500$  y  $L = 20$ .

## 4. Resultados

### 4.1. Variación de la densidad

Para analizar cómo afecta la densidad en el movimiento de las partículas, se realizó la comparación de los valores de orden  $v_a$  en función de las iteraciones. Dicha comparación puede ser observada en la Figura 2. Luego, se calculó el promedio de los valores obtenidos con sus respectivos errores, los cuáles se pueden ver en la Figura 3:

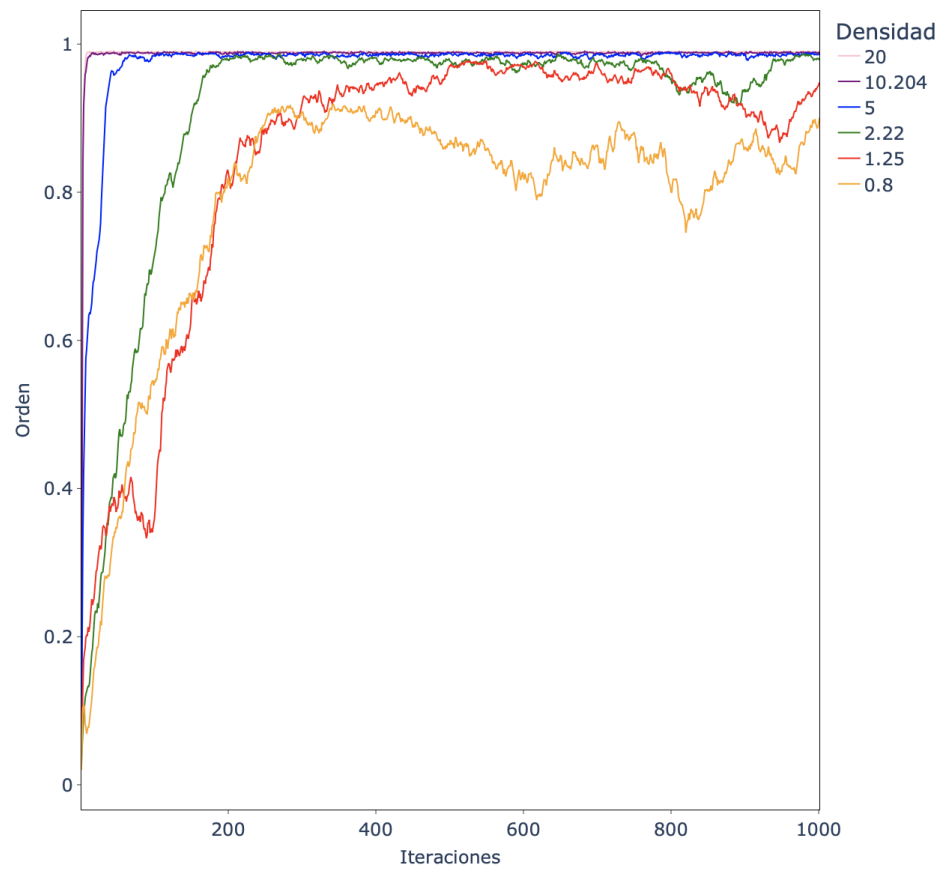


Figura 2:  $v_a$  en función de las iteraciones, para distintos  $\rho$  con  $\eta = 0,5$ .

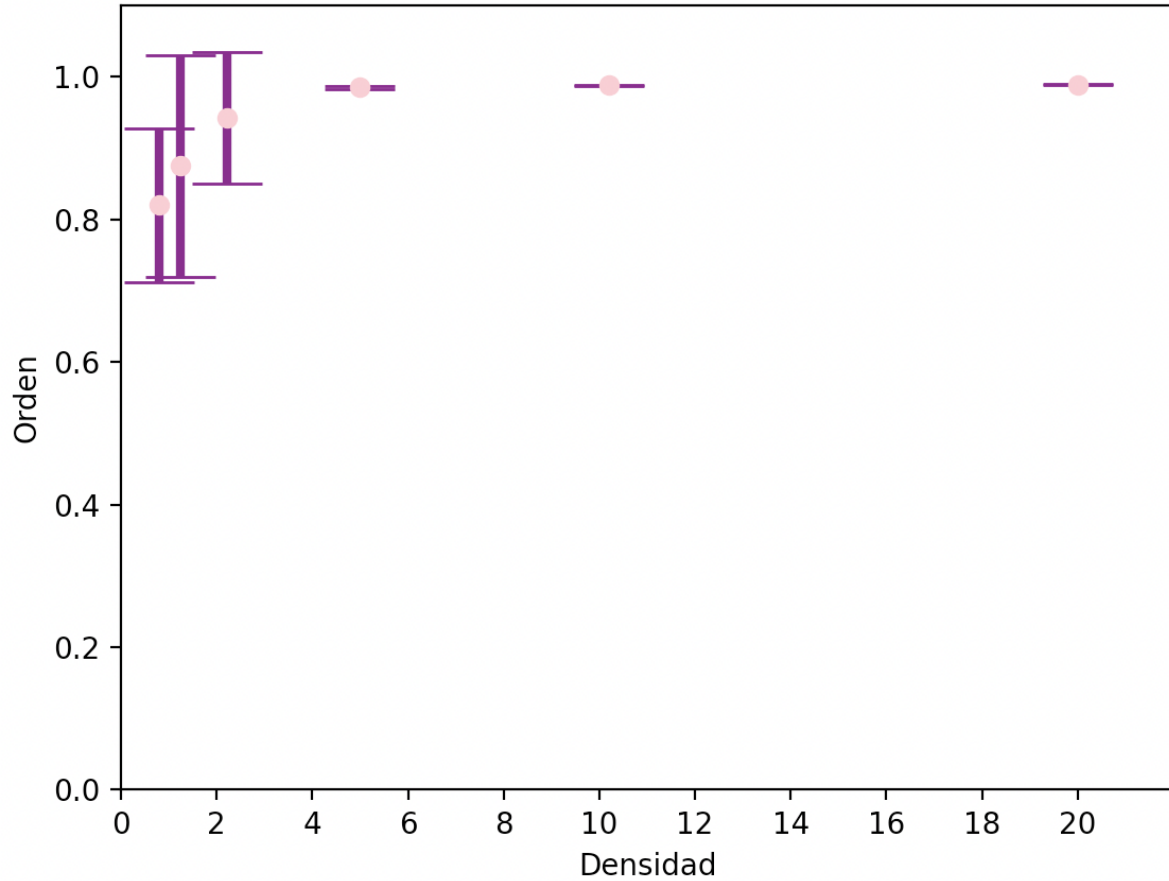


Figura 3:  $v_a$  en función de  $\rho$  con  $\eta = 0,5$ .

En la Figura 2 se puede observar que para altos valores de densidad ( $\rho \geq 5$ ), el valor del orden  $v_a$  tiende a 1 y tiende a ser estable, es decir que las partículas alcanzan un estado de polarización. A medida que la densidad es menor, las partículas no alcanzan el estado mencionado anteriormente y, además, no son estables. También se puede observar que, a mayor densidad, el orden es más cercano a 1.

En la Figura 3, se puede observar que los errores reflejan lo argumentado anteriormente, y que ante una menor densidad, el desorden es mayor, al igual, que la inestabilidad de las partículas.

## 4.2. Variación de la amplitud del ruido

Para analizar cómo afecta el ruido en el movimiento de las partículas, se realizó la comparación de los valores de orden  $v_a$  en función de las iteraciones, cuyo resultado reside en la Figura 4, y luego se calculó el promedio de los valores obtenidos con sus respectivos errores (Figura 5):

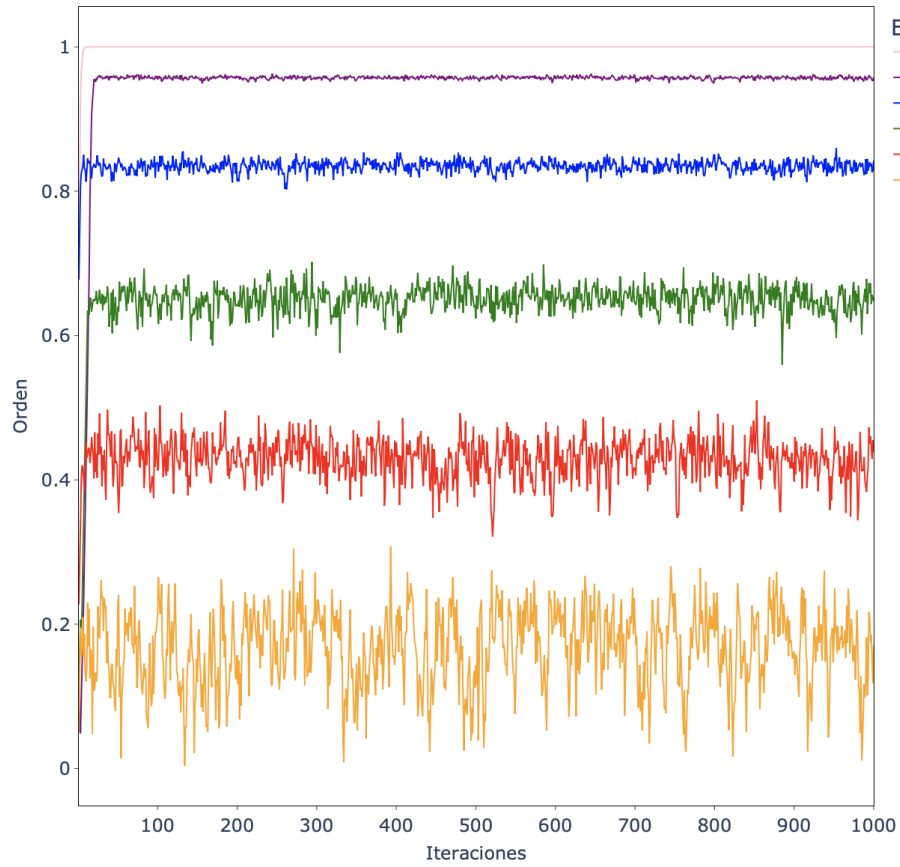


Figura 4:  $v_a$  en función de las iteraciones, para distintos  $\eta$  con  $N=500$  y  $L=5$ .



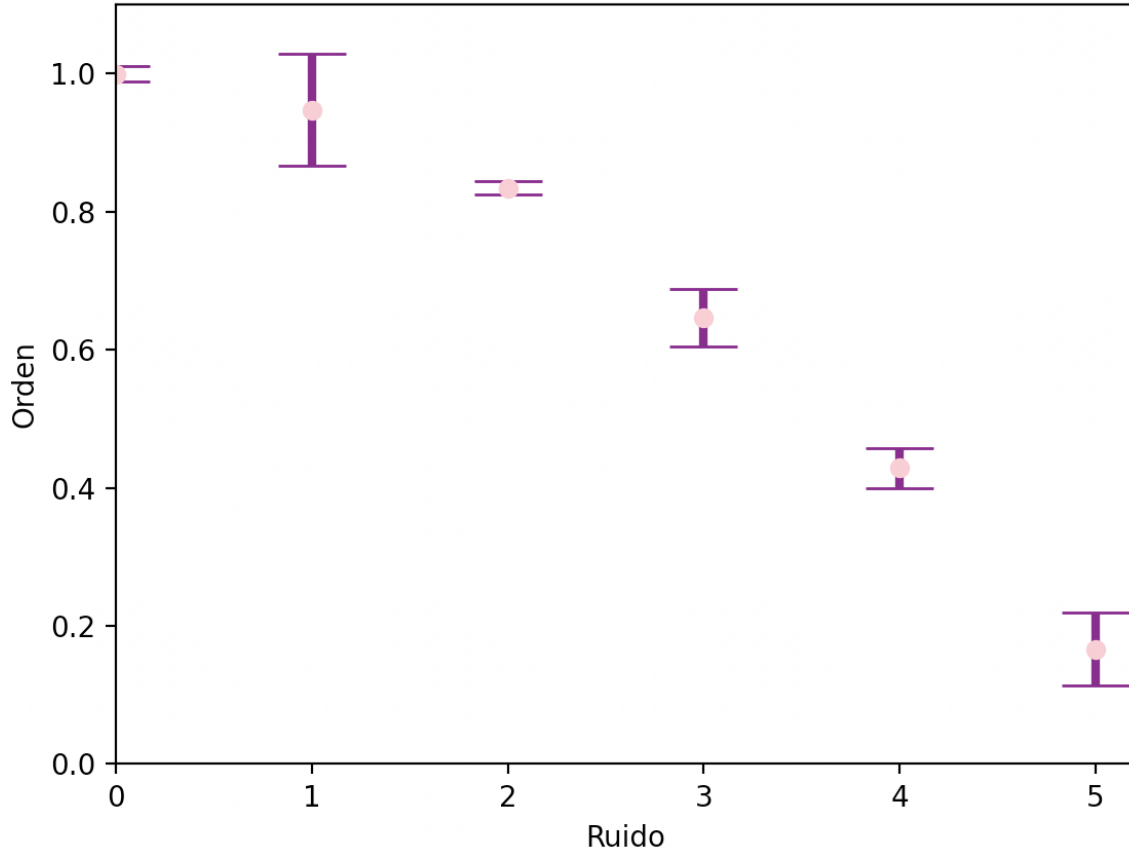


Figura 5:  $v_a$  en función de  $\eta$  con  $N=500$  y  $L=5$ .

En la Figura 4, se puede observar que para los valores de ruido  $\eta \leq 1$ , el valor del desorden tiende a 1 y es relativamente estable, por lo que las partículas alcanzan la polarización. A partir de mayores valores de ruido, el orden no tiende a 1 y es inestable. También, se puede observar, que cuanto menor es el ruido, mayor es el orden y más se acerca a 1.

Por su parte, en la Figura 5, se puede observar que los errores reflejan lo argumentado anteriormente y que, ante un mayor ruido, el desorden es mayor, al igual que la inestabilidad de las partículas. Cabe destacar que, al tomar todas las mediciones de las partículas, con el ruido en  $\eta = 1$ , alcanzar el estado estacionario significa una mayor cantidad de iteraciones, por lo que la amplitud del error es mayor debido a esto.

### 4.3. Variación de diferentes parámetros

Por último, se decidió analizar el comportamiento del sistema ante una densidad baja, y diferentes ruidos. Para esto, se realizó la comparación de los valores de orden  $v_a$  en función de las iteraciones para analizar como afecta el ruido en el movimiento de las partículas (Figura 6):

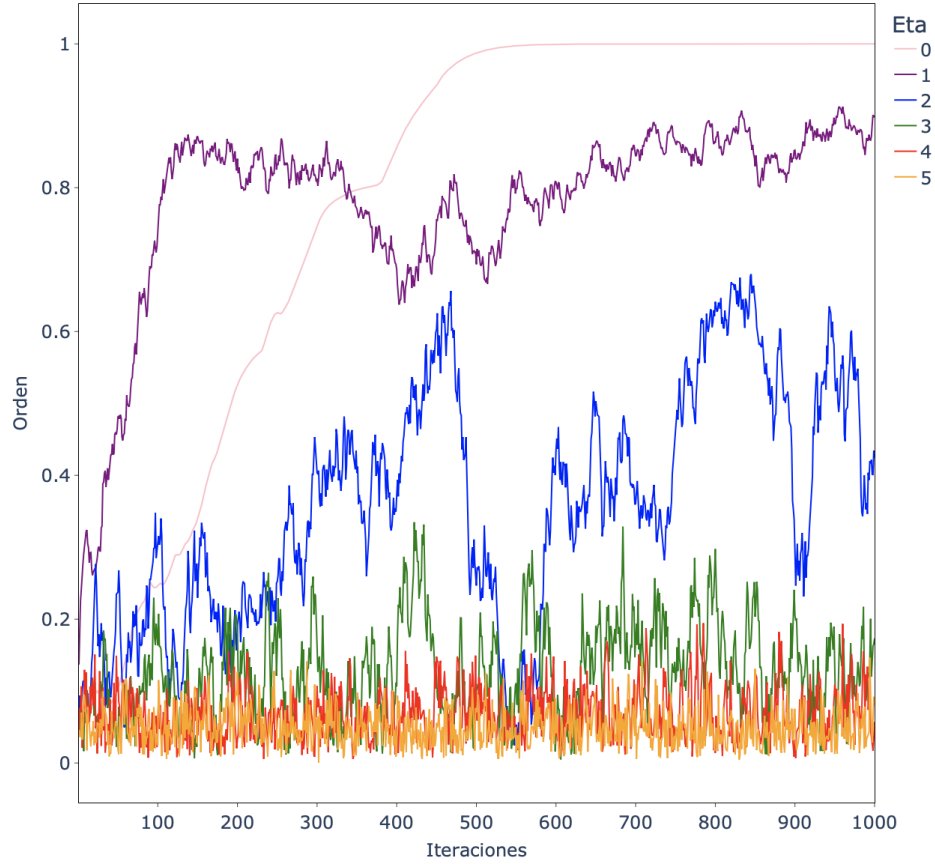


Figura 6:  $v_a$  en función de las iteraciones, para distintos  $\eta$  con  $N=500$  y  $L=20$ .

En la Figura 6, se puede observar que, con una baja densidad, los errores afectan de gran manera la estabilidad y el orden de las partículas. No solo encuentra el estado estacionario cuando  $\eta = 0$ , sino que lo encuentra en una elevada cantidad de iteraciones.

Para el resto de los ruidos, las partículas muestran una gran inestabilidad, y no alcanzan un estado de polarización ya que su orden se aleja de 1.

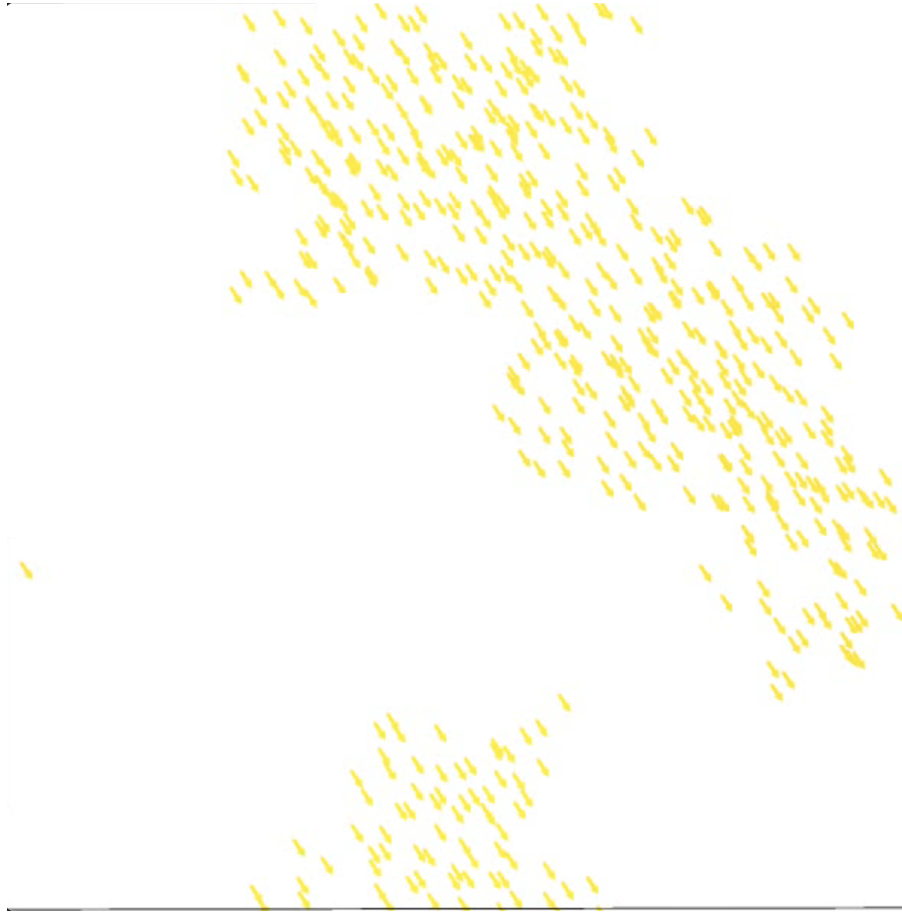


Figura 7: Captura de simulación con  $N=500$ ,  $L=20$  y  $\eta = 0$ .

En la Figura 7, se puede observar como ante una baja densidad y un bajo ruido se alcanza de igual manera el estado estacionario y se produce una polarización aunque en una mayor cantidad de iteraciones.

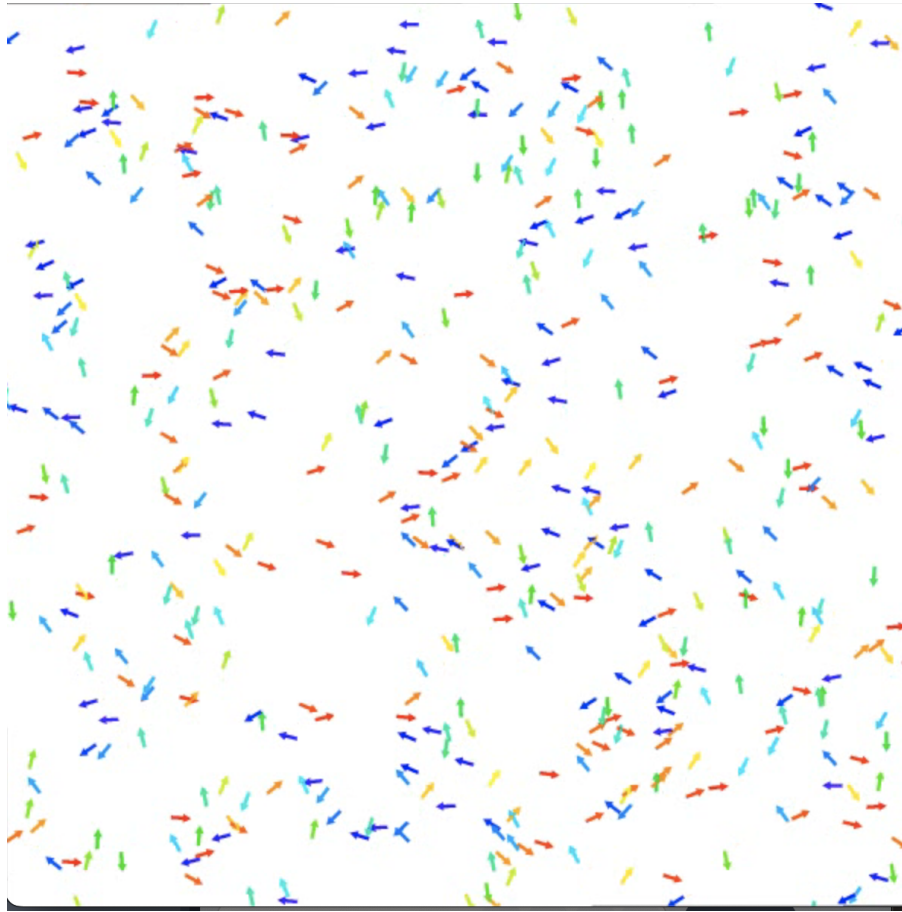


Figura 8: Captura de simulación con  $N=500$ ,  $L=20$  y  $\eta = 5$ .

En la Figura 8, se puede observar como ante una baja densidad y un gran ruido, no se alcanza un estado estacionario y se produce un gran desorden.

## 5. Conclusiones

A partir de los resultados obtenidos a partir del análisis de los parámetros más relevantes de las simulaciones, pudimos obtener las siguientes conclusiones:

Al analizar la variación de la amplitud del ruido, se pudo observar que ante una densidad constante elevada ( $\rho = 20$ ), las partículas logran alinear su dirección cuando el ruido es pequeño ( $\eta \leq 1$ ); es decir, que se encuentran polarizadas. Cuando el ruido es mayor a 1, se puede observar que el sistema mantiene un desorden total, en el cual, las partículas se mueven en diferentes direcciones.

Al analizar la variación de la densidad con un ruido pequeño ( $\eta = 0,5$ ), se puede observar que el sistema logra llegar a un estado estacionario; pero a medida que la densidad es menor, el tiempo en el que alcanza el estado estacionario aumenta.

Al analizar la variación de la amplitud del ruido con una densidad constante  $\rho = 1,25$ , se pudo observar que cuando el ruido  $\eta = 0$ , se alcanza una polarización de las partículas pero en aproximadamente 500 iteraciones. Mientras que, en el resto de los casos, el sistema se encuentra en desorden y no alcanza el estado estacionario.

En conclusión, al analizar el método Off-Lattice podemos llegar a las siguientes conclusiones:

- A menor ruido, aumenta el orden en el estado estacionario.
- A mayor densidad, más rápido se alcanza el estado estacionario.
- Cuanto menor es la densidad, mayor impacto tiene el ruido sobre el estado estacionario.
- A menor ruido y mayor densidad, se alcanza el estado estacionario en menor cantidad de iteraciones.