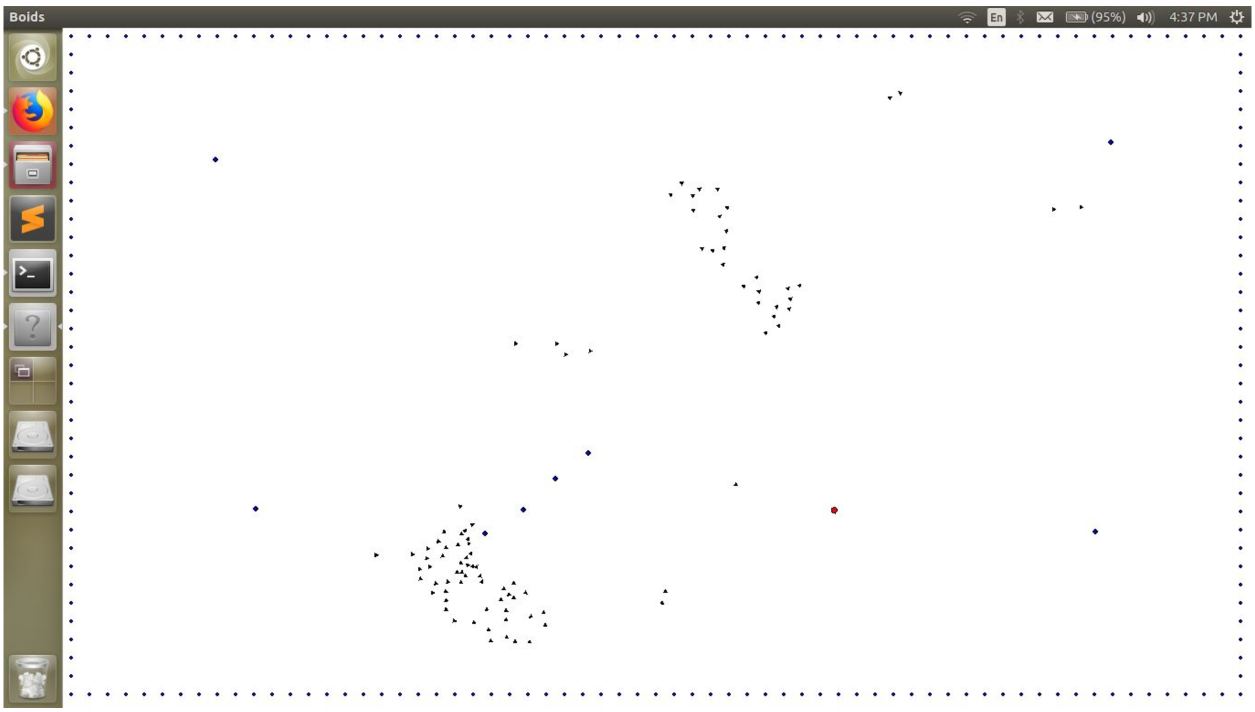
COP 290

Assignment 2

Starling Murmuration

Kamalnath Polakam (2015CS10244)

Akash N (2015EE30522)



Introduction

Flocking behavior is the behavior exhibited when a group of birds, called a flock, are foraging or in flight. There are parallels with the shoaling behavior of fish, the swarming behavior of insects, and herd behavior of land animals.

This program simulates simple agents (boids) that are allowed to move according to a set of basic rules. The result is akin to a flock of birds, a school of fish, or a swarm of insects.

Basic models of flocking behavior are controlled by three simple rules:

1. Separation - avoid crowding neighbors (short range repulsion)
2. Alignment - steer towards average heading of neighbors
3. Cohesion - steer towards average position of neighbors (long range attraction)

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

A basic implementation of a flocking algorithm has complexity O (n^2) - each bird searches through all other birds to find those which fall into its environment.

Cohesion

The 'centre of boids' is simply the average position of all the boids.

Assume we have N boids, called b_1, b_2, ..., b_N. Also, the position of a boid b is denoted b.position. Then the 'centre of boids' c of all N boids is given by:

$$c = (b_1.position + b_2.position + ... + b_N.position) / N$$

Remember that the positions here are vectors, and N is a scalar.

Here the boids consider by a particular boid depends as follows:

1) By a Prey bird: Preys consider the centre of mass of only remaining preys that in a particular range within bird.
2) By a Predator bird: Predators too are attracted towards prey but with a larger search radius.
3) Obstacles: these boids don't attract.

However, the 'centre of boids' is a property of the entire flock;. we prefer to move the boid toward its 'perceived centre', which is the centre of all the other boids, not including itself. Thus, for $boid_J$ (1 <= J <= N), the perceived centre pc_J is given by: (where N is the number of birds within the desired radius)

$$pc_J = (b_1.position + b_2.position + ... + b_{J-1}.position + b_{J+1}.position + ... + b_N.position) / (N-1)$$

Separation

The purpose of this rule is to for boids to make sure they don't collide into each other. we simply look at each boid, and if it's within a defined small distance (say 'd') of another boid move it as far away again a distance inversely proportional to distance. 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 this rule to give us a vector which when added to the current position moves a boid away from those near it.

In pseudocode:

```
PROCEDURE rule2(boid bJ)

    Vector c = 0;

    FOR EACH BOID b
        IF b != bJ THEN
            IF |b.position - bJ.position| < d THEN
                c = c - 100/(b.position - bJ.position)
            END IF
        END IF
    END

    RETURN c

END PROCEDURE
```

We choose to simply chose the inverse of distance from nearby boids, as it means that boids which are very close are "repelled" strongly. Remember that if two boids are near each other, this rule will be applied to both of them. They will be strongly steered away from each other, and at the next time step if they are still near each other they will be pushed further apart but a little gently. Hence, the resultant repulsion takes the form of a strong repulsion which tapers down to 0. It is a good idea 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.

Here too boids consider by a particular boid depends as follows:

1) By a Prey bird: Preys have different proximity thresholds for fellow preys, predators and Obstacles. The lowest threshold will be for the other preys and Highest threshold will be for the other predators. Same will be with the resultant constant multiplied to the acceleration.
2) By a Predator bird: Predators have different proximity thresholds for fellow preys, predators and Obstacles. The lowest threshold will be for the other preys and Highest threshold will be for the other Obstacles. Same will be with the resultant constant multiplied to the acceleration.
3) Obstacles: These repel every boid homogeneously.

Alignment

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

```
PROCEDURE rule3(boid b_J)

    Vector pv_J

    FOR EACH BOID b
        IF b != b_J THEN
            pv_J = pv_J + b.velocity
        END IF
    END

    pv_J = pv_J / N-1

    RETURN (pv_J - b_J.velocity) / 8
```

END PROCEDURE

Here too boids consider by a particular boid depends as follows:

1) By a Prey bird: Preys consider only other fellow preys for alignment.
2) By a Predator bird: Predators consider only other fellow preys for alignment as they want to hunt the preys.

Implementation details:

We like to implement the program in c++ and for graphics part we would like to use 'SFML'

Citations:

http://www.kfish.org/boids/pseudocode.html

https://en.wikipedia.org/wiki/Flocking_(behavior)

https://github.com/jyanar/Boids/tree/master/src