Predator–prey survival pressure

Rafaelle Lacraz, Titouan Steyer

Collective Behaviour Course, Research Seminar Report

December 2025

Supervisor: Iztok Lebar Bajec

This project explores how collective behaviors can emerge in predator–prey systems driven by minimal survival incentives. Following the framework of Li et al. [3], we implement a 2D continuous environment with multiple prey and a few predators, each governed by physical dynamics and subject to active and passive forces. The reward function is deliberately simple predators gain a reward when catching prey, and prey receive a penalty when caught. Using public code as a starting point, we reproduce baseline behavior and visualize system evolution under random motion. Quantitative metrics such as the Degree of Sparsity (DoS) and Degree of Alignment (DoA) allow us to characterize emergent group dynamics. This report serves as a progress checkpoint toward building an adaptive reinforcement learning model that captures realistic group behavior.

1 Introduction

Understanding how collective behaviours such as swarming, flocking, or coordinated pursuit emerge in animal groups is a central question in behavioural biology and bio-inspired robotics.

Li et al. [3] address this limitation with a minimalist predator–prey framework based on multi-agent reinforcement learning (MARL), where no social rules are explicitly encoded. Agents receive only a simple survival-based reward (+1 for a predator catching a prey, -1 for a prey being captured).

In this project, we reproduce and analyse this framework, starting from a simplified baseline environment. Our goal is to study how collective behaviours emerge, how learning influences predator–prey interactions, and how system parameters shape these dynamics.

2 Methods

2.1 Environment and Implementation

To reproduce the conditions described in Li et al. [3], we implemented a continuous two-dimensional square environment in which predators and prey move and interact. Two types of boundary conditions are supported:

Periodic boundaries (torus). When an agent crosses one side of the domain, it reappears on the opposite side with the same velocity. This configuration removes any edge effects and approximates an unbounded environment, matching the setup of the original paper.

Solid boundaries (walls). Agents remain confined inside the square and bounce when reaching the boundary. Collisions are modeled using a repulsive spring force ($k = 50$ N/m), preventing agents from crossing the walls.

At the beginning of each simulation, predators ($n_0 = 3$) and prey ($n_1 = 10$) are placed at random positions with random headings. All agents are represented as disks with mass $m = 1$ kg. Predators and prey differ slightly in their locomotion capabilities, with maximum speeds of 0.35 m/s and 0.30 m/s respectively. Each agent receives two control inputs: a forward acceleration and a rotational command, both bounded to ensure stable and realistic movement.

The physics of the environment follow Newtonian dynamics and include:

- **active forces** generated by the agent's own propulsion (forward thrust + rotation),
- **viscous drag**, proportional to velocity, which slows down motion,

- **collision forces** with other agents or with boundaries, based on Hooke's law.

For prey, we optionally implement Couzin-type social rules (repulsion, alignment, attraction) to test deterministic collective behaviour before applying reinforcement learning. These forces allow us to evaluate whether the environment correctly produces known emergent patterns prior to introducing learned policies.

Overall, the implementation remains faithful to the physical model described in Li et al., while providing enough flexibility for experimenting with different configurations and preparing the environment for MARL training.

2.2 Collective Behavior Metrics

In addition to visual inspection, we plan to use quantitative metrics to assess the emergent behaviors within the agent populations. Following the definitions introduced by Li et al. [3], we will compute two key indicators: the *Degree of Sparsity* (DoS) and the *Degree of Alignment* (DoA). These metrics are designed to evaluate local cohesion and heading synchronization within a species, while being invariant to global translations and rotations of the group.

Degree of Sparsity (DoS). The DoS $\in [0, 1]$ quantifies the spatial distribution of agents by averaging the normalized distance to the nearest neighbor among conspecifics. It is defined as:

$$\mathrm{DoS} = \frac{1}{TND} \sum_{t=1}^{T} \sum_{j=1}^{N} \left\| \mathbf{x}_j(t) - \mathbf{x}_{k(j)}(t) \right\|$$

where $\mathbf{x}_j(t) \in \mathbb{R}^2$ is the position of the j-th agent at time step t, and $k(j)$ is the index of its nearest neighbor: $k(j) = \arg\min_{\substack{k \in \{1,\ldots,N\} \\ k \neq j}} \|\mathbf{x}_j(t) - \mathbf{x}_k(t)\|$ T is the episode length (number of time steps), N is the number of agents of the same type (e.g., prey), and D is the environment size, defined as the maximum possible distance between two agents. A high DoS value reflects a dispersed group, while a low DoS indicates strong cohesion and clustering among agents.

Degree of Alignment (DoA). The DoA $\in [0, 1]$ measures the similarity in heading directions between neighboring agents and is defined as:

$$\mathrm{DoA} = \frac{1}{2TN} \sum_{t=1}^{T} \sum_{j=1}^{N} \left\| \mathbf{h}_j(t) - \mathbf{h}_{k(j)}(t) \right\|$$

where $h_j(t) \in \mathbb{R}^2$ is the heading vector (unit norm) of agent j at time t, and $k(j)$ is again its nearest neighbor as defined above. The factor $1/2$ ensures that the metric remains bounded in $[0, 1]$. A low DoA indicates poor alignment, while higher values reflect synchronized heading directions.

Note that DoA does not reflect the global average heading. A group may split into several aligned sub-flocks moving in different directions; the global heading would cancel out, while the local DoA remains high, making it a more suitable measure of flocking behaviour.

2.3 Reward Function

We adopt a minimal reward structure, consistent with Li et al. [3], to promote emergent behaviors without artificial bias:

- Predators receive $+1$ when catching a prey.
- Prey receive -1 when caught.

No additional incentives (such as cohesion or alignment) are used. This choice allows agents to autonomously develop strategies based solely on their individual objective of survival or capture. We found a public GitHub repository [4] whose author based their work on the same article, and we used some of their Python files as a starting point for our own model implementation.

2.4 Baseline Objective

The goal of this first phase is to reproduce the baseline behavior described by Li et al. [3]:

- A continuous 2D environment with multiple prey and a few predators.
- Agents modeled as disks defined by their position, velocity, and heading.
- Implementation of a reward function based on predator-prey interactions.

- Visualization of the evolution of collective behavior metrics (DoS and DoA) and cumulative reward over time.

This baseline will serve as the foundation for further analysis of collective behaviour and for exploring modifications of the original model.

3 Results

3.1 Baseline observations without learning

As a first step, we examined the interaction dynamics between three predators and multiple prey in the continuous 2D environment, before introducing any reinforcement learning mechanism. In this baseline setup, all agents follow random motion policies: no social, cooperative, or strategic rules are defined. The environment uses periodic boundary conditions, meaning that when an agent crosses one side of the square domain, it reappears on the opposite side. This creates an effectively unbounded space and avoids edge effects.

Even in the absence of behavioural rules, we observed occasional temporary clustering of prey or brief chasing interactions caused purely by random motion and spatial proximity. These qualitative observations helped verify that the environment is functioning correctly and that agents move and interact as intended.

To quantify these baseline dynamics, we computed the Degree of Sparsity (DoS) and Degree of Alignment (DoA) throughout the simulation (Figure 1). In this setting, the DoS remains close to 1, indicating a uniformly dispersed population, while the DoA stays near 0, confirming the absence of coordinated movement. These results match the expected behaviour reported by Li et al. [3] and validate our environment before introducing learning.

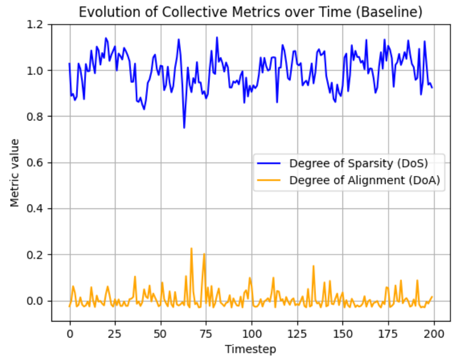

Figure 1: Evolution of the Degree of Sparsity (DoS) and Degree of Alignment (DoA) during the random-motion baseline simulation.

3.2 Extending the Model : Initial learning attempts

In the second part of the project, we attempted to move closer to the framework described by Li et al. [3]. Our goal was to train both predators and prey using the MADDPG algorithm. Before applying reinforcement learning, we first tested a deterministic model based on Couzin's rules in order to better understand how our environment behaves and to build a baseline for comparison.

We also implemented a closed environment with solid borders. In this configuration, agents cannot leave the domain and bounce against the walls. As illustrated in Figure 2, the prey tend to group together while predators actively try to reach them.

In our implementation, when a predator touches a prey, the prey is considered caught: it disappears and respawns at a random location. This choice reflects natural predator–prey interactions more closely. In contrast, the original paper keeps prey in the simulation even after being caught, which may lead to differences in group behaviour.

4 Discussion

The results obtained so far confirm that our environment behaves correctly in simple baseline conditions, and that the collective behaviour metrics (DoS and DoA) react as expected when no learning or coordination is present. The first experiments using Couzin's deterministic rules also helped us verify that prey can display basic grouping tendencies and that predators can successfully chase moving targets, even if these behaviours remain different from those described in the original paper. This deterministic baseline provides a useful reference for interpreting future learning-based results.

Our first attempts to train both predators and prey with MADDPG are still preliminary. The learning curves show some improvement for the

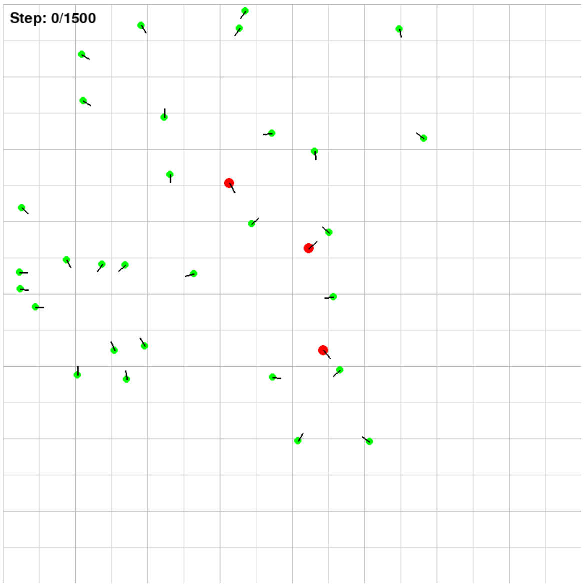

Figure 2: Simulation using Couzin's deterministic rules in a closed environment (solid borders). Prey (green) tend to cluster while predators (red) attempt pursuit.

predators, but remain noisy and unstable. This suggests that the observation space, the reward formulation, and some hyperparameters (such as learning rates or noise levels) still need to be tuned. Comparing deterministic behaviour (Couzin) with learned behaviour will be important in the final stage of the project, as it will help us understand how much of the observed behaviour comes from the environment itself and how much comes from the learning algorithm.

Several related works provide context for our approach. The open-source project PredatorPreyRL [1] shows that training both predators and prey in a similar multi-agent setting is feasible, and gives examples of network architectures and training loops that we can use as inspiration. The work of Ivanov and Palamas [2] demonstrates that swarm-like behaviours can emerge from simple survival pressure and perceptual constraints, without explicitly programming social rules. These studies support our goal of studying how minimal incentives and environmental structure can lead to collective behaviour.

In the final milestone, we will focus on testing our model more systematically and exploring its limits. This will include:

- stabilising the MADDPG training procedure for predators (and, if possible, for prey),
- evaluating how sensitive the learned behaviour is to key parameters (number of agents, boundary conditions, speed ratios, reward weights),
- and comparing the outcomes of the Couzin baseline with those of the learned policies.

By the end of the project, our aim is not only to reproduce some of the behaviours described in Li et al. [3], but also to better understand the limitations of our own implementation and the conditions under which collective predator–prey behaviour can or cannot emerge.

5 Author Contributions

Rafaelle continued improving the predator MADDPG model and worked on stabilising its training. Titouan implemented the closed-boundary environment and developed the corresponding simulation logic. Both authors are now jointly working on extending the MADDPG framework to train the prey as well.

References

[1] AmanPriyanshu. Predatorpreyrl: Multi-agent communication in rl systems. https://github.com/AmanPriyanshu/PredatorPreyRL.

[2] Georgi Ivanov and George Palamas. How predator confusion shapes swarm-like behaviors. *arXiv preprint arXiv:2209.06338*, 2022.

[3] Zhengyang Li, Emanuele Cuccoli, Marco Villani, and Andrea Cavagna. Emergence of collective behaviour in coevolving predator–prey systems. *New Journal of Physics*, 25(9):093028, 2023.

[4] xxnnnnn. Predatorprey_rl_reproduction. https://github.com/xxnnnnn/PredatorPrey_RL_Reproduction, 2024.