

# Trabajo Práctico N° 2

## Autómatas Celulares

- Banfi, Malena - 61008
- Caeiro, Alejo Francisco - 60692
- Fleischer, Lucas - 61153

# Tabla de contenidos

**01**   Introducción

**02**   Implementación

**03**   Simulaciones

**04**   Resultados

**05**   Conclusiones

01

# Introducción

Automatas celulares: Off-Lattice  
Bandadas de agentes autopropulsados



# 1.1 Introducción

- Objetivo: representación del comportamiento de sistemas de partículas autopropulsadas.
- Comportamiento colectivo de sistemas naturales.
- Este enfoque captura fenómenos naturales complejos.



## 1.2 Fundamentos

→ Las partículas se desplazan con rapidez constante.

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t)\Delta t$$

→ La dirección de movimiento depende de partículas vecinas y un valor de ruido:

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

$$\langle \theta(t) \rangle_r = \arctan2(\langle \sin(\theta(t)) \rangle_r, \langle \cos(\theta(t)) \rangle_r)$$

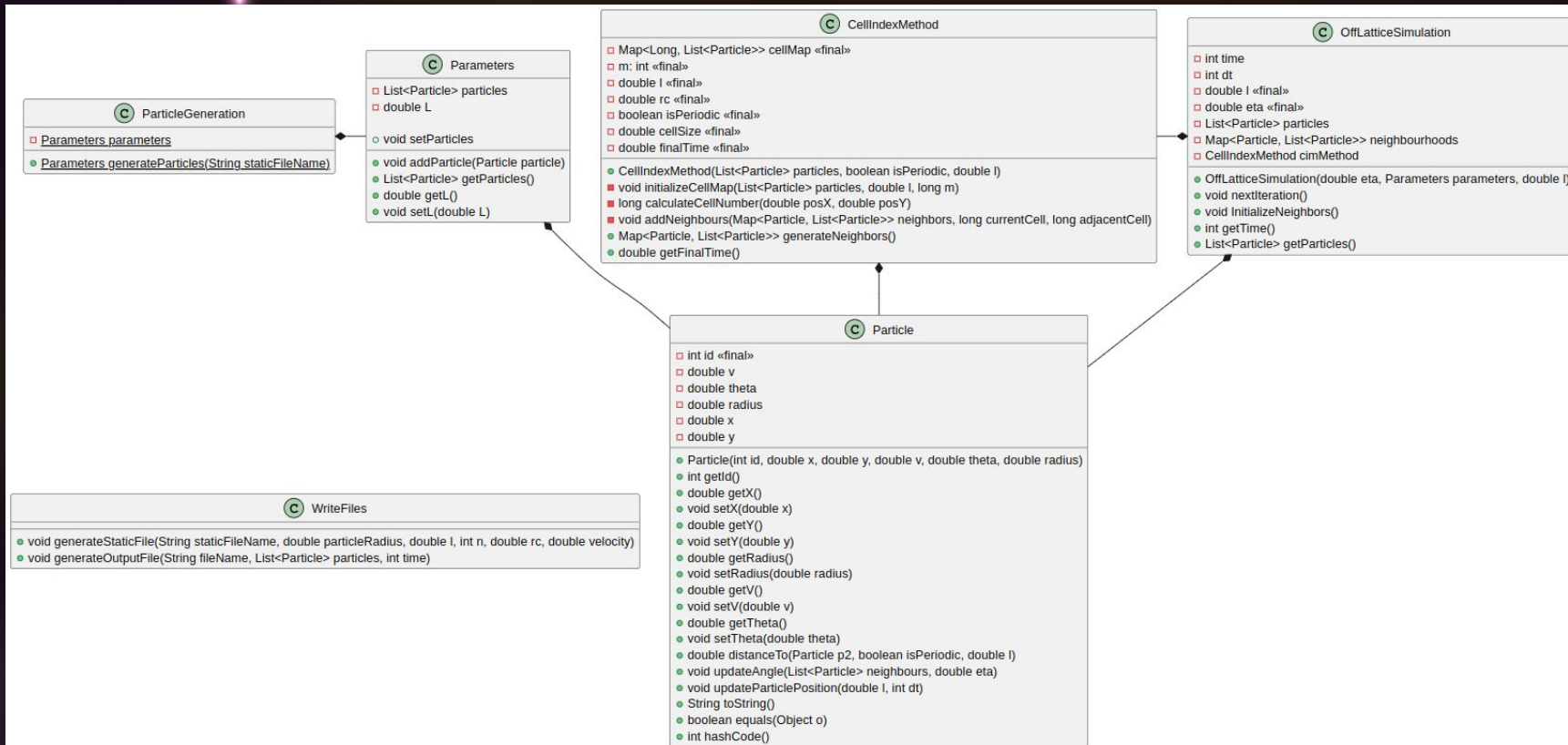
$$\Delta\theta \in \left[-\frac{\eta}{2}, \frac{\eta}{2}\right]$$

02

# Implementación



# 2.1 Arquitectura



## 2.2.1 Actualización de estado

```
void nextIteration() {  
    InitializeNeighbors()  
  
    foreach(Particle particle : particles){  
        particle.updateParticlePosition(l, dt)  
        particle.updateAngle(neighbourhoods.get(particle), eta)  
    }  
  
    time += dt  
}
```



## 2.2.2 Actualización del ángulo y posición

```
void updateParticlePosition(double l, int dt){

    double vx = getV() * cos(getTheta())
    double vy = getV() * sin(getTheta())

    newX = (getX() + vx * dt) % l
    newY = (getY() + vy * dt) % l

    updatedX = newX < 0 ? newX + l : newX
    updatedY = newY < 0 ? newY + l : newY

    setX(updatedX)
    setY(updatedY)
}
```

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t)\Delta t.$$

```
void updateAngle(List<Particle> neighbours, double eta){

    totalSin = sin(getTheta())
    totalCos = cos(getTheta())

    foreach(Particle p : neighbours){
        totalSin += sin(p.getTheta())
        totalCos += cos(p.getTheta())
    }

    avgSin = totalSin / (neighbours.size() + 1)
    avgCos = totalCos / (neighbours.size() + 1)

    setTheta(atan2(avgSin, avgCos) + random(-eta/2, eta/2))
}
```

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

03

# Simulaciones



## 3.1 Sistema a simular

- $N$  partículas puntuales con una dirección variable, dada por un ángulo  $\theta$  con velocidad  $v$  constante que se mueven en el continuo dentro de la celda de lado  $L$ , con condición de periodicidad.
- Amplitud de ruido  $\eta$  y paso temporal  $dt$ .
- Radio de interacción  $r_c$ .

### Parametros Fijos:

- $r = 0.00$
- $r_c = 1.00$
- $v = 0.03$
- $dt = 1$
- Iteraciones = 3000

### Parametros Variables:

- $\rho \in [0.25, 10.0]$
- $\eta \in [0.0, 3.5]$
- $N \in [100, 4000]$
- $L \in [5, 20]$

## 3.2 Parámetro de orden

$$v_a = \frac{1}{Nv} \left| \sum_{i=1}^N \mathbf{v}_i \right|$$

- Tiende a cero cuando las partículas apuntan en direcciones aleatorias.
- Desorden total.
- Tiende a uno cuando las partículas apuntan en la misma dirección.
- Partículas polarizadas.

04

# Resultados



## 4.1 Animación 1 - Ruido Bajo

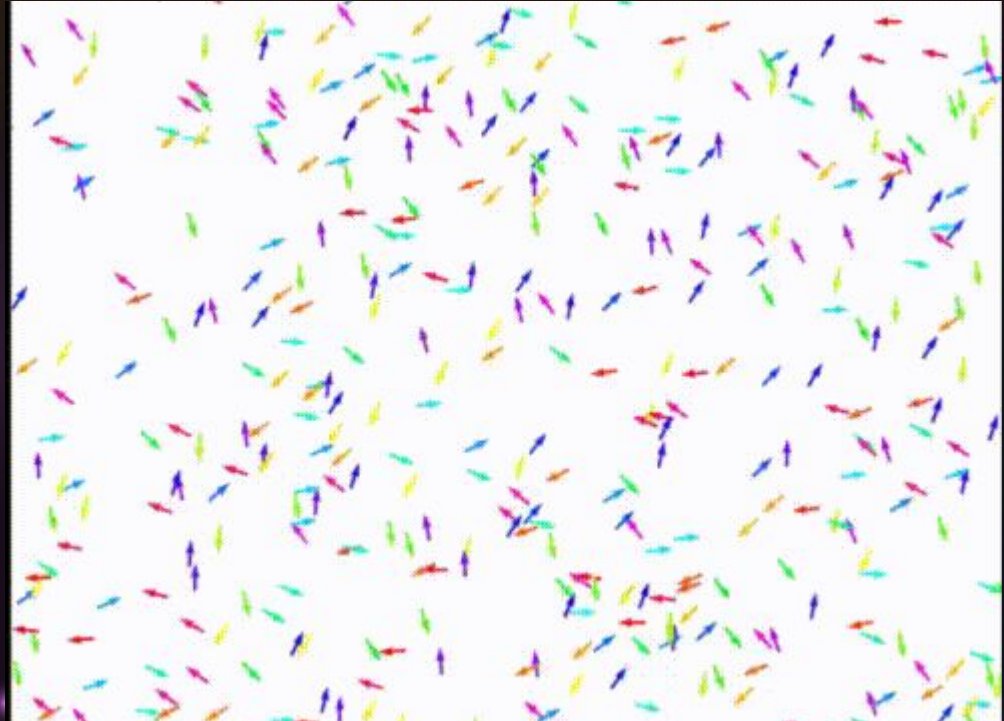
- $L = 10.0$
- $\eta = 0.0$
- $N = 400$



<https://www.youtube.com/watch?v=N8FSBUcgrIU>

## 4.1 Animación 2 - Ruido Medio

- $L = 10.0$
- $\eta = 1.5$
- $N = 400$

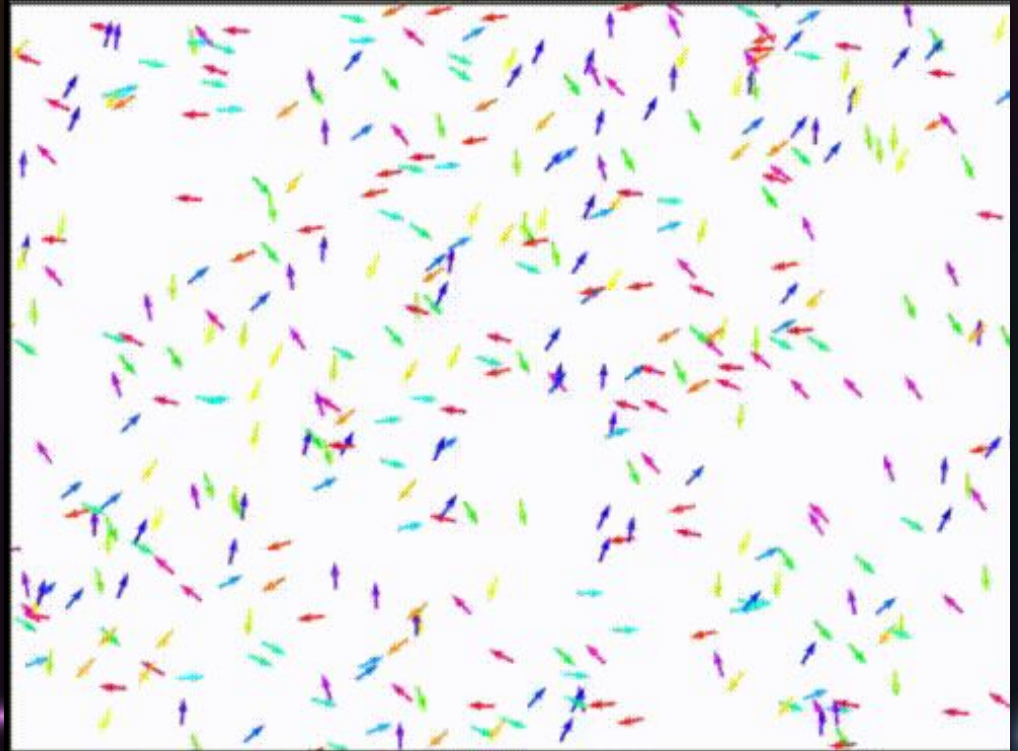


<https://www.youtube.com/watch?v=BPPpHRPVnyY>



## 4.1 Animación 3 - Ruido Alto

- $L = 10.0$
- $\eta = 3.5$
- $N = 400$

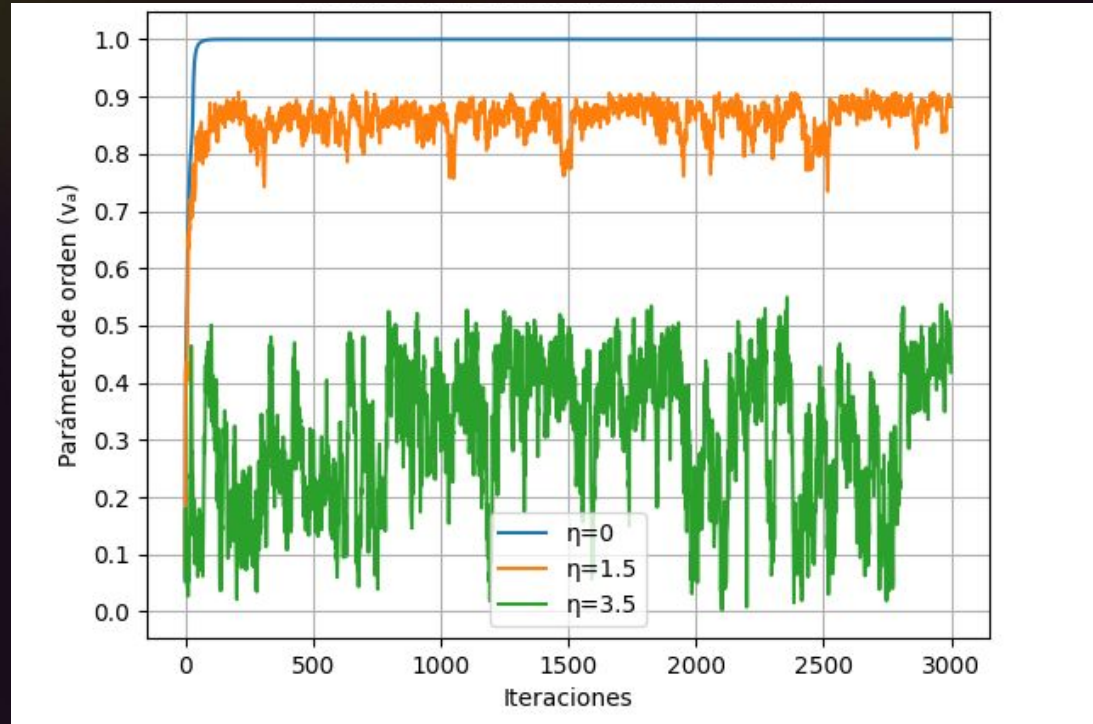


<https://www.youtube.com/watch?v=AiuhNnWX0vk>

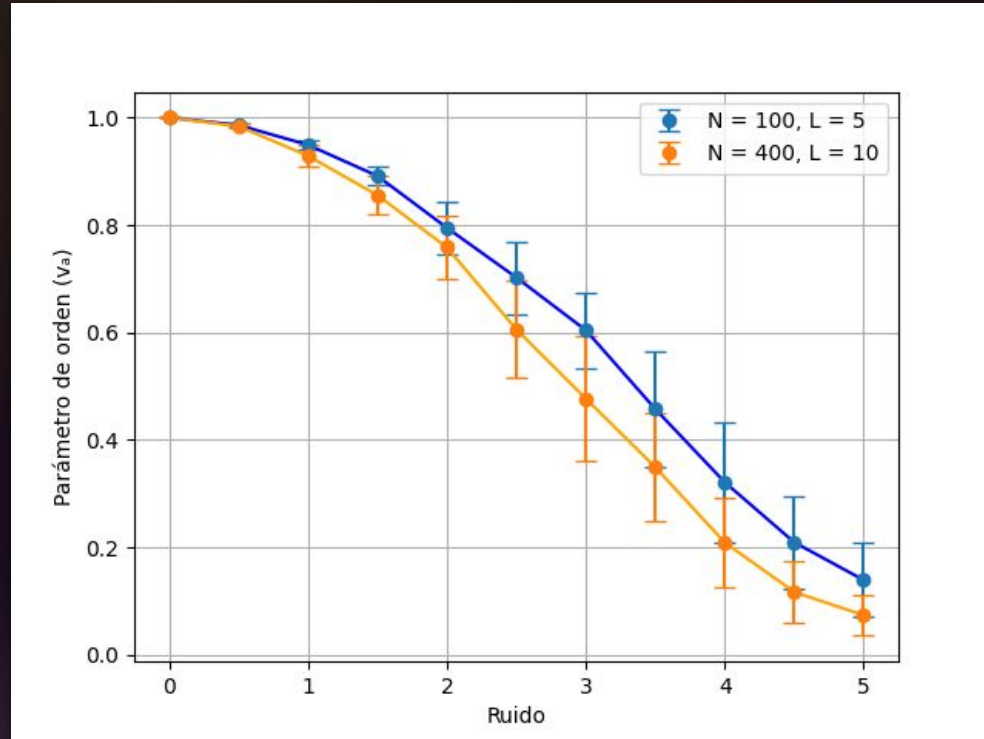


## 4.2 Evolución de $V_a$ en función de las iteraciones

- $L = 10.0$
- $N = 400$

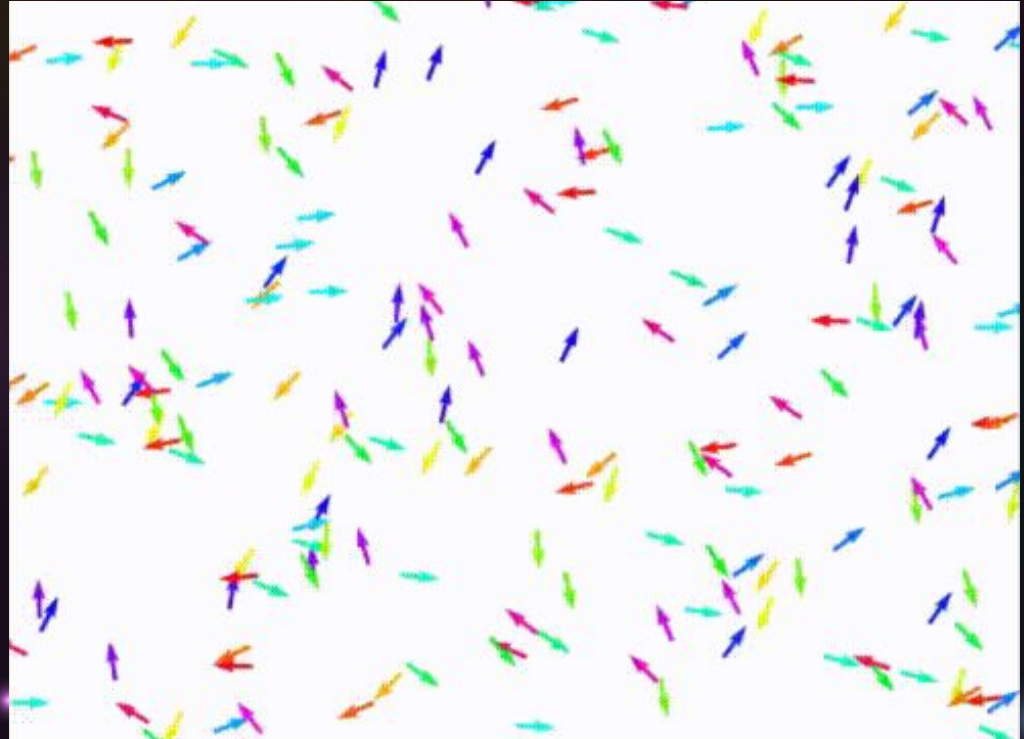


## 4.3 Evolución de $V_a$ en función del ruido



## 4.4 Animación 4 - Densidad Baja

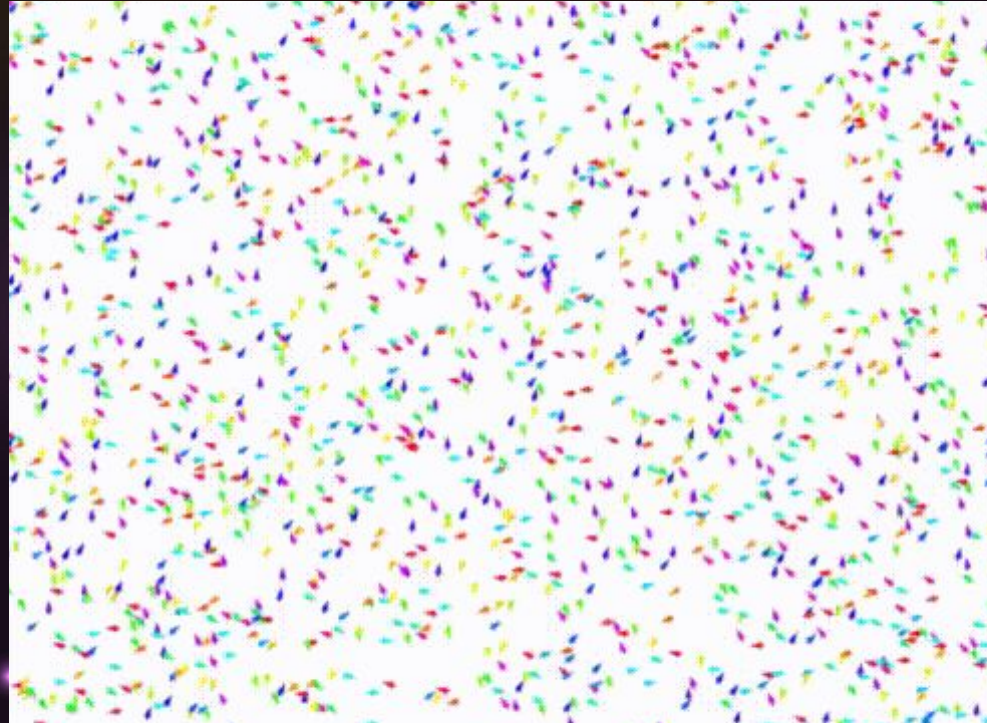
- $\rho = 1$
- $\eta = 2.0$
- $N = 400$



<https://youtu.be/EbXbxLcZh5s>

## 4.4 Animación 5 - Densidad Media

- $\rho = 3$
- $\eta = 2.0$
- $N = 1200$

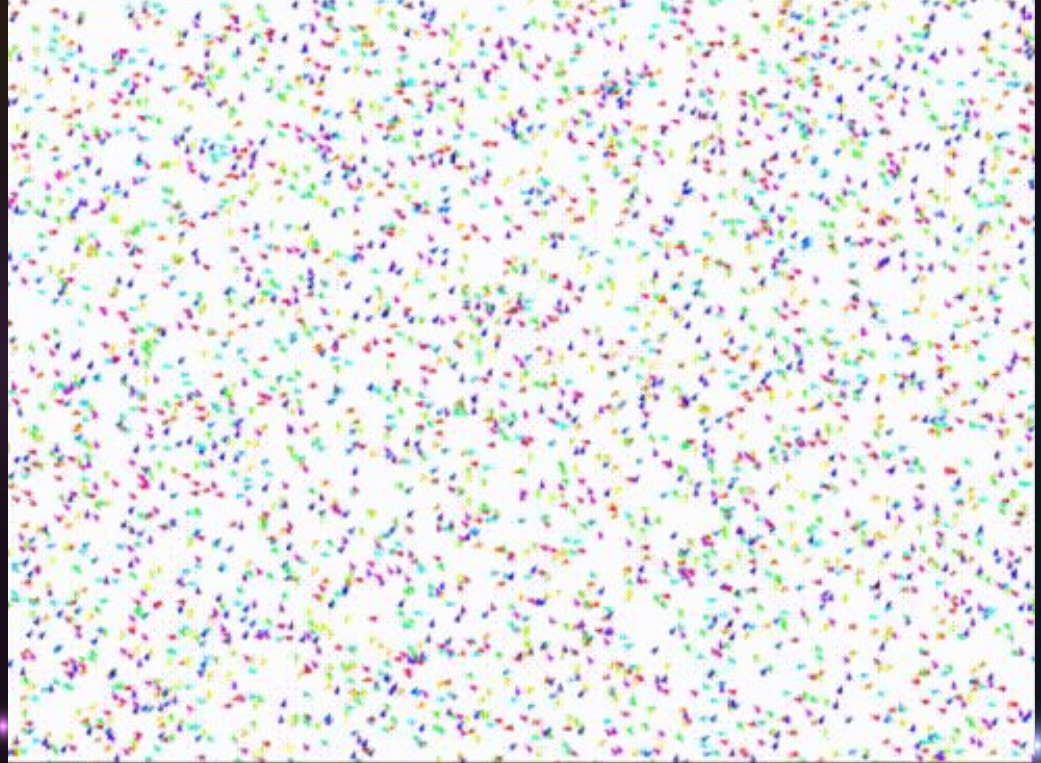


[https://youtu.be/eJXG\\_7RRzLw](https://youtu.be/eJXG_7RRzLw)



## 4.4 Animación 6 - Densidad Alta

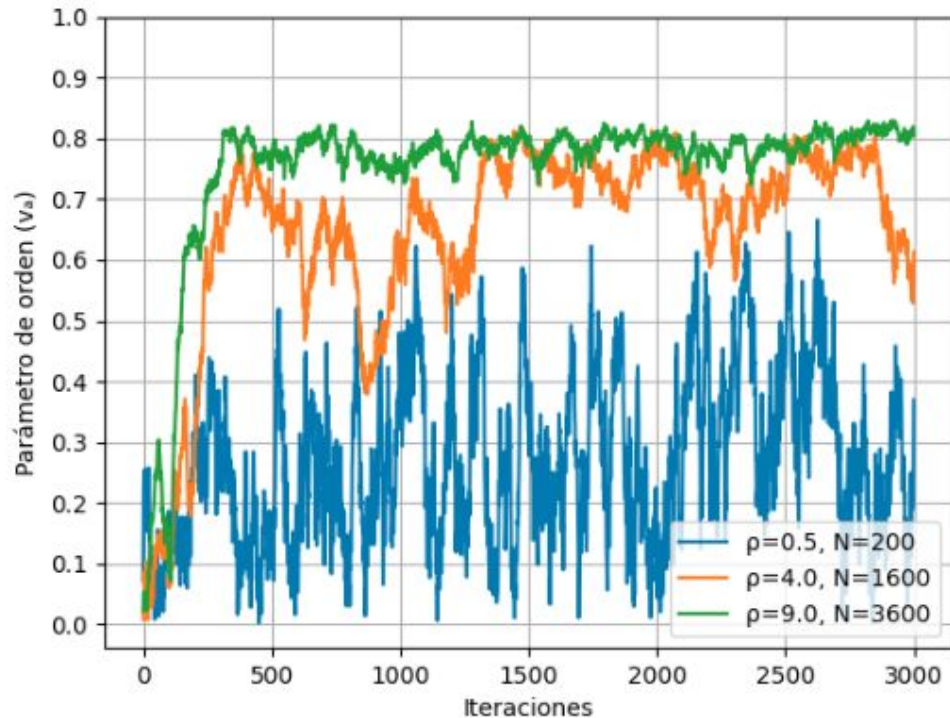
- $\rho = 9$
- $\eta = 2.0$
- $N = 2400$



<https://youtu.be/IVLASONFLnc>

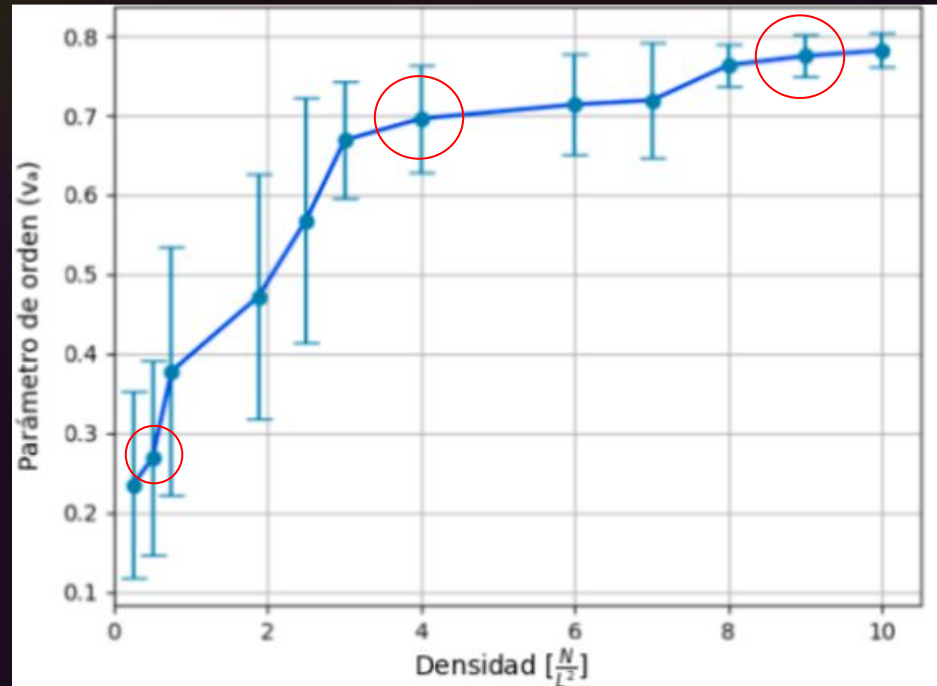
## 4.7 Evolución de $V_a$ en función de las iteraciones

- $\eta = 2.0$



## 4.8 Evolución de $V_a$ en función de la densidad

- $L = 20.0$
- $\eta = 2.0$



05

# Conclusiones





## 5. Conclusiones

- Un aumento en el ruido provoca una disminución de la polarización. Es decir, el orden es inversamente proporcional al ruido.
- Mientras haya menos partículas, mayor es la polarización manteniendo la densidad constante.
- Un aumento en la densidad provoca un aumento en la polarización. Es decir, el orden es proporcional a la densidad.
- A mayor densidad se alcanza un estado estacionario más rápidamente y la variación del parámetro de orden disminuye.

The background is a dark, deep purple gradient. It is decorated with several glowing, translucent spheres in shades of yellow, orange, and blue. These spheres are connected by thin, swirling lines of the same colors, creating a sense of motion and depth. The overall effect is reminiscent of a stylized atomic model or a cosmic scene.

# Fin

## Muchas gracias