Swarm Intelligence

1 Emergent Behaviour

So far, we've looked at environments in which there has been only one reactive agent. Here, we consider what happens when there are several such agents at large in the world.

Some of the more interesting cases arise when there is a large number of reactive agents in the environment. These agents will sense the presence of other agents and this will partly determine their actions. While each agent has only very simple behaviour, we may see patterns in their collective behaviour. For example, we might observe that, together, the agents form and move en masse (like a flock of birds, swarm of insects, school of fish or herd of wildebeest).

Here's an example of this idea. Suppose each reactive agent is governed by the following behaviour:

> Identify two members of your flock that are near to you and try to stay close to them, but not too close.

These agents will 'swarm'. (See demo in lecture/on web site.)

Suppose we allow two types of agent into the environment: locals and strangers. Locals are additionally governed by the following behaviour:

> When you spot a stranger, go after him; if you are close enough, attack.

Strangers are governed by the following behaviour:

> If locals chase you, run away.

The locals will swarm, but they will chase and attack strangers in a concerted fashion. (See demo in lecture/on web site.)

These agents are called Floys, and they provide a simple example of the idea. (See the module web site for links to their originator.)

Boids are a more famous agent that, collectively, exhibit flocking behaviour. They are governed by three behaviours. (Different implementations use different behaviours; the ones below are not necessarily the most standard ones, but they are illustrative of the ideas and they are also the ones used in the demo in the lecture, which is linked to from the module web site).

- Boids try to fly towards the average position of neighbouring boids;
- Boids try to keep a small distance away from other objects (including other boids); and
- Boids try to match their velocity with the average velocity of nearby boids.

(See demo in lecture/on web site.)

We'll look at how this kind of thing is implemented. You'll be able to discern the sense/plan/act cycle. These agents sense the world (typically to find out what other agents are in the vicinity), they decide on an action (typically how much to move along each axis) and then they act (their position in the world is updated).

Here's some pseudocode. At each moment of time:


```
for each boid, b
{   COMPUTENEWPOSITION(b);
}
```

Suppose each boid b is an object whose instance variables include one called $\mathbf{position}$ and one called $\mathbf{velocity}$. Both of these variables are vectors (arrays) of length 2 (for 2D simulations) or 3 (for 3D simulations). In the case of 2D simulations, $\mathbf{position} = \langle x, y\rangle$ and $\mathbf{velocity} = \langle \Delta_x, \Delta_y\rangle$ (indicating by how much the x and y values should be changed to compute the boid's new position).

We're going to have to be able to do computations with vectors (arrays) such as addition/subtraction of two vectors, multiplication/division of a vector by a 'scalar' (a number!) and comparisons of two vectors. The definitions are very obvious, but for those of you who need to have them spelled out, below I show how they are defined in the case of vectors of length 2:

$$\begin{array}{rll}
\mathbf{v_1} + \mathbf{v_2} & =_{\text{def}} & \langle v_1.x + v_2.x, v_1.y + v_2.y\rangle \\
\mathbf{v_1} - \mathbf{v_2} & =_{\text{def}} & \langle v_1.x - v_2.x, v_1.y - v_2.y\rangle \\
\mathbf{v} \times n & =_{\text{def}} & \langle v.x \times n, v.y \times n\rangle \\
\mathbf{v}/n & =_{\text{def}} & \langle v.x/n, v.y/n\rangle \\
\mathbf{v} < n & =_{\text{def}} & v.x < n \wedge v.y < n \\
\text{abs}(\mathbf{v}) & =_{\text{def}} & \langle \text{abs}(v.x), \text{abs}(v.y)\rangle
\end{array}$$

A boid's new position is computed as follows:

```
Algorithm: COMPUTENEWPOSITION(b)
Vector changes1 := BEHAVIOUR1(b);
Vector changes2 := BEHAVIOUR2(b);
Vector changes3 := BEHAVIOUR3(b);
b.velocity := b.velocity + changes1 + changes2 + changes3;
b.position := b.position + b.velocity;
```

In other words, we compute the changes resulting from the three behaviours, add them to the current velocity to get the new velocity, and add this to the current position to get the new position. (Note that extending this algorithm to more than three behaviours is easy.)

Now for the behaviours.

- Boids try to fly towards the average position of neighbouring boids.

```
Algorithm: BEHAVIOUR1(b)
Vector sumOfPosnOfOthers;
int n := 0;
for each boid c in the world
{   if c ≠ b
    {   sumOfPosnOfOthers := sumOfPosnOfOthers + c.position;
        n := n + 1;
    }
}
Vector avPosnOfOthers := sumOfPosnOfOthers/n;
Vector changes := (avPosnOfOthers − b.position)/100;
return changes;
```

This computes the average position of the other boids and moves boid b 1% of the way towards this average position.

- Boids try to keep a small distance away from other objects (including other boids).

```
Algorithm: BEHAVIOUR2(b)
Vector changes;
for each object o in the world
{   if o ≠ b
    {   if abs(b.position − o.position) < 100
        {   changes := changes − (b.position − o.position);
        }
    }
}
return changes;
```

If other objects are within 100 units away, boid b is moved away from them by the same amount again.

- Boids try to match their velocity with the average velocity of nearby boids.

```
Algorithm: BEHAVIOUR3(b)
Vector sumOfVelOfOthers;
int n := 0;
for each boid c in the world
{   if c ≠ b
    {   sumOfVelOfOthers := sumOfVelOfOthers + c.velocity;
        n := n + 1;
    }
}
Vector avVelOfOthers := sumOfVelOfOthers/n;
Vector changes := (avVelOfOthers − b.velocity)/8;
return changes;
```

This computes the average velocity of the other boids and changes boid b's velocity one eighth of the way towards this average velocity.

When we observe boids (or real birds) flying as a flock, it is tempting to assume that there is a leader whom the other boids are following. This might explain how the flock arises: by following the leader, they all fly in the same direction, they all turn at the same time, etc. But, of course, this is *not* how flocks arise. Instead, as we've seen, each boid is a simple autonomous agent; each boid exhibits some simple behaviours; and, crucially to the formation of flocks, some of those behaviours are quite 'social' behaviours (e.g. matching your velocity with that of your neighbours). The formation of the flock is a by-product of the collective behaviour of the individual boids.

This is a case where, to use a cliché, 'the whole is more than the sum of its parts'. When a 'higher-level' behaviour or property comes about in this way, we describe it as an *emergent behaviour* or *emergent property*.

Let's look at another example.

2 Simple Communication

This lecture has introduced the idea of having more than one agent in the world. And so far, these agents have been able to sense and react to the presence of other agents. But, a natural capability we should consider at this point is *communication* between agents.

We need to be careful here. We need to realise that the kind of communication we can get from a reactive agent is generally a fairly impoverished form of communication. It tends to be implicit and unintended, and it tends to happen indirectly through the effects of the agents' actions in the environment.

For example, as it moves around its environment, an agent might secrete some substance (e.g. a spray of scent or a trail of slime) that can be sensed and responded to by other agents. In biology, such a substance is called *pheromone*. In this way, one agent implicitly, unintentionally and indirectly communicates information about where it has been to other agents.

Or, as another example, one agent, on sensing the presence of some significant object in its environment (e.g. a predator or a food source) might emit a noise. In this way, the agent, again largely unintentionally, communicates information to other agents who are able to hear that noise.

In contrast to this impoverished communication between reactive agents, agents with greater intelligence may be capable of more explicit, intentional and direct forms of communication. They may explicitly decide whether to communicate, to whom to communicate, what they will communicate, how to communicate and when to communicate. They deliberate rather than react: they work out, in their heads, what the likely effects of their communication will be on their audience, and make their communication decisions in a way that they hope will have desired effects. Since we are currently looking only at reactive agents, we cannot explore this richer form of communication at this point in the module.

Even impoverished forms of communication can lead to interesting emergent behaviour. We'll look at one example of this. Consider ant-like agents that forage for food. When they find food, they return to their nest. (They find their nest again by its smell.) When they're carrying food back to the nest, they secrete quantities of pheromone. Other agents who are foraging for food, when they detect pheromone, will follow the pheromone trail in the direction in which it is at its strongest. Hence, we have here some implicit, unintentional, indirect communication.

It looks a little as though the agents coördinate their foraging. It's as if there is centralised control. The colony of agents appears to exploit nearby food sources until the food is exhausted, and then it turns its attention to more distance food sources. But, of course, this seemingly coördinated behaviour is an illusion. It is an emergent property of the simple behaviour of the individual agents interacting (and indirectly communicating) with each other. (See demo in lecture.)

3 Artificial Life

If biology studies life by *analysing* life that is to be found on Earth, then *Artificial Life* or *ALife* studies life by *synthesising* it (creating it). Interestingly, this may help us to understand not only life-as-we-know-it, but also life-as-it-could-be. It is a broad, multi-disciplinary field, drawing many ideas from Computer Science and AI, as well as from biology, chemistry, physics, economics, etc.

The systems we've looked at in this lecture are all examples of ALife. It is perhaps true to say that ALife is very much concerned with emergent behaviour as this seems to be a dominant feature of natural life. (ALife is also very much concerned with evolution and learning.)

ALife research can help us to understand properties of all sorts of systems, quite apart from those involving animals such as birds and ants. For example, its techniques can be used to study the formation of traffic jams, or the formation of 'corridors' of pedestrians in crowded high streets, or the operation of stock markets.

It has also had industrial applications. For example, ant colony food foraging has inspired algorithms for optimisation problems, such as finding the shortest paths between all nodes in a network, which can be applied to real tasks such as telecom routing. (See next lecture.) More sexy examples are to be found in the entertainment business. The bats in *Batman Returns* and the wildebeest stampede in *the Lion King*, for example, were animations based on the behaviours of boids.

4 The Mind as a Society

So, what are the relationships between AI and ALife? There are several.

Firstly, ALife has borrowed techniques from AI: neural nets and GAs, for example. This gives us an overlap in the things we research and use.

Secondly, AI has an interest in systems where there are many agents. (AI people uses phrases such as 'distributed AI' and multi-agent systems' when referring to this work.) And ALife is also interested in such systems. If there is a difference here, it is that AI *tends* to be more interested in the case where the agents within the system have rich, complex behaviours, whereas ALife *tends* to be more interested in the case where the agents have simple behaviours.

The third relationship is the most thought-provoking. Ideas coming from ALife might help to explain human intelligence. Some people believe that our intelligence (or properties of our intelligence, such as our supposed consciousness) are emergent properties, arising from the collective behaviour of interacting low-level (simple, unintelligent) agencies. This is sometimes summed up as a slogan: *the society of mind*.

If this is right, it might even be a challenge to the way that AI research is carried out. Some would argue that AI research is too top-down and is focused too much on attempts to directly model complex behaviours. They argue that the right methodology is more bottom-up with a focus on simple components, emergence, evolution and learning. (However, in my opinion, modern AI is more bottom-up than it ever was anyway. The structure of this module reflects this.)