Predator-Prey Simulation Using Boids Model

Matija Ojo, Miha Krajnc, Marko Adžaga, and Janez Kuhar

Collective behavior course research seminar report

January 8, 2024

The collective behaviors observed in nature, such as flocking, herding, or schooling, often serve as adaptive strategies that enhance the survival chances of individuals within a group. Understanding these natural behaviors serves as inspiration for designing autonomous agents capable of sophisticated interactions within a simulated environment. Our goal is to simulate prey and predator with different predator tactics (attack center, attack nearest, attack isolated, attacks from various directions), escape maneuvers (split, hourglass, herd, vacuole, flash expansion, fountain) and parameters (perception radius, moving speed, turning speed) in order to conclude how different escape maneuvers affect predator's success.

Collective behavior | Boids | Simulation | Prey-Predator | Escape patterns

Introduction

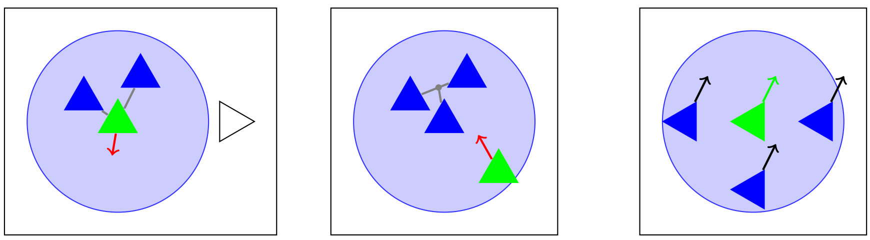
One of the most striking patterns in biology is the formation of animal aggregations. Classically, aggregation has been viewed as an evolutionarily advantageous state, in which members derive the benefits of protection, mate choice, and centralized information, balanced by the costs of limiting resources [2]. We would like to experimentally determine which flocking behaviors help the prey best defend itself against a predator.

The flocking behavior can be simulated in different ways. For example, Heppner and Grenader [3] were modeling birds behavior with stochastic nonlinear differential equations. Oweis, Ganesan, and Cheok [4] took a different approach and modeled birds with a centralized logic (as in the server-client architecture). In 1987, Reynolds [5] proposed a simple algorithm, which was groundbreaking at the time, to model the flocking behavior of birds, herding of sheep, and similar phenomena, known as the Boids (Bird-oid objects) model. In contrast to controlling the interactions of the entire flock, the Boids simulation focuses on dictating the behavior of each individual boid. Despite consisting of a few simple rules, this algorithm produces complex and lifelike behaviors similar to those observed in nature.

Our research is based on a paper by Papadopoulou and others [1], which we will extend with the results of our predator and prey simulation. Although we are not using fuzzy logic to set the direction and speed of our boids, which makes the movement less natural, we have taken some elements for our model from [6]. Specifically, we've set the field of vision for our boids to 300° and implemented occlusion for the predator.

Methods

The Boids model is the foundation of our flocking model. Every object in such a model adheres to the three simple rules as shown in Figure 1.



(a) Collision avoidance (separation). **(b) Cohesion**: gravitate toward the center of the flock. **(c) Alignment**: maintain the same heading and speed.

Figure 1. The basic three rules of the Boids model. We show how the rules apply to a particular boid, marked green, and its neighbors, marked blue. Red arrows indicate the direction in which the observed boid has the tendency to move.

Boids model implementation overview. Each boid B possesses three basic properties: position, velocity, and acceleration. Behavioral attributes include perception radius (r_P), separation radius (r_S), and perception angle (fov). The Euclidean distance, given by $d(p,q)^2 = (p_1 - q_1)^2 + (p_2 - q_2)^2$, is utilized for distance computations.

Collective behavior course research seminar report

Predator-prey interactions is of significant importance in biology and nature itself. The insights gleaned from this research can offer more than a theoretical understanding; they pave the way for the design and optimization of autonomous agents capable of adaptive and context-aware behaviors. The applications range from research in biology to simulations of large amounts of boids found in computer graphics.

Collective behavior | Boids | Simulation | Prey-Predator | Escape patterns

The simulation loop updates boid directions based on three rules: **collision avoidance** (or **separation**), **alignment**, and **cohesion**. The avoidance direction is determined by summing the vector differences between the boid B and its neighbors B_i when their distance is within r_S. The cohesion direction is obtained by averaging the vector differences between the positions of boid B and its neighbors B_i. The alignment direction is computed as the average velocity of neighboring boids B_i, subtracted from the velocity of the boid B, considering neighbors within a perception radius r_P.

The neighbors (all B_i) of a boid B are determined using distance and angle conditions:

$$d(B, B_i)^2 < {r_P}^2 \wedge \text{AngleBetween}(B, B_i) \leq fov \quad [1]$$

Modifying the base Boids model with the **field of vision** is an improvement inspired by [6].

Additionally inspired by [6] is **occlusion**. This effectively disregards boids that remain hidden from the view of a specific boid, as closer boids obstruct their visibility (see Figure 2). Given a list of potential neighboring boids, we must determine which are occluded and in turn take only the nearest (non-occluded) boids as neighbours. This is done by iterating through the list of neighboring boids of boid B and computing the angle between all neighbor pairs (B_i, B_j). If the angle is below a threshold, boids B_i and B_j are considered occluded. Then we just have to determine which neighbor is closer (which one blocks the other). This is done by computing the minimum distance: $min(d(B, B_i), d(B, B_j))$. It is worth noting that we have only added occlusion checks to the predator in our model.

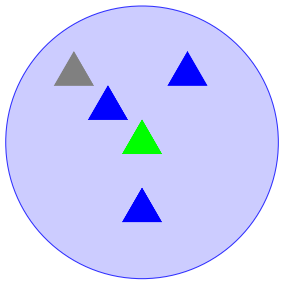


Figure 2. Neighbors (blue) of the observed boid (green). Occluded boid is marked gray.

In order to add even more realism, the **turn speed** of a boid is limited. Whenever the acceleration of a boid is computed, the angle between the acceleration vector and the current heading vector is checked. If it exceeds a threshold, the old heading is rotated by the maximum amount in the given direction and scaled by the magnitude of the acceleration. Therefore boids have a maximum value in which they can turn at each step of the simulation.

Lastly, **predator confusion factor** and **prey's reaction time** were implemented, which further enhances the realism.

Escape maneuvers. In the HoPE model, which was proposed in [1], discrete escape maneuvers are introduced, which we reproduced. These maneuvers involve individual turns away from the predator's heading either based on its position, direction or with turning angles and durations drawn from gamma distributions tailored to empirical data. During maneuvers, coordination with neighbors is absent. Each flock member's likelihood of maneuvering is determined by a unique baseline escape tendency and proximity to the predator.

In our implementation however, maneuvering of flock members is not based on chance, but on whether prey sees the predator or not. This is the only difference between our implementation and the original (aside from additional realistic factors), since using randomness in this situation did not seem logical to us.

We have implemented three escape maneuvers: **position based**, **direction based**, and **zig-zag** escape maneuvers. The position based escape manuever works the same way as *separation* in the Boids model. In the direction based escape behavior, we simply compute the angle between the heading of the predator and prey. We take the sign of this angle and rotate the heading of the prey by + or − 90°, depending on the sign. The zig-zag escape maneuver simply alternates the prey direction in fixed time intervals. Some of these maneuvers result in patterns shown in Figure 3.

Predators. We have implemented four predator attack strategies, with three presented and simulated in [6]. The first targets the flock's center, the second goes for the closest prey, and the third selects the most isolated prey. In the last strategy, the predator simply attacks a random target.

Figure 3. The patterns emerging from position-based (left picture) and direction-based (right picture) escape maneuvers.

Results

We will first compare the results from the original research [1] with our own. Specifically, we will compare our simulated data with the empirial data obtained from the original research, as shown in figure 4.

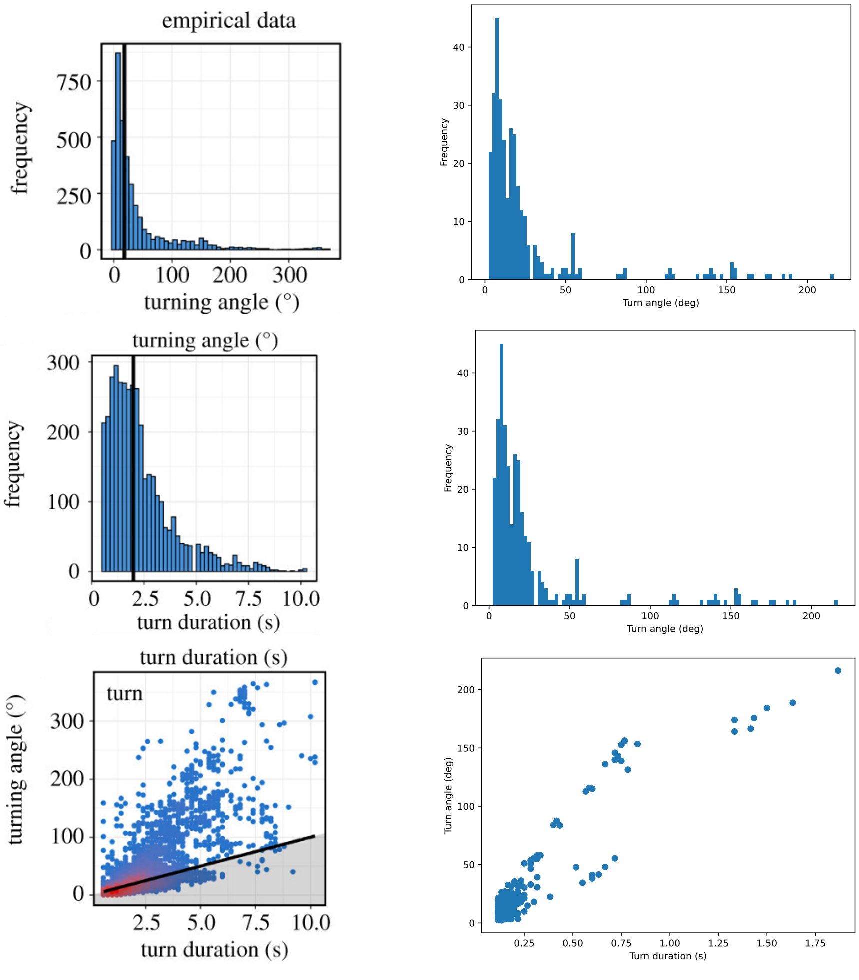
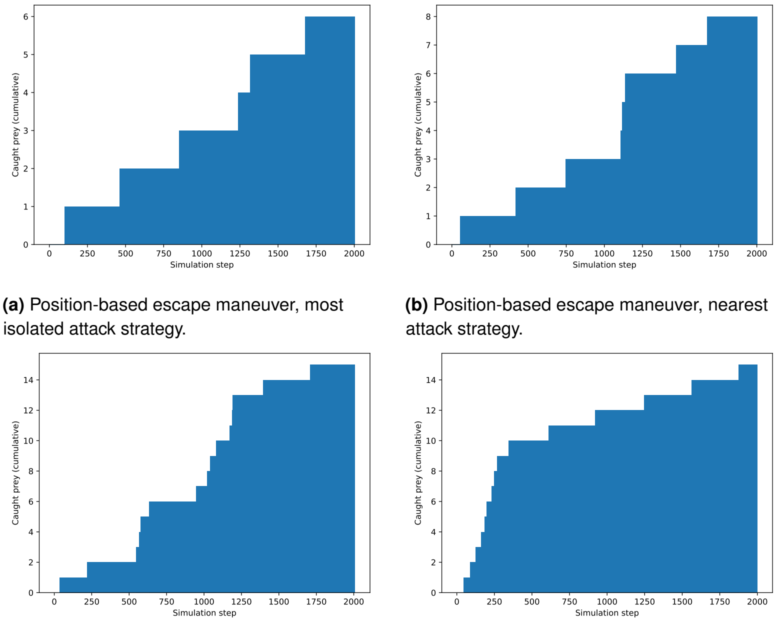


Figure 4. Graphs obtained from empirical data (left) and graphs obtained in our Boids simulation using the *avoid direction* escape manuever (right)

We can see that the resulting graphs are quite similar, except graphs obtained from our simulation are more sparse. This is most likely because we only ran the simulation up to 2000 steps (about 30 seconds), while the empirical data was most likely gathered throughout a longer period. We can therefore be confident that our simulation works as expected.

Lastly, we will compare how different attack strategies perform against different prey escape tactics. Although we have implemented 6 escape manuever tactics and 4 attack tactics, we will 2 of each.

Note that graphs in figure 5 are cumulative, since it makes it easier to see the effect. Plots suggest that direction based escape maneuver is inferior to position based tactics, indicated by larger prey capture count. By inspecting the first row, we can also see that attacking the nearest prey outperforms the most-isolated strategy.



(a) Position-based escape maneuver, most isolated attack strategy.

(b) Position-based escape maneuver, nearest attack strategy.

(c) Direction-based escape maneuver, most isolated attack strategy.

(d) Direction-based escape maneuver, nearest attack strategy.

Figure 5. Comparison of various predator attack strategies and prey evasion maneuvers.

Discussion

We have successfully implemented a basic Boids model with selected modifications and took elements from the HoPE model to implement escape maneuvers. As part of the project, a Python app that simulates predator-prey behavior has been developed.

Potential improvements might include a nicer visualization of the simulation and additional features (such as traces of movement for a particular boid, marked predator targets, . . .). Additional escape maneuvers and attack strategies could be incorporated into the model. We could extend the simulation by introducing multiple predators and implement constant-bearing hunting. In order to speed up the simulation, the application could be re-written to run on a GPU.

CONTRIBUTIONS. Matija Ojo: Added realistic features, fixed escape manuevers, report, Miha Krajnc: Escape manuevers, Janez Kuhar: report, Marko Adžaga: Researching sources and report

Bibliography

1. Papadopoulou M, Hildenbrandt H, Sankey DW, Portugal SJ, Hemelrijk CK (2022) Emergence of splits and collective turns in pigeon flocks under predation. *Royal Society Open Science* 9(2):211898.
2. Parrish JK, Edelstein-Keshet L (1999) Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science* 284(5411):99–101.
3. Heppner F, Grenander U (1990) *A Stochastic Nonlinear Model for Coordinate Bird Flocks.*
4. Oweis S, Ganesan S, Cheok K (2014) Illustration of centralized command and control for flocking behavior. *International Journal of Handheld Computing Research* 5:1–22.
5. Reynolds CW (1987) Flocks, herds and schools: A distributed behavioral model in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques.* pp. 25–34.
6. Demšar J, Lebar Bajec I (2014) Simulated Predator Attacks on Flocks: A Comparison of Tactics. *Artificial Life* 20(3):343–359.