



## 72.25 - Simulación de Sistemas: Autómatas Celulares

Julián Francisco **Arce**; 60509  
Roberto José **Catalán**; 59174  
Gian Luca **Pecile**; 59235

2 de septiembre de 2022

# Índice

<b>1. Introducción</b>	<b>3</b>
1.1. Sistema Real . . . . .	3
1.2. Fundamentos . . . . .	3
<b>2. Implementación</b>	<b>3</b>
2.1. Arquitectura . . . . .	3
2.2. Algoritmo . . . . .	4
<b>3. Simulaciones</b>	<b>5</b>
3.1. Parámetros . . . . .	5
3.2. Observable . . . . .	5
<b>4. Resultados</b>	<b>5</b>
<b>5. Conclusiones</b>	<b>11</b>

# 1. Introducción

En el informe presente se detalla el análisis de la implementación computacional del modelo Off Lattice, que busca representar el comportamiento de sistemas de partículas autopropulsadas, tal como se describe en el paper *Novel Type of Phase Transition in a System of Self-Driven Particles* de Tamas Vicsek, Andras Czirok, Eshel Ben-Jacob, Inon Cohen y Ofer Shochet.

La implementación actúa como motor de la simulación para el modelo del que se definen variables observables que luego son analizadas cuantitativamente.

## 1.1. Sistema Real

El modelo planteado busca analizar la orientación de las partículas dependiendo de sus vecinos, produciendo un efecto de agrupamiento entre las mismas. En la naturaleza se pueden observar similitudes en el comportamiento con sistemas biológicos tales como bandadas de aves, rebaños de ovejas o enjambres de langostas.

## 1.2. Fundamentos

El sistema se basa en dos ecuaciones [1]:

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

Donde se tiene que  $x_i$  es una coordenada de una partícula determinada,  $t$  es un paso temporal y  $v_i$  indica la velocidad de la partícula.

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

Donde  $\langle \theta(t) \rangle$  es el promedio de los ángulos de todas las partículas dentro del radio de interacción  $r$ , mientras que  $\Delta\theta$  es un ruido uniforme que puede variar entre  $-\eta/2$  y  $\eta/2$ .

# 2. Implementación

La lógica de las simulaciones que implementa los algoritmos vistos en la teórica se realizó con el lenguaje Java, en su versión 17 en conjunto con Maven para el manejo de dependencias.

Se utilizan dependencias tales como OpenCsv para facilitar la exportación de archivos en formato csv. Lombok se utiliza como decorator para las clases de Java en pos de evitar boilerplate y mejorar los tiempos de desarrollo. Por último, para realizar testeos unitarios, se utiliza JUnit.

En cuanto al desarrollo se partió de la base realizada en el TP1: Cell Index Method donde se extienden sus funcionalidades. Se añade una clase por encima denominada OffLattice que se encarga de realizar la simulación utilizando las funcionalidades provistas por lo implementado anteriormente.

Además, se tienen dos programas principales. Uno es App, el cual permite correr una simulación dados dos archivos de entrada, static.csv y dynamic.csv. El otro, Generator, permite generar archivos de entrada para ser consumidos por App. Generator permite definir como parámetros la cantidad de partículas y el tamaño del espacio.

La simulación es realizada mediante el uso de threads concurrentes para mejorar el rendimiento y disminuir el tiempo de ejecución.

## 2.1. Arquitectura

A continuación se muestra el diagrama UML de la arquitectura optada para el desarrollo de la implementación realizada en la figura 1:

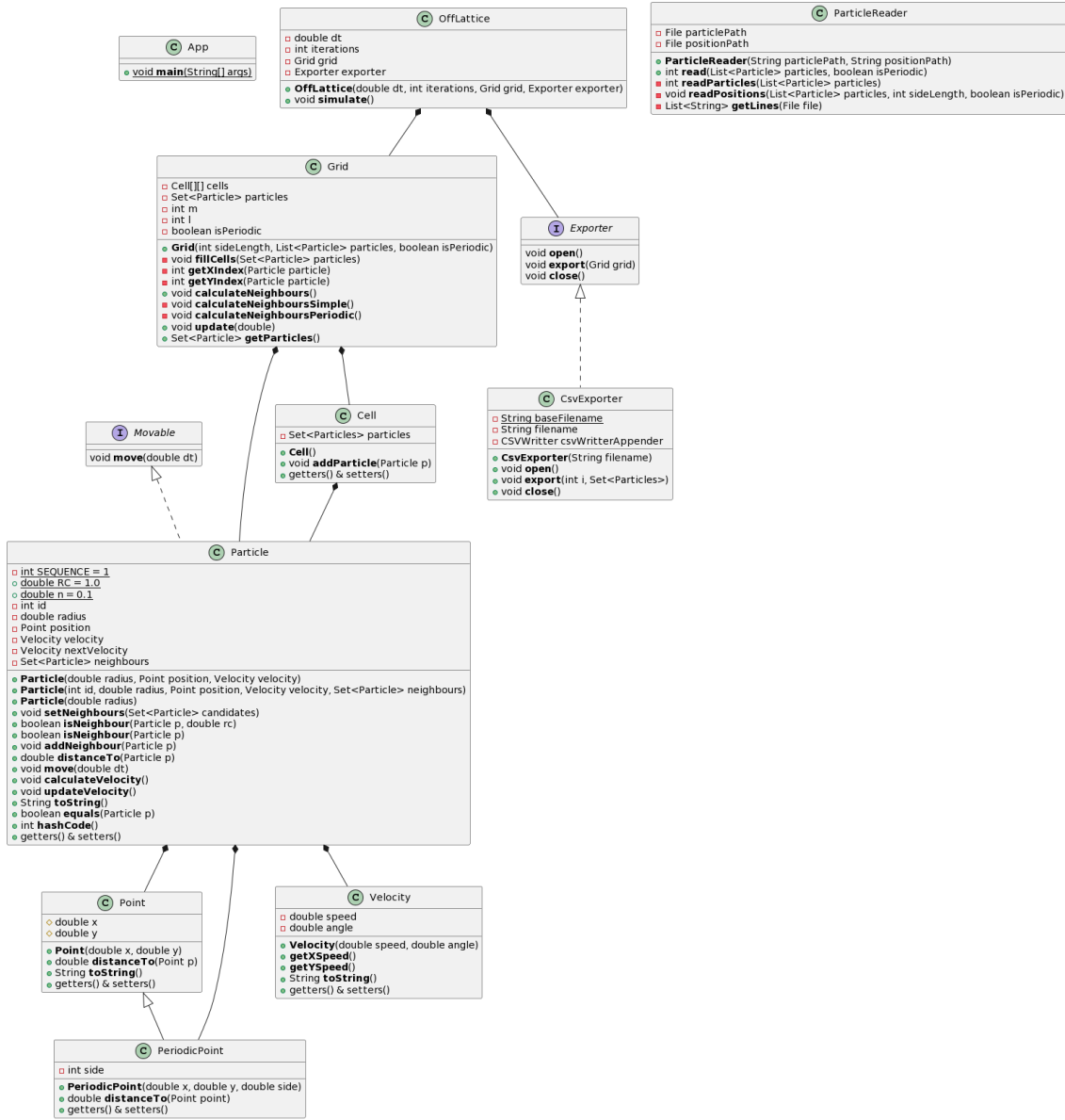


Figura 1: Diagrama UML.

A su vez, se realizaron testeos unitarios para los métodos de la clase de las partículas, al igual que los puntos (incluyendo periódicos) y velocidad.

## 2.2. Algoritmo

En la sección de fundamentos se explicita dos funciones para el sistema de simulaciones general pero en el caso de lo particular, para cada paso temporal se hace uso de la siguiente ecuación para el cálculo de  $\langle \theta(t) \rangle_r$ :

$$\text{atan2}(\langle \sin(\phi) \rangle_r, \langle \cos(\phi(t)) \rangle_r) \quad (3)$$

donde se tiene que el uso de *atan2* en vez de *atan* debido a que para cubrir todos los cuadrantes se tiene que tener un dominio entre  $[-\pi, \pi]$  que la segunda función no posee.

El algoritmo principal, visto como pseudo-código, es el siguiente:

En cada iteración:

Exportar posición y velocidad de las partículas en el instante  $t$ .

Actualizar las partículas para el instante  $t+1$ :

Calcular vecinos de cada partícula.

Mover cada partícula.

Calcular la nueva orientación de cada partícula.  
Actualizar la orientación de cada partícula.  
Reposicionar las partículas.

### 3. Simulaciones

#### 3.1. Parámetros

Para todas las simulaciones se decidieron mantener los siguientes parámetros fijos:  $r = 1,0$ ;  $v = 0,03$ ;  $t = 1$  y 1000 iteraciones. Donde los primeros parámetros son los hallados en el paper [1] y la cantidad de iteraciones -si bien no hay una definida por defecto- luego de analizar diversos valores, se concluyó que este valor es el apropiado para el análisis requerido del estado estacionario.

Por otro lado, los parámetros variados son la densidad y el ruido. En primer instancia se realizó una animación con densidad 15 y ruido 0.5, luego densidad 10 y ruido 2.0 y por último densidad 1 y ruido 0.1 seguido de un análisis de la evolución temporal y una comparación entre out vs input.

#### 3.2. Observable

El observable se define como parámetro de orden  $v_a$ . Donde se desarrolla como:

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

Donde  $N$  representa a la cantidad de partículas y  $v$  es el módulo de velocidad de las mismas. Además,  $v_a$  tiende a 0 cuando se encuentra en total desorden y, por el contrario tiende a 1 para partículas polarizadas.

Los promedios se calculan según el promedio aritmético y las barras de error de acuerdo al desvío estándar de la muestra.

### 4. Resultados

El primer caso de estudio posee la densidad más alta de los 3 analizados, y como se puede observar en la fig. 2, las partículas se encuentran altamente polarizadas luego de ciertas iteraciones.

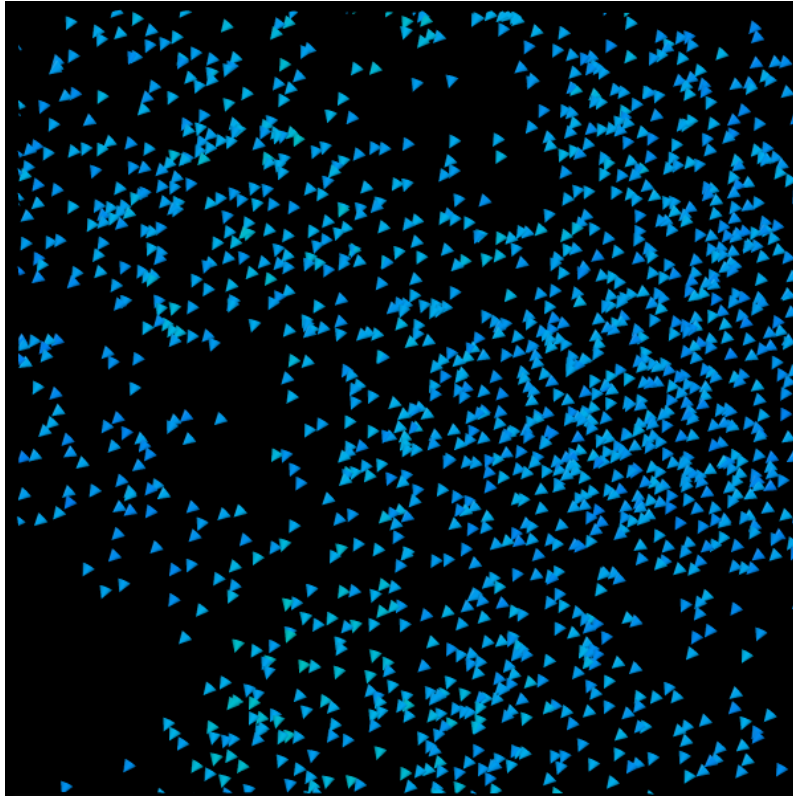


Figura 2: Fotograma de animación con 1500 partículas, densidad 15 y ruido 0.5.

Luego en la figura 3 se muestra cómo evoluciona el parámetro de orden a lo largo de las iteraciones. Se evidencia cierta tendencia a mantener una polarización constante hasta cierto nivel de ruido, es decir, la polarización se estabiliza a un intervalo acotado.

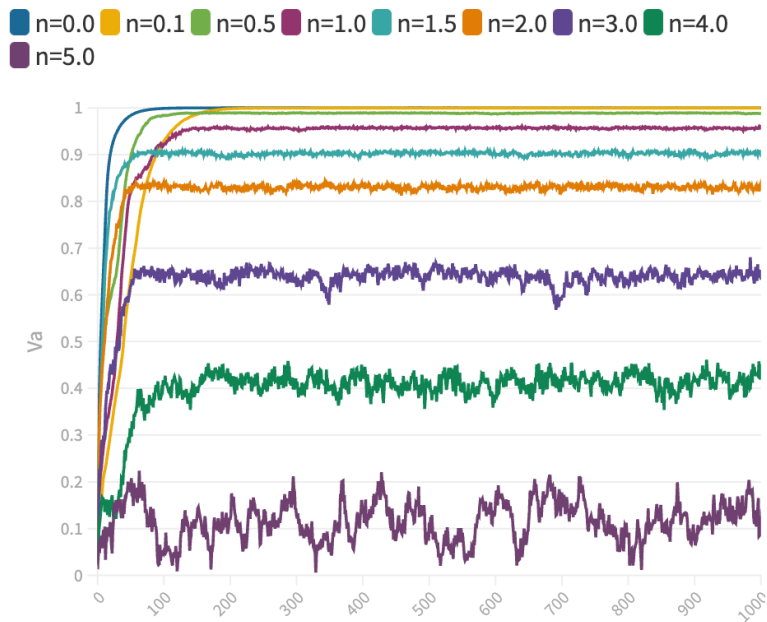


Figura 3:  $V_a$  en función de las iteraciones con 1500 partículas y densidad 15.

Por último, en la figura 4 se puede ver el valor promedio del parámetro de orden en el estado estacionario para distintos niveles de ruido. Como se mencionó anteriormente, y se evidencia con las barras de error, la variación de dicho parámetro es muy acotada.

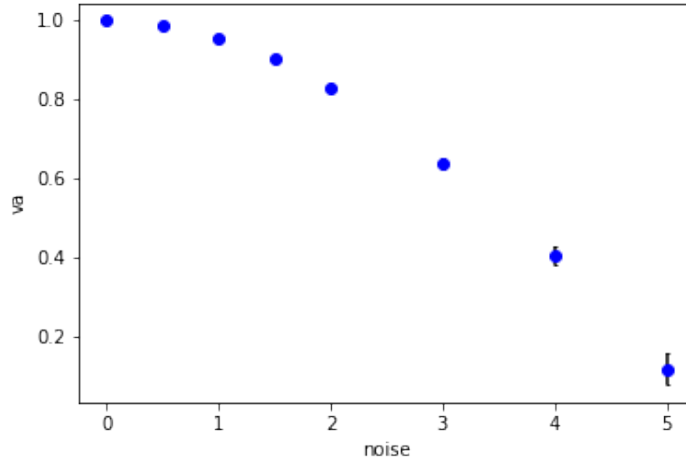


Figura 4:  $V_a$  en función del ruido con 1500 partículas y densidad 15 en el estado estacionario.

El segundo caso de estudio posee una densidad media, y como se puede observar en la fig. 5, las partículas no logran polarizarse, sino más bien se encuentran dispersas con cierta correlación.

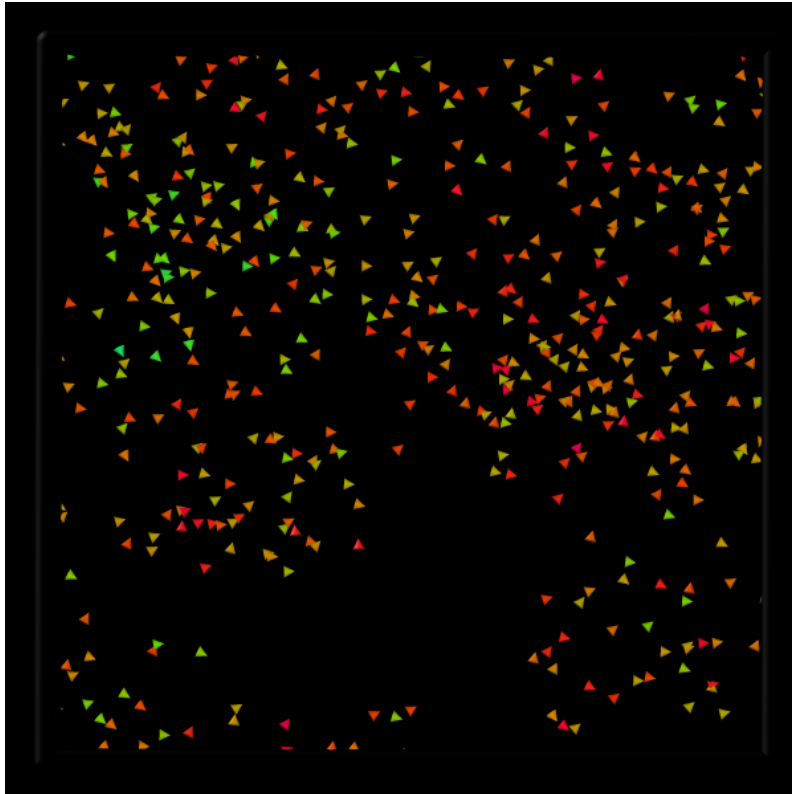


Figura 5: Fotograma de animación con 500 partículas, densidad 5 y ruido 2.0.

Luego en la figura 6 se muestra cómo evoluciona el parámetro de orden a lo largo de las iteraciones. A diferencia del caso anterior, aquí se evidencia una mayor influencia del ruido generando mayores variaciones en los valores del parámetro de orden a lo largo de las iteraciones.

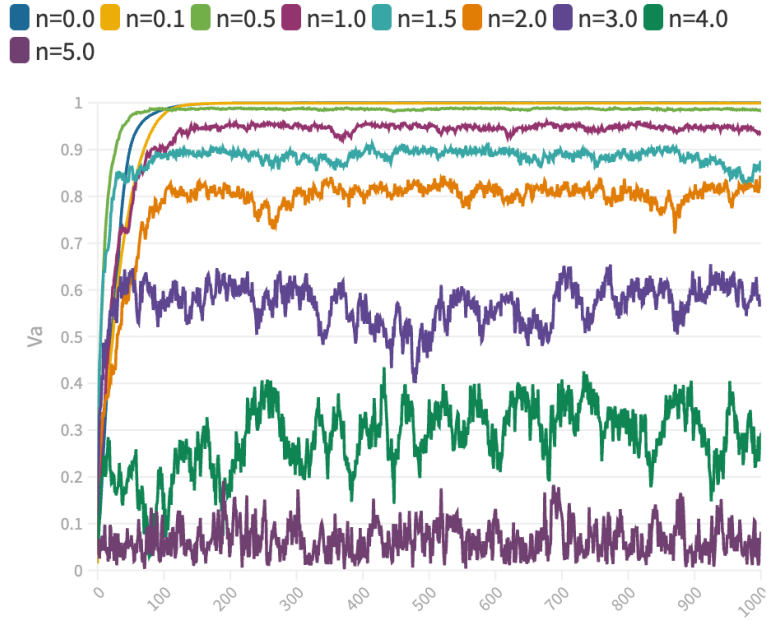


Figura 6:  $V_a$  en función de las iteraciones con 500 partículas y densidad 5.

Por último, en la figura 7 se puede ver el valor promedio del parámetro de orden en el estado estacionario para distintos niveles de ruido. Comienza a verse mayor desvío estándar a medida que aumenta el ruido.

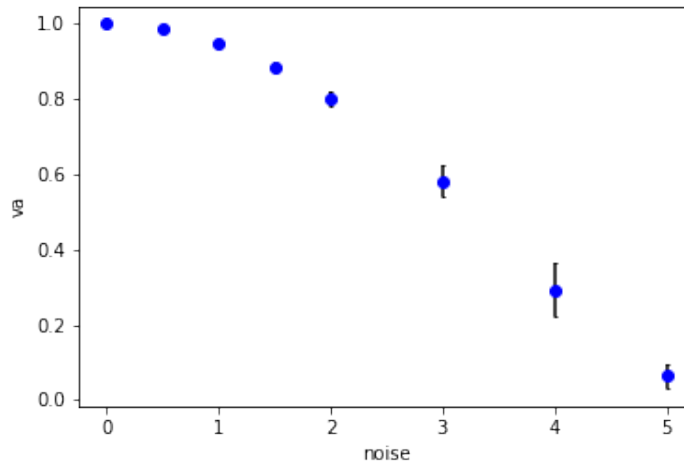


Figura 7:  $V_a$  en función del ruido con 500 partículas y densidad 5 en el estado estacionario.

El tercer y último caso de estudio posee la densidad más baja, y como se puede observar en la fig. 8, las partículas forman grupos.



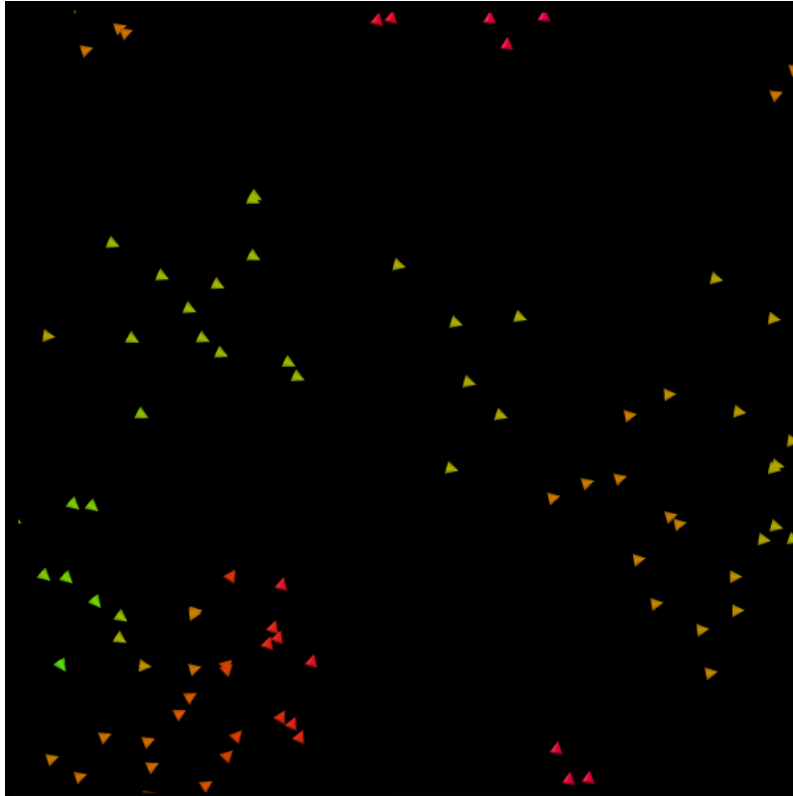


Figura 8: Fotograma de animación con 100 partículas, densidad 1 y ruido 0.1.

Luego en la figura 9 se muestra cómo evoluciona el parámetro de orden a lo largo de las iteraciones. Se puede apreciar que es el caso más afectado por el ruido produciendo grandes saltos en el parámetro de orden cuando el ruido es alto.

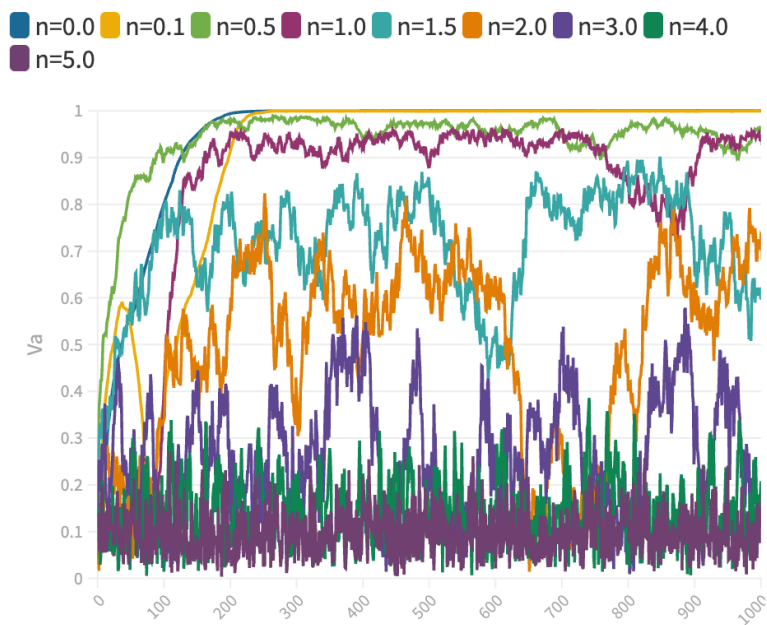


Figura 9:  $V_a$  en función de las iteraciones con 100 partículas y densidad 1.

Por último, en la figura 10 se puede ver el valor promedio del parámetro de orden en el estado estacionario para distintos niveles de ruido. Se observa que es el caso con mayor desvío estándar a mayores niveles de ruido.

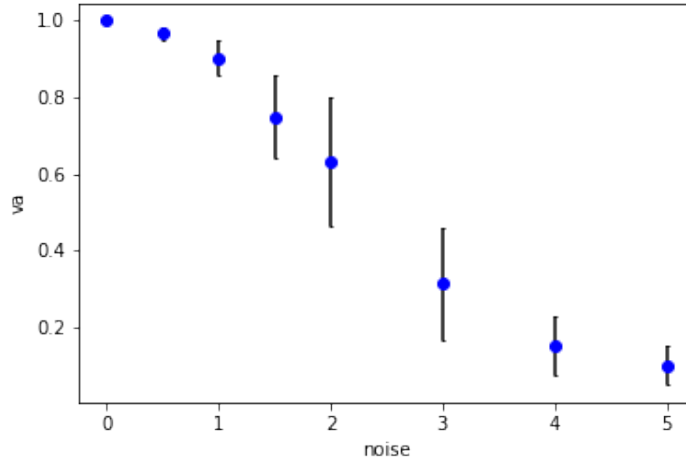


Figura 10:  $V_a$  en función del ruido con 100 partículas y densidad 1 en el estado estacionario.

En la figura 11 se superponen las figuras 4, 7, 10 mostrando el parámetro de orden promedio en función del ruido aplicado para los distintos valores de densidad. Se observa que a medida que la densidad disminuye, el ruido afecta en mayor nivel al valor promedio que alcanza el parámetro de orden.

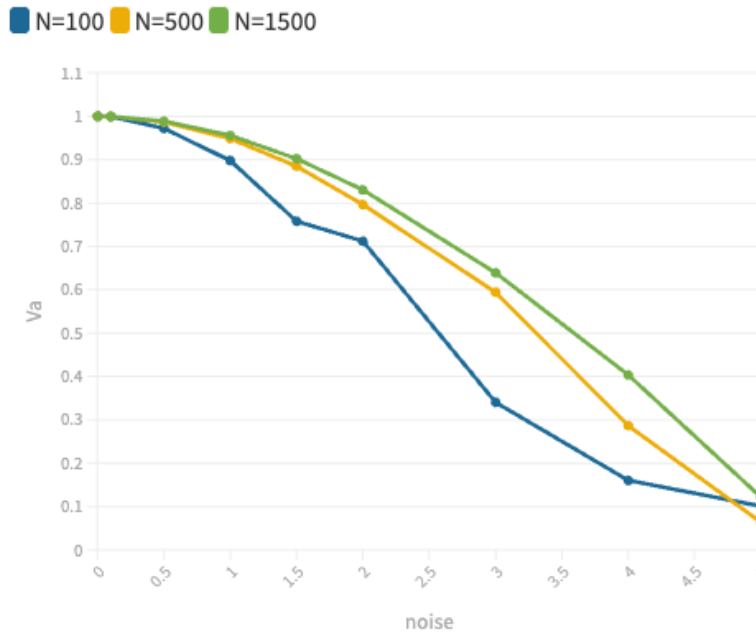


Figura 11:  $V_a$  en función del ruido para distintos  $N$  en el estado estacionario.

Para finalizar, la figura 12 expone cómo varía el valor promedio alcanzado del parámetro de orden cuando el ruido es constante para todas las simulaciones, en este caso 2.0. Como se puede apreciar, cuando la densidad aumenta, el ruido tiene menos influencia y permite al sistema alcanzar un nivel más alto de polarización.

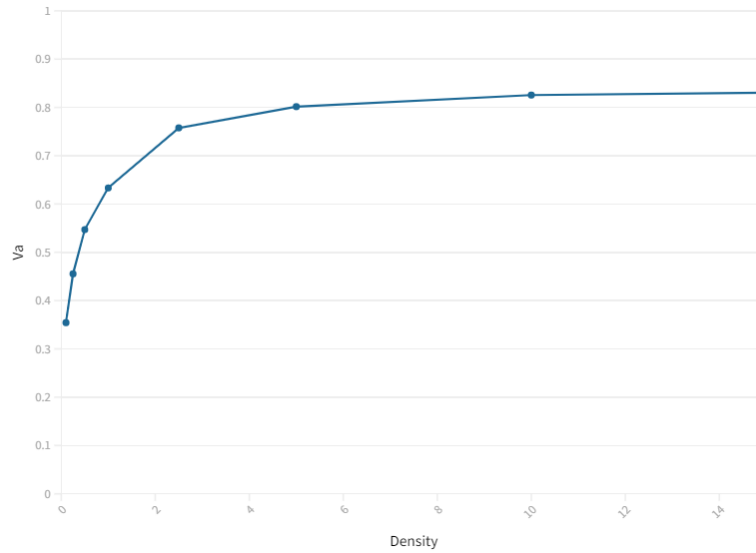


Figura 12:  $V_a$  en función de la densidad con ruido 2.0 en el estado estacionario.

## 5. Conclusiones

A partir de los resultados previamente expuestos, se logró llegar a las siguientes conclusiones:

1. A mayor densidad del sistema, el parámetro de orden aumenta. Es decir que el sistema alcanza un estado estacionario en menor tiempo.
2. A mayor ruido dentro del sistema, el parámetro de orden disminuye y el sistema no alcanza un estado estacionario (existe una variación entre máximos y mínimos).
3. A menor cantidad de partículas en el sistema el ruido produce un mayor impacto negativo en el parámetro de orden.
4. A mayor densidad, la variación del parámetro de orden disminuye.

## Referencias

- [1] Tamas Vicsek; Andras Czirok; Eshel Ben-Jacob; Inon Cohen; Ofer Shochet, "Novel Type of Phase Transition in a System of Self-Driven Particles," *PHYSICAL REVIEW LETTERS*, vol. 75, no. 6, pp. 1226–1229, 1995.