

Project specification

Extending Reynolds' flocking model to allow simulation of sheep in the presence of a predator

Introduction

This project is an attempt to extend Reynolds' classic flocking model¹ to create a simulation of a flock of sheep in the presence of a human-controlled predator. The model made by Reynolds describes a flock of birds moving through three-dimensional space, following a set of simple rules. This model is probably the most well known model for flocking behavior, analyzed extensively by different sources, and even used in movies. In this report we will present a modified version of the Reynolds flocking model.

We modify Reynolds flocking model by modeling sheep in two-dimensional space instead of birds in three dimensional space. In addition to these modifications, we extend the model by adding a non-autonomous predator that the sheep reacts to. Having sheep react to the predator means adding another rule to the model, as described by Delgado-Mata². These modifications and extensions changes the scenario Reynolds flocking model is used in.

With this new scenario, can the Reynolds' model still function as a foundation to build the simulator upon?

An attempt to answer this question will be done by implementing a simulator using our modified version of the Reynolds flocking model, and comparing the behavior of the simulated sheep with the behavior of real sheep. King *et al* performed an experiment where a flock of sheep were tracked with GPS units when being herded by a herding dog³. The resulting data of the experiment will be an excellent source for evaluating our simulation.

Problem statement

The primary questions we hope to answer with this project are:

- Can the Reynolds model be modified to work well with two dimensions?
- Can the Reynolds model be modified to work well with sheep instead of birds?

¹ Craig W. Reynolds (1987). "Flocks, Herds, and Schools: A Distributed Behavioral Mode"
<http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/> (2013-02-22)

² Delgado-Mata C, Ibanez J, Bee S, *et al.* (2007). "On the use of Virtual Animals with Artificial Fear in Virtual Environments". *New Generation Computing* 25 (2): 145–169

³ Andrew J. King, Alan M. Wilson, *et al.* (2012) "Selfish-herd behaviour of sheep under threat". *Current Biology* 22 (14): 561 - 562

- Can the Reynolds model be modified to work well with the introduction of a predator?

Approach

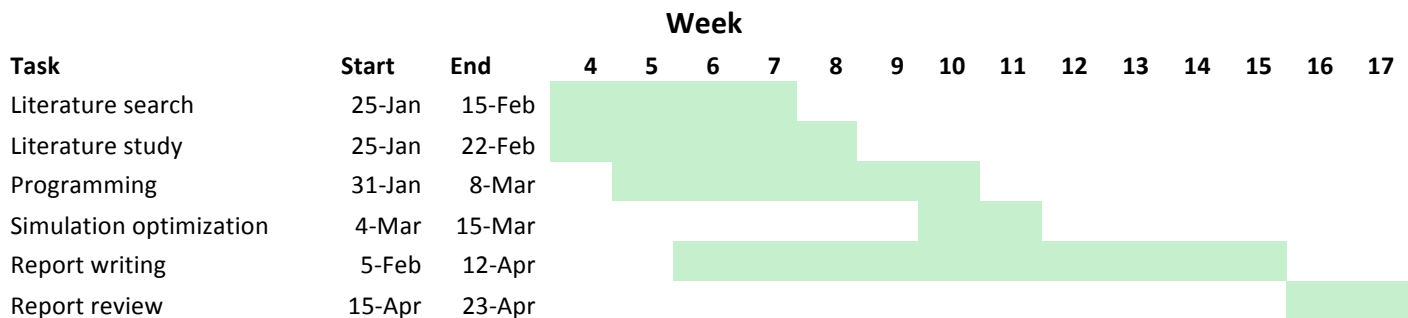
The simulator will be written in Java, using the Swing graphics library. All of the simulation will be done on a Canvas⁴ where the sheep are represented as circles and with the direction they are facing clearly shown. The predator will also be displayed as a circle but with a separate color to distinguish it from the sheep. When moving the predator the sheep will act according to our modified version of Reynolds flocking model. The movement of the sheep will be done in small discrete time steps and simultaneously for all sheep.

The position of the sheep will be tracked by the program and saved. This will allow us to plot out graphs similar to those by King *et al* and allow us to compare how close our model is to theirs.

References

- Delgado-Mata C, Ibanez J, Bee S, *et al.* (2007). "On the use of Virtual Animals with Artificial Fear in Virtual Environments". *New Generation Computing* 25 (2), p 145–169
- Andrew J. King, Alan M. Wilson, *et al.* (2012) "Selfish-herd behaviour of sheep under threat". *Current Biology* 22 (14), p 561 - 562
- Temple Grandin, Mark J. Deesing. "Genetics and Behavior during Handling, Restraint, and Herding". <http://www.grandin.com/references/cattle.during.handling.html> (2013-01-28)
- Reynolds, C W (1987). "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Computer Graphics*, 21 (4) (SIGGRAPH '87 Conference Proceedings) p 25-34
- Parent, R, "Computer Animation: Algorithms and Techniques", *Morgan Kaufmann*, San Francisco, 2002, p 241-242
- Everham, J D, Ruiz, V F (2011). "Experimental Analysis of the Reynolds flocking model", *Paladyn* 2 (3), p 145-155

Time plan



⁴ Java Standard Library API, Canvas class, <http://docs.oracle.com/javase/6/docs/api/java/awt/Canvas.html>