# COP 290 Common Project II
Starlings

Rajas Bansal and Animesh Singh

April 2018

## 1 Introduction

Starlings are small passerine birds in the family Sturnidae. Asian species, particularly the larger ones, are called *mynas*. Owing to their lusturous plumage, their family name comes from the Latin word for starling, sturnus. They are considered an invasive species and generally compete for habitat with native birds. Starlings are native to Europe, Asia and Africa, as well as northern Australia and the islands of the tropical Pacific.

Starlings associate in flocks of varying sizes throughout the year. A flock of starlings is called a murmuration. These flocks may include other species of starlings and sometimes species from other families. This sociality is particularly evident in their roosting behaviour. In the non-breeding season, some roosts can number in the thousands. Flocking behaviour is particularly interesting in the case of Starlings as their murmurations are a phenomenal site - like a large organism moving in unison.

In this project, through computer simulations and mathematical models, we try to emulate the flocking behaviour of Starlings. *Flocking* is the collective motion of a large number of self-propelled entities and is a collective animal behavior exhibited by many living beings such as birds, fish, bacteria, and insects. It is considered an emergent behavior arising from simple rules that are followed by individuals and does not involve any central coordination.

Flocking behavior was simulated on a computer in 1987 by Craig Reynolds with his simulation program, *Boids*. This program simulates simple agents that are allowed to move according to a set of basic rules.

## 2 Assumptions

For implementing simulation of murmurations of starlings, some assumptions related to particles and boids are as follows.

1. A boid is not a uniform point. Specifically, it has a complex geometric state and orientation.

2. Boid behavior is dependant on internal and external state.
3. The *internal state* relates to particle parameters.
4. The *external state* is related to the knowledge about other flock members.
5. A neighborhood, or field of view, is used to describe the range of a boid's perception. Most behavioral rules apply based on conditions in the neighbourhood.
6. Individual boids only have a local (limited) knowledge of the flock. All rules take advantage of this local knowledge.
7. The key idea is that *local rules lead to compelling flock behavior*.

# 3 Flocking Rules

Model of murmuration of starlings can be controlled by three basic rules:

1. **Separation** - This rule states that crowding of neighbours should be avoided. There shall be *short range repulsion* i.e. the starlings would maintain a certain minimum distance between themselves to avoid collision. They would not fly into a certain range of the other birds.
2. **Alignment** - This rule states that the starlings would steer towards the average heading of neighbours. This means that a single starling shall head towards the direction in which the flock (as a unit) heads. This ensures *alignment* of the single starling with the murmuration.
3. **Cohesion** - This rule states that a starling shall steer towards the average position of its neighbours. There shall be *long range attraction* i.e. there will be a sense of cohesion among starlings while flying in the murmuration.

Following these three simple rules, the murmuration moves in an extremely realistic way, creating complex motion and interaction that would be extremely hard to create otherwise.

The basic model of *Boids* has been extended in several different ways since Reynolds proposed it.

1. Delgado-Mata extended the basic model to incorporate the effects of *fear*. Olfaction was used to transmit emotion between animals, through pheromones modelled as particles in a free expansion gas.
2. Hartman and Benes introduced a complementary force to the alignment that they call the *change of leadership*. This steer defines the chance of the bird to become a leader and try to escape.
3. Hemelrijk and Hildenbrandt used attraction, alignment and avoidance and extended this with a number of traits of real starlings.

(a) Birds fly according to fixed wing aerodynamics i.e. while rolling when turning. Thus, losing lift.

(b) Birds coordinate with a limited number of interaction neighbours i.e. 7, like real starlings.

(c) Birds try to stay above a sleeping site, like starlings do at dawn, and when they happen to move outwards from the sleeping site, they return to it by turning

(d) Birds move at a relative fixed speed. They showed that the specifics of flying behaviour as well as large flock size and low number of interaction partners were essential to the creation of the variable shape of flocks of starlings.

### 3.1 Rule 1: Cohesion - Boids moves in the direction of the average position of their neighbours.

Boids try to fly towards the centre of mass of the neighbouring boids. This centre of mass would be the average position of all the boids. The term of centre of mass is analogical with the corresponding physical formula, ignoring individual masses and treating all boids as having the same mass.

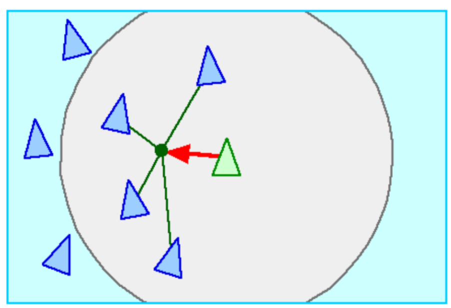

Figure 1: Cohesion

Assume we have N boids, called b1, b2, ..., bN. The position of a boid bJ is characterized by a vector quantity in 3 dimension. The centre of mass is a property of the entire flock and it would not be considered by an individual boid. Hence, a perceived centre is calculated for bJ which is the centre of all the other boids, not including itself. To calculate perceived centre of bJ we use

$$perceived\ centre\ of\ bJ = \sum_{i=1; i \neq J}^{N} \frac{position\ of\ bi}{N-1}$$

Having calculated the perceived centre, we move the boid towards it. To maintain graduality, we move the boid bJ towards 1% of this centre. This displacement vector is given by $1\%\ of\ (perceived\ centre\ of\ bJ - position\ of\ bJ)$.

Thus, we compute the direction to the average position of local flock mates and steer in that direction. The first vector offset for the boid is given by

$$\frac{perceived\ centre\ of\ bJ - position\ of\ bJ}{100}$$

### 3.2 Rule 2: Separation - Pushes boids apart to keep them from crashing into each other by maintaining distance from nearby flock mates.

Boids try to keep a small distance away from other objects (including other boids). The purpose of this rule is to for boids to make sure they don't collide into each other. We look at each boid, and if it is within a defined small distance $\Delta$ of another boid, we move it as far away again as it already is. This is done by subtracting from a vector c the displacement of each boid which is near by. We initialise c to zero as we want to generate a vector which when added to the current position, moves a boid away from other boids near it.

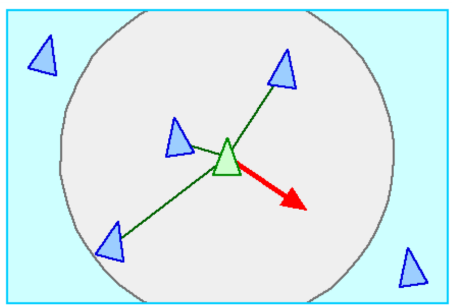

Figure 2: Separation

For a boid bJ, we check for each boid b, if $|position\ of\ b - position\ of\ bJ| < \Delta$ then

$$c = c - (position\ of\ b - position\ of\ bJ)$$

Boids which are very close are not immediately *repelled* because we just double the distance from the nearby boids for each boid. When two boids are near each other, this rule will be applied to both of them. They will be slightly steered away from each other and at the next time step, if they are still near each other, they will be pushed further apart. Hence, the resultant repulsion takes the form of a smooth acceleration. This is designed to maintain a principle of ensuring *smooth motion*. If two boids are very close to each other, it's probably because they have been flying very quickly towards each other, considering that their previous motion has also been restrained by this rule. Suddenly jerking away boids from each other, such that they each have their motion reversed, would appear unnatural (as if they bounced off each other's invisible force fields). Instead, we have them slow down and accelerate away from each other until they are sufficiently enough far apart.

Hence, each boid considers its distance to other flock mates in its neighborhood and applies a repulsive force in the opposite direction, scaled by the inverse of the distance.

### 3.3 Rule 3: Alignment - Drives boids to head in the same direction with similar velocities.

Boids try to match velocity with nearby boids. Assume we have N boids, called b1, b2, ..., bN. The velocity of a boid bJ is characterized by a vector quantity in 3 dimension. The average velocity is a property of the entire flock.

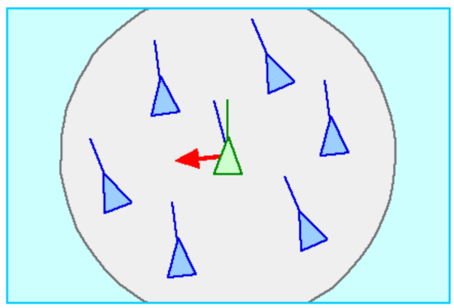

Figure 3: Alignment

Hence, like Rule 1, a perceived velocity is calculated for bJ which is the velocity average of all the other boids, not including itself. To calculate perceived velocity of bJ we use

$$perceived\ velocity\ of\ bJ = \sum_{i=1;i\neq J}^{N} \frac{velocity\ of\ bi}{N-1}$$

Having calculated the perceived velocity, we add a small portion (about an eighth) to the boid's current velocity.

$$\frac{perceived\ velocity\ of\ bJ - velocity\ of\ bJ}{8}$$

Hence, we calculate average velocity of flock mates in the neighbourhood of a boid and steer towards that velocity.

# 4 Steering Rules

## 4.1 Rule 1: Obstacle Avoidance - Steering away from approaching objects.

We assume a cylindrical line of sight. Computing cylinder-sphere intersection, we veer away from any objects in the path. As the velocity vector describes the direction of the boid, it is used to produce a new vector called ahead, which is a copy of the velocity vector, but with a different length.

$$ahead = position\ of\ b + normalized\ velocity \times \vartheta$$

The ahead vector length adjusted with $\vartheta$ defines the range of vision for a boid. The greater $\vartheta$ is, the earlier the boid will start acting to dodge an obstacle, because it will perceive it as a threat even if it's far away.

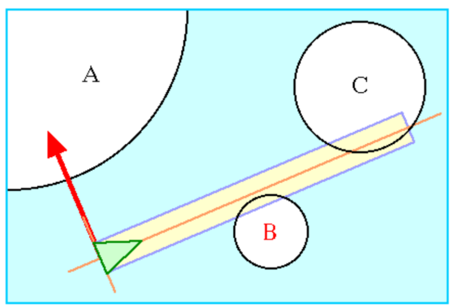


Figure 4: Obstacle Avoidance

In order to check for collision, every obstacle (or its bounding box) must be described as a geometric form. In 3 dimension, every obstacle in the environment can be described as a sphere. To check for collision, the method of line-sphere intersection is used. The line is the ahead vector and the sphere is the obstacle. A collision check is performed to test whether the ahead vector is inside the obstacle sphere by comparing the Euclidean distance between the vector's end and the sphere's center. If the distance is less than or equal to the sphere radius, the vector is inside the sphere and that obstacle is blocking the way of the boid.

The avoidance force must push the boid away from the obstacle, allowing it to dodge the sphere. It is done using a vector formed by using the center of the sphere (position vector) and the ahead vector. We calculate this avoidance vector as follows.

$$avoidance = ahead - obstacle\ center$$

$$avoidance = normalized\ avoidance \times \theta$$

Avoidance is normalized and scaled by $\theta$, which is a number used to define the avoidance magnitude. The greater $\theta$ is, the stronger is the avoidance force pushing the character away from the obstacle.

### 4.2 Rule 2: Goal Seeking - Driving the flock in the direction of a target/goal.

Each boid determines the direction to the goal and then steers in the same direction. Goal setting is used to steer a flock down a set path or in a general direction.

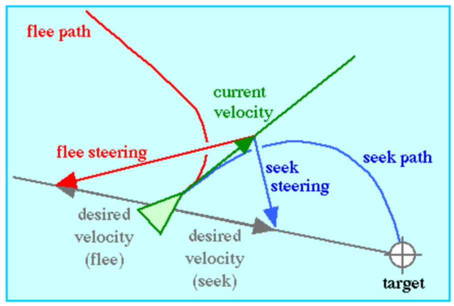


Figure 5: Goal Seeking

A goal can be set for the following phenomena.

1. Action of a strong wind or current

   To simulate the murmuration flying through a strong breeze characterized by a vector v, for each boid b, we add v to the velocity of b.

   $$velocity\ of\ b + wind\ velocity$$

   The same value is added independent of the boid being examined. Hence, the entire flock will have the same push due to the wind.

2. Tendency towards a particular place

   To steer a murmuration through a narrow gate, upon reaching this point, the goal for a particular boid would be changed to encourage it to move away to make room for other boids. If the gate is flanked by impenetrable objects, then the flock will realistically mill around the gate and slowly trickle through it. We move the boid 1% of the way towards the goal at each step. For each boid b, the vector for the target place is updated as follows.

   $$\frac{target\ place\ for\ b - position\ of\ b}{100}$$