

# **Flocking y Foraging Cooperativo mediante Aprendizaje por Refuerzo Multiagente**

Proyecto Final – Maestría en Ciencia  
de Datos

Aprendizaje de Máquina II

Enrique Ulises Báez Gómez Tagle

November 25, 2025



# 1. Motivación y Problema

*Coordinación emergente bajo escasez de recursos*



- En la naturaleza: bandadas de aves, cardúmenes de peces, colonias de hormigas.
- Comportamientos colectivos complejos a partir de reglas locales simples.
- Pregunta central: **¿podemos reproducir esta coordinación con agentes que aprenden?**
- En particular: **forrajeo** en entornos con recursos limitados que se regeneran.

- Sistema multiagente:
  - $N$  agentes en un mundo 2D continuo.
  - $M$  parches de recursos con stock finito y **regeneración logística**.
  - Observaciones mayormente locales, sin comunicación explícita.

- **Pregunta principal:**

*¿Pueden agentes que comparten una política aprendida con RL desarrollar estrategias cooperativas efectivas incluso cuando hay más agentes que fuentes de recursos?*

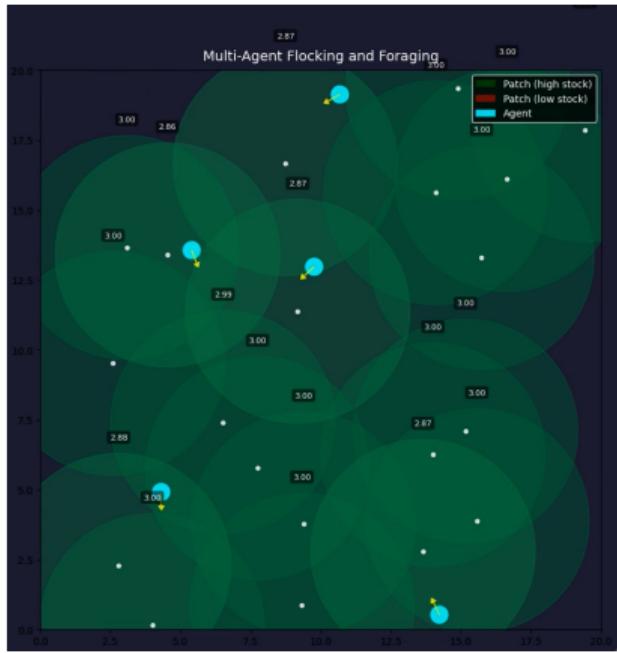
## 2. Entorno y Metodología

*Del mundo simulado a la  
política compartida*



- Mundo 2D continuo con límites reflectivos.
- Agentes:
  - 5 acciones discretas: girar, acelerar, frenar, no-op.
  - Observaciones de 13 dimensiones (vecinos + parche cercano + stock medio).
- Parches de recursos:
  - Stock  $S(t) \in [0, S_{\max}]$ .
  - Regeneración logística:

$$\frac{dS}{dt} = rS \left(1 - \frac{S}{S_{\max}}\right)$$



**Figure:** Easy mode: 5 agentes, 20 parches.

| Modo   | # Agentes | # Parches | Ag/Patch |
|--------|-----------|-----------|----------|
| Easy   | 5         | 20        | 0.25     |
| Medium | 10        | 18        | 0.56     |
| Hard   | 10        | 15        | 0.67     |
| Expert | 12        | 10        | 1.20     |

- Escalamos desde **abundancia** (0.25) hasta **escasez extrema** (1.20).
- En Expert mode hay más agentes que parches: **no todos pueden alimentarse a la vez.**

- Algoritmo: **Recurrent PPO** (Stable-Baselines3 + sb3-contrib).
- Arquitectura:
  - MLP (2 capas de 64 neuronas) + **LSTM de 256 unidades**.
  - Actor-Crítico con política **compartida** entre todos los agentes.
- Función de recompensa **multiobjetivo**:
  - Forrajeo: intake de alimento.
  - Flocking: cohesión, alineación, separación, bonus de grupo.
  - Penalización por aglomeración en parches saturados.
  - Bonus de **equidad**: término basado en  $(1 - \text{Gini})$  al final del episodio.

### 3. Resultados

*Eficiencia, equidad y comportamientos emergentes*



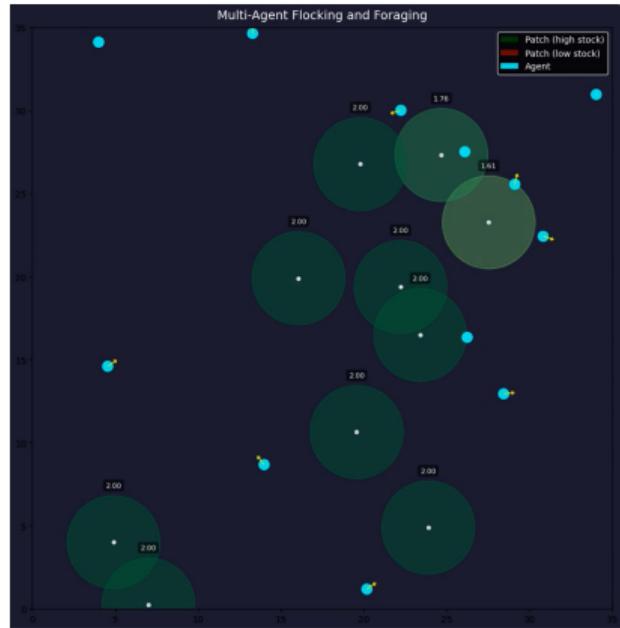
| Modo   | Eficiencia media | Máx. observada |
|--------|------------------|----------------|
| Easy   | 87.2%            | 100%           |
| Medium | 72.6%            | 100%           |
| Hard   | 49.9%            | 94.3%          |
| Expert | 37.1%            | 64.5%          |

- La eficiencia decrece con la escasez, pero **no colapsa** en Expert mode.
- Incluso con agentes > parches, el sistema mantiene  $\sim 37\%$  de eficiencia.
- Múltiples episodios en Easy y Medium alcanzan el óptimo teórico.

| <b>Modo</b> | <b>Gini medio</b> | <b>Interpretación</b> |
|-------------|-------------------|-----------------------|
| Easy        | 0.11              | Muy equitativo        |
| Medium      | 0.27              | Buena equidad         |
| Hard        | 0.48              | Desigualdad moderada  |
| Expert      | 0.57              | Alta desigualdad      |

- A mayor escasez, **más desigual** la distribución de recursos entre agentes.
- Observamos un **trade-off** eficiencia vs. equidad:
  - Los episodios más eficientes tienden a tener menor Gini.

- **División dinámica de grupos (flock splitting):**
  - Sub-grupos explotan regiones distintas del mapa.
- **Rotación de parches:**
  - Agentes abandonan parches casi agotados y regresan tras regeneración.
- **Compartición implícita:**
  - Agentes exitosos “arrastran” a otros vía señales de flocking.



**Figure:** Expert mode: 12 agentes, 10 parches.

- Baseline: reglas clásicas de *boids* (cohesión, alineación, separación) + atracción al parche más cercano.
- En Easy mode:
  - Boids  $\approx$  100% eficiencia, Gini  $\approx$  0 (distribución casi perfecta).
  - RL (PPO-LSTM)  $\approx$  87% eficiencia, Gini  $\approx$  0.11.
- **Interpretación:**
  - Con abundancia, reglas fijas bien calibradas son casi óptimas.
  - El valor del RL aparece al escalar a **escasez extrema**, donde se adapta sin re-diseñar reglas.

## 4. Conclusiones y Demos

*De la simulación a las aplicaciones prácticas*

ons

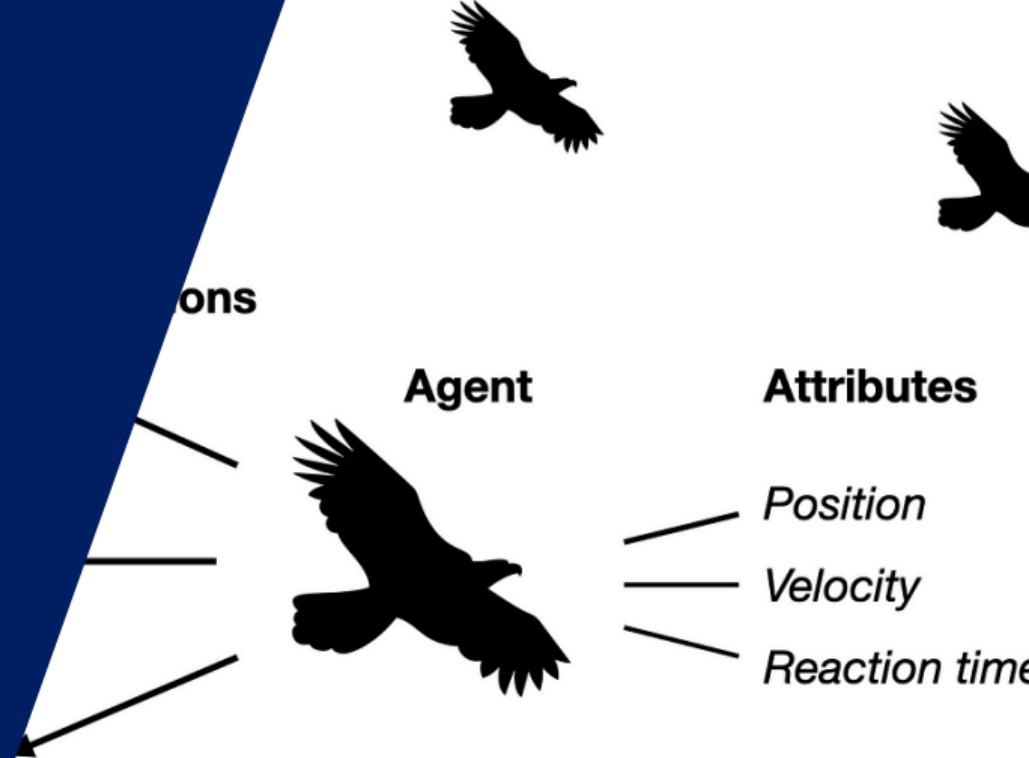
Agent

Attributes

Position

Velocity

Reaction time



- Es posible lograr **coordinación emergente sin comunicación explícita** usando RL multiagente + flocking.
- La combinación de **recompensas de forrajeo y flocking** es esencial; remover una de ellas degradaría fuertemente el desempeño.
- El sistema escala desde abundancia (87%) hasta escasez extrema (37%) manteniendo comportamientos cooperativos.
- La memoria (LSTM) ayuda a evitar revisitaciones ineficientes a parches agotados.

- **Trabajo futuro:**

- Añadir canales de comunicación aprendida (CommNet, GNNs).
- Escalar a 20+ agentes y entornos con obstáculos.
- Agentes heterogéneos con roles diferenciados.

- **Código y resultados:**

- Repositorio GitHub: [multi-agent-flocking-foraging-rl](#)
- Videos de demostración: [videos](#)
- Dashboard interactivo (Streamlit):  
<http://54.165.139.51:8502>

Let's Get into  
***CODE!***

