

# Animal Swarming\*

Jan Polivka (ID:51553399),<sup>†</sup> Struan Murray (ID:51550442),<sup>‡</sup> and Daniel Bain (ID:51443692)<sup>§</sup>  
*University of Aberdeen*  
(Dated: March 29, 2019)

Swarming of animals has been known to form complex patterns and this report covers the basic processes required to recreate these forms in a simulation as well as the theory and analysis of these swarms as a form of defense against predators.

PACS numbers: 87.19.St, 87.19.lp

## INTRODUCTION TO SWARMING

The collective movement of animals can produce eye-catching patterns in a seemingly complex yet ordered fashion. (FIG. 1) These patterns can hypothetically be assigned to the similar reactions of the individual animals in the presence of danger, prospect of food or safety. Similar patterns of “Swarming” have also recently been discovered in purely chemical reactions where individual units are self propelled.[?] This suggests that a large proportion of the swarming behaviour may be as a result of very simple rules.



FIG. 1. A classic example of swarming starlings.[?] In this case the swarm has coincidentally taken on a shape resembling a whale.

## STANDARD BEHAVIOURS

A large proportion of behaviour may be interpreted logically, through the base instincts of fear and the requirement of social interaction. The heavy emphasis of social interaction is a result of years of evolution requiring the animals to seek safety in numbers against their predators. The various theories behind swarming range from complex “group minds”[?] to the significantly simplified set of rules which govern what is effectively a stochastic model.[?] These simplified behaviours were classically modeled in 1986 with a program named “Boids”[?] to great success. This program provides a

set of rules to allow the simple simulation of swarming animals.

## Reaction to a predator

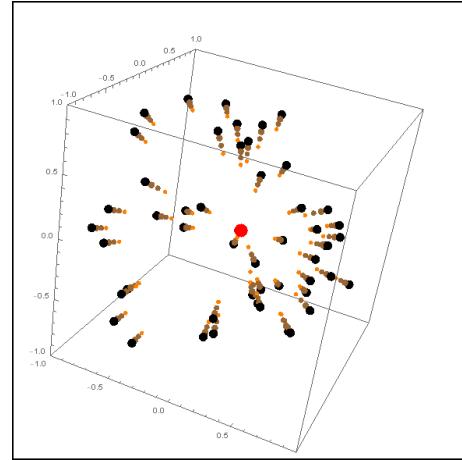


FIG. 2. Several animals (in black) fly from a predator (in red). The velocity at which the creature attempts to escape is greatly affected by the distance to the predator.

The most drastic change to swarm behaviour occurs when a predator, lapse in cohesion or any other danger to the swarm occurs. The natural instinct for any swarm-capable animal when faced with immediate danger is to quickly remove itself from the general path of said danger. This results in a swarm reacting and possibly collapsing in one of the following scenarios:

- : An individual leaves the swarm - The tamest scenario for the swarm, one individual is forced out and made vulnerable but the swarm retains integrity.
- : The swarm splits - If there is a large enough disturbance, but the alignment of most animals remains in sync, the swarm may split into separate smaller swarms which may re-merge at a later time.
- : Chaotic collapse - If the alignment of the swarm is challenged (i.e. a sufficiently large group within the swarm rapidly changes direction) then the swarm

may rapidly expand in a chain reaction and lose the majority of its protective properties (FIG. 2), requiring a complete reorganisation.

### Reaction to a friend

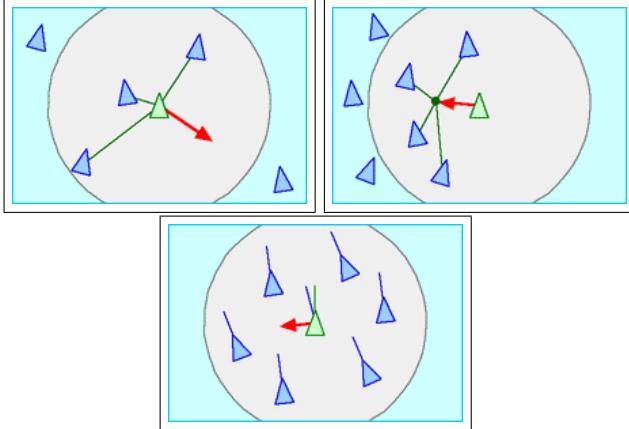


FIG. 3. The main principles of direct flock interaction: Separation, Cohesion, Alignment.[?]

The integrity of a swarm relies on the individual animals being aware of their surroundings, speed and direction. This is almost entirely governed by neighbouring animals in a swarm. This is one of the sticking points in a model not based on OOP principles. Each animal, or represented point has to act independently of the other points and calculation complexity increases to the power of points simulated. To maintain stability, each animal in a swarm must maintain some distance from the animals surrounding it. The animals however must be as closely packed as possible. To ensure these rules are met, the creatures tend to align itself in a similar way to its neighbours. These calculations must be carried out for each animal and in relation to every other animal.

- : Separation - Finding the space to move in the swarm.
- : Cohesion - Bringing the swarm together.
- : Alignment - Allowing the swarm to stay together.

### Reaction to a goal/leader

To generally control the swarm, a target (for example a food source) must be utilised in the model. In the simplest case this may be the centre of the screen (0,0). It may also be a moveable point, possibly representing a desired direction or leader, although studies have shown that swarming lacks a definite leader in almost all cases.[?] The draw to the leader is programmatically defined to be directly proportional to the distance from the target. (FIG. 4)

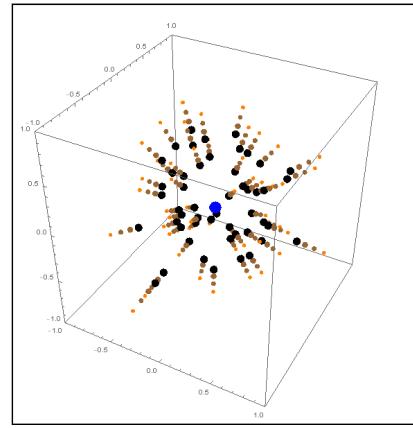


FIG. 4. Several animals (in black) fly towards a goal (in blue). The velocity at which the creature attempts to reach the goal is affected positively by distance.

### OBJECT ORIENTED PROGRAMMING (OOP)

Mathematica does not inherently support object oriented programming, so the majority of the calculations for larger swarms must be solved using matrices. The crux of the matter is that the swarm, leader and predator all behave differently, and so must be programmed as separately as possible. The 3 differently behaving types should also be able to interact, obtaining data from each others positional matrices.

### Heavy OOP method

The first attempted solution involved the strict separation of:

- : Swarming Animals - Known as boids.
- : Leader - To allow a target point, or food source.
- : Predator - To disturb the group of boids.

These individual "Objects" were programmed in the following ways:

1. The boids were programmed to randomly choose another boid to follow, and a boid to avoid. Each calculation round would have the boid move towards the followed boid based on the euclidean distance and away from the avoided boid based on inverse euclidean distance.
2. The leader and predator were programmed to follow intersecting paths to simulate the collision of the hunter and the swarm. In both cases the speed was normalised to be independent of movement path diameter.

The program initially assigns the boids to a random point on a map of fixed size. In each calculation stage, a unit vector, representing the direction of the leader to the boid is calculated. This unit vector is multiplied by a scalar value corresponding to a fraction of the distance between the boid and the leader. The resulting vector is added to each boid's current position.

A nearly identical process occurs for the enemy, however it moves the boid away from the predator. The scalar value also increases with the decrease in euclidean distance, resulting in a large reaction should the predator be nearby.

To calculate the movement between boids, a matrix containing a list of followers calculates a new matrix based on the distance between each boid and the boid it is following. This vector distance is then divided by a scalar value and added to each boid. The same process occurs in reverse for each boid's enemy.

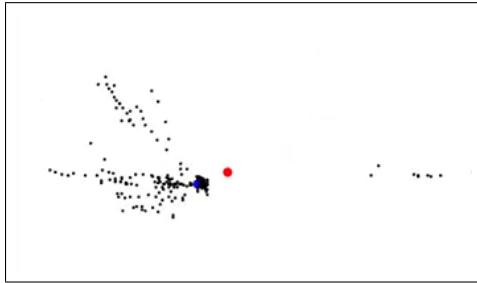


FIG. 5. Several animals form lines around the leader. The predator has succeeded in separating a group from the main swarm.

The working code is shown in 2D Video and 3D Video formats.

### Light OOP method

It was decided that a swarm did not inherently require a moveable target and instead should simply attempt to retain integrity. It is thought that of the 20,000+ known species of fish, half will at some point in their lives form a school.[? ] This led to the development of a second model, loosely based on the first. The behaviour of the predator was also improved to instead pick a specific boid and hunt it down. The interactions also allow the predator to "eat" the boid, removing it from the swarm.

One of the most distinctive properties of an underwater swarm is its use in evading enemies. This is due to the "blindness" of the predator and preyed fish in murky waters. The maximum theoretical distance visible underwater is 80 m, and the furthest observed in calm, clear water is 74 m.[? ] By forming a tight swarm, the predator has a significantly reduced chance of randomly finding the fish than if they were spread throughout an area. (FIG. 6)

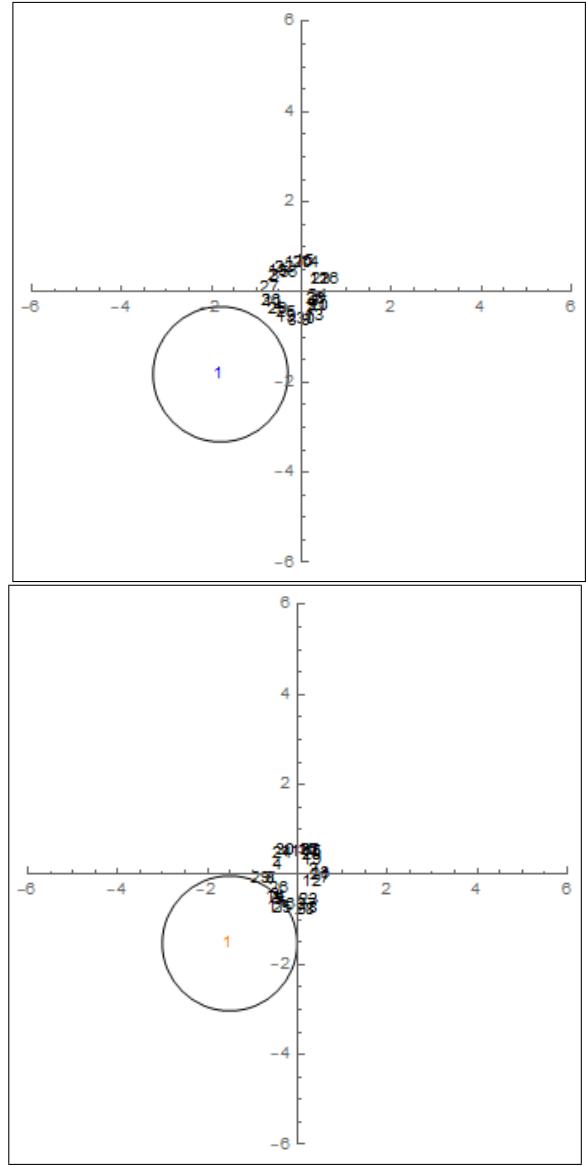


FIG. 6. The effects of swarming mitigating the ability (represented by the black circle) of the hunter to detect the fish.[? ]

To keep the boids on screen, a slight move towards the center of the model each calculation round was added. Other vector components included were random noise and a move towards the center of the group of boids around the boid being calculated. It was decided to use a logarithmic weighting for these calculations, where the closest boid represented the largest influence and a calculation for these weightings was calculated as a constant array, allowing quick application and the ability of the model to run in real time. To gain data on the effectiveness of this swarming and hiding, a "kill" function was added to the hunter, where boids within a certain distance of the hunter would be eaten, and the data for this function recorded for each calculation step, resulting in

a plot of living boids with time.

---

<sup>†</sup> j.polivka.15@aberdeen.ac.uk  
<sup>‡</sup> struan.murray.15@aberdeen.ac.uk  
<sup>§</sup> daniel.m.bain.14@aberdeen.ac.uk

\* Simulation & Theory