Evolving Behavioural Animation Systems

David Griffiths[1] and Anargyros Sarafopoulous[2]

National Centre for Computer Animation, School of Media Arts and Communication, Bournemouth University, Talbot Campus, Fern Barrow, Poole, Dorset BH12 5BB, United Kingdom.
dave@blueammonite.f9.co.uk Asarafop@bournemouth.ac.uk
http://www.blueammonite.f9.co.uk/alifeart

Abstract. One of the features of Artificial Life (AL), is its ability to cross boundaries between traditionally separate disciplines. While its foundations are in biological research and computing, its visual nature has implications for the fields of art and entertainment. Using a continuous genetic algorithm, adaptive autonomous agents can explore user created environments. If these agents have pressures of 'natural' selection imposed on them, they can exploit the environment and create simple solutions to survive. When the environment becomes complex enough, the emergent solutions can, in turn gain in complexity, leading to unexpected and visually pleasing results. We produce animation sequences whose content/aesthetic is defined by the foraging and mating behaviour of simulated agent colonies.

1 Introduction

Artificial Life systems represent a promising method of creating visually interesting and organic animation. A wide variety of AL systems have been created for many uses. The first examples of agent based AL were purely abstract, inhabiting one-dimensional environments (Tom Ray [5]). As these were difficult to visualise, they were later expanded into two-dimensional forms. Sims [6] concentrates more on genetically generating individual movement and behaviour in a physically based world, while Yeager [9] creates ecologically based environments to contain populations of autonomous agents. The visual content of such systems are arguably their most important component, as they are the only way to understand the dynamics of their behaviour, and draw comparisons with their natural world counterparts. To date, systems written to fully explore the inherently visual properties of these techniques have been restricted to the evolution of texture [7] and shape, Todd and Latham [8].

The artificial life simulation detailed here, is designed to be used as a tool for exploring the visual properties of emergent behaviour. The focus of this simulation are the creatures which inhabit it. The environment is where all the behavioural elements are combined, and where the creature's survivability is put to the test. This survivability is the driving force behind the evolutionary algorithm, which is based on a GA with fixed length integer representation. The creature's behaviour is encoded in an artificial neural network and the resulting animation is a visual expression of this data, through the movement and extra detail via the shaders.

2 The Environment

The environment entirely defines the behaviours of the creatures, which evolve within it. The environments used here consist of several simple elements for the creatures to interact with. (see fig. 1) These elements are designed to present problems which are an broad abstractions of those found in the real world. It is hoped that this strategy will encourage actions in the creatures which are recognisable by us. The elements in the environment are food and barriers, creatures are free to roam in this environment in three dimensions, but may not intersect each other. Food is represented by small spherical objects, their distribution is interactively defined. Barriers are rectangular objects, which impose a small penalty on the creature's energy supplies if they collide. Barriers may be used to contain the creatures in a finite space, or to break up the environment in a more complex way.

Fig. 1. An Example Environment.

Colour is an extremely important part of the environment. In fig 1, barriers are shown as semi-transparent blocks, the small objects are food and the other multicoloured objects are creatures. All objects have a colour assigned to them, it is the only way the creatures can differentiate between different objects.

A creature's handling of the energy flow of the environment largely dictates it's implicit fitness. All energy in the environment comes from the food (apart from that contained in the first generation of creatures) so to obtain energy to live and survive; a species has to learn to eat.

The other source of energy available to the creatures is obtainable through fighting. If a creature activates its fighting action, it may be able to take a 'bite' from its prey's energy store. This way, energy can flow from creature to creature. This method of predation is implemented to encourage specialisation into predator and prey.

The maximum energy that a creature can store is set by it's size, the larger the creature, the more energy it can carry.

There are two modes of energy use in an environment:

1. "Energy loss environment", where energy is constantly being used up by creatures - and has to be constantly replaced. This is the most physically accurate.
2. "Circular energy environment", where there is no energy loss. All energy used by a creature is deposited in food when and where it dies. This approach seems to encourage very lazy creatures which live together in densely packed groups, the reason being, that they don't ever have to move very far to get energy.

Energy is used up in most of the creature's actions. Firstly, there is an energy price taken off all the creatures every frame. This price is proportional to the creature's size. Energy is also taken off when a creature moves, relative to the speed of the movement. When a creature gives birth to its children, an amount of the parent's energy is given to each of the offspring (this amount is defined in the genome), if this value is too high, the drain of energy will kill the parent. This should not be seen necessarily as a disadvantage; the genes of the parent will have been passed on. If this value if too low, the young creatures will have very little time to find food before they die.

3 Agent Architecture

The visual architecture of a creature is used as a method of displaying its individual attributes. There are three main attributes, which dictate a creature's abilities. Action distance, or reach, Sense distance, or how far it can see, and maximum speed. These attributes are defined in its genes by a ratio to encourage specialisation, and are expressed by scaling the creature's three body pods. The three representational pods are shown in fig. 2, and represent the following:

Clockwise from top right on the creature are the spiky action pod, smooth sense pod and rippled motion pod. Holes in the surface indicate energy loss. The effects of selection are clearly visible through the shaders, which change the surface appearance, and also the final geometry of the creatures. These shaders are part of the renderer.

Aggression is expressed through the use of a spiky displacement shader on the creature's action pod. Movement expressed through a rippling displacement shader on the creature's transport pod, which simply projects a sine wave through the object. The speed of the rippling is modulated by the current speed, and the wavelength is also modulated by an arbitrary gene connection.

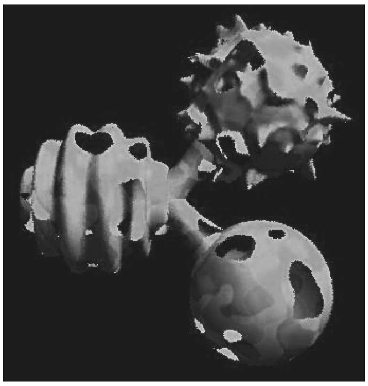

Fig. 2. Creature showing the shaders used to indicate it's attributes.

The Creature's surface shader was the most complex shader in the system (see section 6). It incorporates the age, energy level and colour of the creature. The age of a creature is visualised by interpolating between different shading functions. With a young creature the shader is even, as it ages, the shader becomes distorted. Using "toon shader" techniques, the shader also gradually loses its smoothness, and eventually comprises three colours. A transparency component is used to show the energy level. As a creature loses it's energy, semi transparent holes appear on the creature, which grow until either energy is replenished, or the creature becomes completely transparent, and dies from lack of energy. To add to the creature's pictorial individuality, various arbitrary genes map to shader attributes, which control colour distortions on the surface of the creature. This serves the purpose of increasing the visual indications of mutations and variability.

3.1 Behavioural Modeling

The only information a creature has of its environment is the colour and relative position of the nearest object. This means that to be successful they must associate certain objects with certain colours, and actions.

There are 4 basic building blocks of the creature's behaviour; some of them are necessary for the creature to learn before it can survive on its own, others are more optional behaviour.

1. Eating
   This is the first thing that a species has to learn, as it must gain energy to move about its world.
2. Mating
   If a species is to continue to adapt and evolve then mating is a necessary

action to learn. It is also beneficial (or though not vital) to learn to recognize members of it's own species.
3. Fighting
Fighting is not necessary for survival, although it is a secondary resource of energy. Some species practice cannibalism to some extent.
4. Signaling
Signaling does not have any immediate benefit for a creature, but it can lead to some of the most interesting results. To signal, a creature changes it's surface colour to that of the signal colour stored in its genome.

3.2 Artificial Neural Networks

As in existing artificial life systems [9], Artificial Neural Networks (ANN's) are used to control the creature's behaviour. The ANNs implemented here, are versions of feed forward and recurrent networks. The difference between other such implementations is the lack of a learning function. When a creature is born, its ANN is hardwired, and cannot be changed. Despite this limitation, the GA is able to select creatures with successful modifications in their ANN topology.

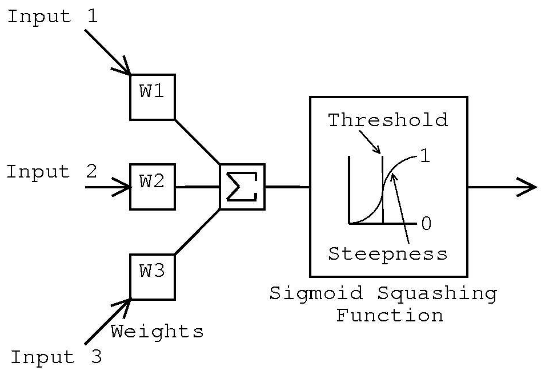

Fig. 3. Neuron Diagram.

These neurons are networked together as described in section 3.2.2 to form ANN's with eight inputs and eight outputs and seventeen neurons. The amount of neurons was fixed, due to the nature of the genome representation. The small amount of neurons helped with the speed of the simulation. The ANN inputs simulate the creature's perception of the world, and are:

1. Relative Angle of nearest object.
2. Colour of nearest object.
3. Current energy level.

4. Current age.

The creature is also given information on its internal status, how much energy it has left (or how hungry it is) and it's current age. Four outputs control the direction and speed of the creature, and four outputs are dedicated to the action selection of the creature, which simply uses the highest activated output to select the action (Eat, Mate, Fight, Signal).

Fig. 4 shows an example ANN taken from the net view window, inputs are at the top, outputs at the bottom. Activated neurons are indicated by their brightness. To add a little realism to this simple structure, a small amount of noise was added to each of the inputs

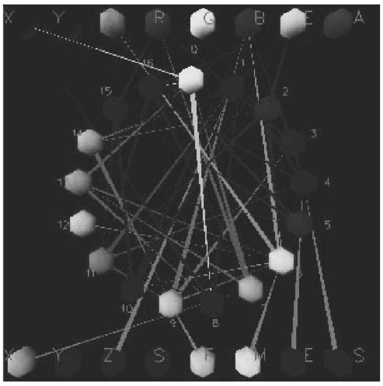

Fig. 4. Neural net visualisation.

3.3 ANN Genome encoding

The genome for each creature supports a description of seventeen neurons, each using three weighted analogue inputs. The threshold value and steepness of the sigmoid squashing function are contained within the genetic representation for each neuron.

The attributes section details other factors of the creature's description, such as signal colour, and number of children - these do not directly affect the ANN. Each ANN has eight inputs and outputs. The topology stored in the genome describes the source of each neuron input and ANN output. The neuron definitions were separated in this way to allow meaningful crossover, which would not be liable to break the networks. The ANN's seemed quite resilient to mutational changes in their structure.

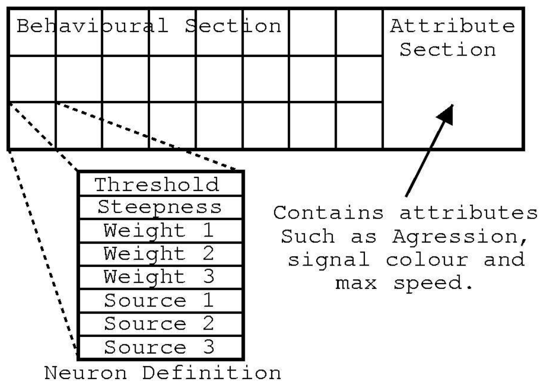

Fig. 5. Genome mapping, incuding the ANN description.

3.4 ANN Topological approaches

Two different methods were used to describe the architecture of the ANN, the first was a fairly rigid two layer ANN. The first eight neurons were connected exclusively to the inputs of the ANN, while the second layer were permitted to connect to the first layer or the inputs (to allow for jump ahead connections). The outputs connected to the second layer. This structure allowed the GA to start with a fairly promising topology, and meant that creatures of early generations were generally relatively fit. However, this method was too inflexible to allow interesting structures to be found, and it also lacked the temporal features of a recurrent network.

The second method allowed a much more freeform architecture, including recurrent connections. Any neuron could simply connect to any ANN input or any neuron output. ANN Outputs could connect to any neuron output. This structure allows neurons to feed back into themselves via other neurons, which creates temporal properties, neurons may oscillate, or fire off each other in complex ways. This can also be seen as a very simple form of memory.

4 Evolutionary Algorithm

A species of creature becomes viable when it is capable of gathering enough energy to mate and reproduce its genetic code. Until this point, help is given via a GA style fitness function. Fitness is awarded incrementally for various activities: choosing any action, eating an item of food or successfully reproducing. If no viable creatures exist, the environment will eventually empty and the fittest creature can then be reloaded into a new generation with mutations.

The mating function activated by a viable creature incorporates crossover and mutation to create the child from the parent's genomes. Once this becomes the sole method of creature reproduction, the explicit fitness function becomes redundant, and the evolutionary method can be seen as modeling "natural" selection, i.e. an implicit selection based on reproduction.

5 Visual Representation: 3D real-time display

The three dimensional environment is displayed in real-time for previewing creature's behaviour. The viewing camera may be moved around the environment interactively, or set to follow an individual creature for more detailed examination of behaviour. Extra information can also be shown, such as the sense and action distance for each creature. Graph, population viewing and family tree view windows are all part of the real-time system.

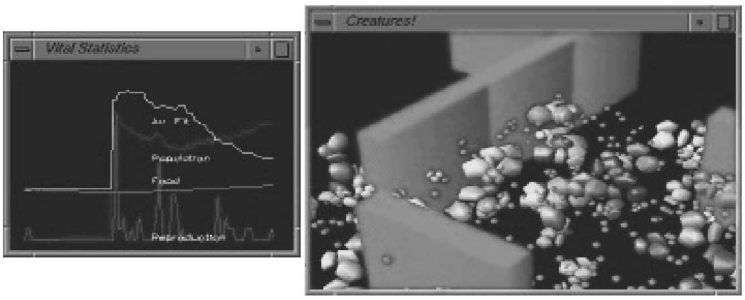

Fig. 6. Statistics window and Preview window.

6 Rendered Output

Final output for the animation is a high quality rendering. Each creature is represented by three primitive spheres and two hyperbolas, which connect them. The creatures are all shaded using their genetic textures. Barriers are shaded as semi transparent to allow creatures to be seen behind them. Fog can be used to represent their general viewing distance. The system communicates all the gene information to the renderer, and converts it into shader arguments.

7 Results

The initial purposes of this system were to generate lifelike animation, which had an interesting visual quality. However, the effectiveness of the solutions found can be examined in a more rigorous benchmark foraging test. This test consists of a random path of food, which provides an accurate idea of how effective the individual's foraging strategy is.

Another indication of effectiveness of a species is through the ecology of the environment. If the amount of food in the environment is low, while the amount of creatures or the reproduction level is high, then this either reflects favorably on the species, or tells us that the environment is too easy. Even in a situation where the environment is too easy, progress will still be made, as competition between the creatures will be high.

Fig. 7. An example species bred in an environment containing barriers.

Many species have been evolved; most of the interesting ones are described below. It was surprising that from such a simple system, so many solutions and traits emerged.

The first problem to be overcome was that of gaining energy, or feeding. Two broad strategies were employed, one of random path following, in the hope that food would be accidentally found at some time, and one of active food foraging. The creature that seemed to do this best (the "Blue forager") was tested in the benchmark environment. The mechanism it uses is to look around rapidly until an object is sighted. If this object has a high green colour channel then the creature will fire it's forward neuron, and activate its desire to eat. It is not known at what point in the generation this fairly simple strategy appeared, but in subsequent tests, it has occurred again. The effect this has on a population in an environment of randomly placed food (which exist inside a certain boundary), is to prevent the individuals from ever leaving the area of food.

Interestingly, in tests where creatures are evolved from scratch in the benchmark environment, they rely on the patterns in the environment. Without randomness, they tend not to develop their sensory input data.

The more advanced problems are much more difficult to test, as they involve creature-to-creature interaction. The viable species evolved differed radically in their effectiveness, their main objective was to be able to decide how to prioritize their energy and reproductive needs. One strategy observed, was based on what colour the nearest object was, while another was to use it's internal energy status to decide what to do next.

Most of these viable species were happiest when grouped together quite closely, for obvious reasons, it is much easier to locate a mate if everything is close together. Attempts at breaking up the environment with barriers resulted in more diversification, due to the isolation of groups, but didn't alleviate this problem. Some of the best species evolved were the direct descendants of the "Blue foragers" described above, who adapted their foraging to mating.

One strategy, which emerged briefly at an early stage of the system, was an intriguing use of the abilities to signal (this may still be used, as signaling is very common). If a creature contained a high level of energy, it would activate its signal neuron. At the same time, surrounding creatures would attract to this signal colour and use it as an indication of mating potential. The result, that creature's save energy by not trying to mate with others with low energy.

The kind of animation produced by these species is very similar to the movements created by colonies of bacteria; early primitive creatures often travel in abrupt jumpy movements while more refined species smooth out their paths. Some results reflected those by Yeager [9], such as the grouping together of lazy creatures. It was possible to interactively develop the creatures indirectly by changing the environment while the simulation was being run. For example, if a population develops in close proximity, the distribution of food can be increased, making the environment more difficult and encouraging more active behaviour. As the simulation runs in real-time this adds a level of interaction, in being able to guide evolution.

8 Conclusions

The fascinating property of evolving systems is their ability to adapt and change to outside pressures. Any self-replicating process, which is open to mutation, will evolve. If the rules guiding the survival of these processes are manipulated, the results will change. Throughout the building of this system, the creatures have shown that they are able to adapt to, and make use of, errors and loopholes in the program code. It is this ability to survive and manipulate which is interesting.

The use of colour as a central property of recognition resulted in an emergent coherency of colour. Early experiments with 2D showed that that species could be distinguished easily and also made for good visual output. These ideas could be expanded further into full computer vision, where the texturing could also be used as a factor in object identification.

While the main purpose of this system was purely to produce animation, it is the processes, which define the animation, that hopefully make it worth watching. In general, the highly evolved behaviours provide much more interesting animation.

Theoretically is would be possible to breed creatures which exhibit a desired behaviour by tailoring the environment to suit. This way, an environment could be designed to create flocking systems or object avoidance. As shown by these early results, the solutions discovered would not always be obvious, and hopefully they would contain novel or interesting behaviours.

References

[1] Alcock, J.: Animal Behavior: an evolutionary approach, Fifth Edition. Sinauer. Sunderland MA. (1993)

[2] Dawkins, R.: The Blind Watchmaker. W.W. Norton, New York (1986)

[3] Gould, S. J.: Wonderful Life. Hutchinson Radius, London (1990)

[4] Langton, G.: [ed.] Artificial Life. The MIT Press, London. (1995)

[5] Ray, T. S.: In press. A computational approach to evolutionary biology. "Advanced Mathematical Approach to Biology", Takeyuki Hida, [ed.]. World Scientific Publishing Co. Pte. Ltd., Singapore. Also, ATR Technical Report TR-H-176.

[6] Sims, K.: Evolving Virtual Creatures, Computer Graphics (Siggraph '94) Annual Conference Proceedings, July, pp.15-22. New York: ACM Siggraph.

[7] Sims, K.: Artificial Evolution for Computer Graphics, Computer Graphics (Siggraph '91 proceedings), Vol.25, No.4, July, pp.319-328.

[8] Todd, S.,Latham, W.: Evolutionary Art and Computers, Academic Press. (1986)

[9] Yaeger, Larry. Computational genetics, physiology, metabolism, neural systems, learning, vision, and behavior or PolyWorld: life in a new context. Artificial Life III, [Ed.] Christopher G. Langton, SFI Studies in the Sciences of Complexity, Proc. Vol. XVII, Addison-Wesley. Pp. 263–298.

This article was processed using the LaTeX macro package with LLNCS style