

Assignment 4: Swarm Simulation using Quadrees

Jens Fredskov (chw752)

October 6, 2013

1 Introduction

The following report describes the implementation and testing of a quadtree library and fish farm simulation in Erlang.

2 Implementation

2.1 The Quadtree

The quadtree library has been implemented in the file `quadtree.erl`. The functions used for the coordinator, nodes and leaves all communicate with asynchronous messaging. This e.g. means that we do not wait around for an element to be inserted, by just send the message and move on.

The children of a node is contained in a list of tuples, where each child is matched with its respective corner, allowing us to easier decide where messages should be sent.

When a leaf reaches its limit, it converts to a node by spawning four new empty leaves, and then afterwards sending adds to these with the correct elements.

When the tree applies a function (with `MapFunction/3`) the function propagates down through the tree to all leaves which have elements within the specified bounds. After applying the function if all elements in a leaf which are out of bounds, will be sent to its parent as an add. In this way the elements move up through the tree and down to the new leaf where it should be added.

2.1.1 The Fish Farm

The fish farm has been implemented in `swarm.erl`. The implementation of the fish farm is not yet complete, as the majority of time was spent implementing the quadtree library.

3 Testing

It was not possible to get the given printer example to work.

As most of the time was spent implementing the tree only “on the fly” tests have been made using `io:formats` to print the contents of the tree. These all seem to have been correct, but as no proper automated tests have been made and supplied, we cannot say much about the correctness of the implementation

4 Conclusion

We have now described the implementation of the quadtree library and fish farm simulation. We have also described how the implementation has been tested and have concluded that we cannot say much about whether the implementation is correct.