STARLING MURMURATION
(Simulation)

COP290

Assignment 2

Ayush Patel (2016CS10396)
Samarth Aggarwal (2016CS10395)

May 2, 2018

Contents

Project 1

Starling Murmuration (Simulation)

1.1 Objective

The main objective of this project is to model and simulate this fascinating phenomenon of Starling Murmuration.

We will computationally simulate the phenomenon by modelling each bird as an independent agent communicating and cooperating with other neighbouring agents. Our objective will be to measure from a realistic simulation the average energy spend by each bird, the angular momentum and the force that each bird has to withstand in a typical flight ritual.

1.2 Introduction

The starlings are generally a highly social family. Most species 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 of birds.

1.2.1 Boids

Boids is an artificial life program, which simulates the flocking behaviour of birds. His paper on this topic was published in 1987 in the proceedings of the ACM SIGGRAPH conference.[1] The name "boid" corresponds to a shortened version of "bird-oid object", which refers to a bird-like object.

As with most artificial life simulations, Boids is an example of emergent behavior; that is, the complexity of Boids arises from the interaction of individual agents (the boids, in this case) adhering to a set of simple rules. The rules applied in the simplest Boids world are as follows:

- **separation**: steer to avoid crowding local flockmates
- **alignment**: steer towards the average heading of local flockmates
- **cohesion**: steer to move toward the average position (center of mass) of local flockmates

1.3 Mathematical Analysis

1.3.1 Basic Kinematics

From the fundamental equation of kinematics, we know that
If,
u = initial velocity of a bird
v = final velocity of a bird
a = acceleration of the bird
t = time taken to get a velocity 'v' starting from 'u'
s = distance travelled in time t, then

$$v = u + a * t$$
$$s = u * t + a * t^2/2$$
$$v^2 = u^2 + 2 * a * s$$

Force

$$F = \delta p/\delta t = dp/dt$$
$$F = m * a$$

Angular Momentum

$$L = rxp = m * v * r$$

Energy

$$E = F.s = m * a.s = m * v^2/2$$

Power

$$P = F.v = m * a.v$$

1.3.2 Cohesion

Boids try to fly towards the centre of mass of neighbouring boids.

Assume we have N boids, called b1, b2, ..., bN. Also, the position of a boid b is denoted b.position. Then the 'centre of mass' c of all N boids is given by:

$$c = (b1.position + b2.position + ... + bN.position)/N$$

Let us define 'perceived centre', which is the centre of all the other boids, not including itself.

$$pcJ = (b1.position + b2.position + ... + bJ-1.position + bJ+1.position + ... + bN.position)/(N-1)$$

Having calculated the perceived centre, we need to work out how to move the boid towards it. To move it 1% of the way towards the centre (this is about the factor I use) this is given by

$$(pcJ - bJ.position)/100$$

Pseudo Code

```
PROCEDURE rule1(boid bJ)
Vector pcJ
FOR EACH BOID b
IF b != bJ THEN
pcJ = pcJ + b.position
END IF
END
pcJ = pcJ / N-1
RETURN (pcJ - bJ.position) / 100
END PROCEDURE
```

1.3.3 Separation

Boids try to keep a small distance away from other objects (including other boids). We take a distance say 'd' units. If two boids are at a distance closer than 'd', then a repulsion force acts on both of them. This repulsive force decreases as the distance between them increases.

Pseudo Code

```
PROCEDURE rule2(boid bJ)
Vector c = 0;
FOR EACH BOID b
IF b != bJ THEN
IF —b.position - bJ.position— ¡ 100 THEN
c = c - (b.position - bJ.position)
END IF
END IF
END
RETURN c
END PROCEDURE
```

1.3.4 Velocity Difference

This is similar to Rule 1 (cohesion), however instead of averaging the positions of the other boids we average the velocities. We calculate a 'perceived velocity', pvJ, then add a small portion (about an eighth) to the boid's current velocity.

Pseudo Code

```
PROCEDURE rule3(boid bJ)
Vector pvJ
FOR EACH BOID b
IF b != bJ THEN
pvJ = pvJ + b.velocity
END IF
END
pvJ = pvJ / N-1
RETURN (pvJ - bJ.velocity) / 8
END PROCEDURE
```

1.4 References

- Flocking (behavior)
- Starling
- Boids
- Simulated Flock of Birds
- How do Boids Work? A Flocking Simulation
- Motion Captured: The Equation of Murmuration
- The Startling Science of a Starling Murmuration
- First Simulation of the Flocking Behavior of Starlings
- Boids Algorithm
- Self-organised complex aerial displays of thousands of starlings: a model
- Role of projection in the control of bird flocks