Creatures: Artificial Life Autonomous Software Agents for Home Entertainment

Stephen Grand, Millennium Interactive Ltd, Quern House, Mill Court, Cambridge, U.K., stepheng@cyberlife.co.uk
Dave Cliff, Cognitive and Computing Sciences, University of Sussex, Brighton BN1 9QH, U.K., davec@cogs.susx.ac.uk
Anil Malhotra, Millennium Interactive Ltd, Quern House, Mill Court, Cambridge, U.K. anil@cyberlife.co.uk

Abstract

This paper gives a technical description of ***Creatures***, a commercial home-entertainment software package. ***Creatures*** provides a simulated environment in which exist a number of synthetic agents that a user can interact with in real-time. The agents (known as “creatures”) are intended as “virtual pets”. The internal architecture of the creatures is inspired by animal biology. Each creature has a neural network responsible for sensory-motor coordination and behavior selection, and an “artificial biochemistry” that models a simple energy metabolism along with a “hormonal” system that interacts with the neural network to model diffuse modulation of neuronal activity and staged ontogenetic development. A Hebbian learning mechanism allows the neural network to adapt during the lifetime of a creature.

Additionally, both the network architecture and details of the biochemistry for a creature are specified by a variable-length “genetic” encoding, allowing for evolutionary adaptation through sexual reproduction. ***Creatures***, available on Windows95 and Macintosh platforms from late 1996, offers users an opportunity to engage with Artificial Life technologies. In addition to describing technical details, this paper concludes with a discussion of the scientific implications of the system.

1. Introduction

Autonomous software agents have significant potential for application in the entertainment industry. In this paper, we discuss an interactive entertainment product based on techniques developed in Artificial Life and Adaptive Behavior research (e.g. Brooks and Maes 1994, Cliff et al 1994). The product, called ***Creatures***, allows human users to interact in real-time with synthetic agents which inhabit a closed environment. The agents, known as “creatures”, have artificial neural networks for sensory-motor control and learning, artificial biochemistries for energy metabolism and hormonal regulation of behavior, and both the network and the biochemistry are “genetically” specified to allow for the possibility of evolutionary adaptation through sexual reproduction.

Although a commercial product,[1] we believe aspects of ***Creatures*** will be of interest to the science and engineering communities. This paper discusses the most significant aspects of the product relevant to autonomous agent researchers. The product, available in North America and Europe from late 1996, runs in real-time on Windows95 (486/66Mhz) and Macintosh platforms.

Section 2 discusses related work. Section 3 presents a description of technical aspects of ***Creatures***, and Section 4 concludes with some speculative comments on the possible scientific impact of the product.

2. Related Work

2.1 Autonomous Agents for Entertainment

Here we briefly summarize work in Artificial Life and Adaptive Behavior research that is relevant to ***Creatures***. For background discussion and description of a selection of other entertainment-oriented research projects, see (Maes 1995).

Seminal work by Reynolds (1987) established the possibility of using autonomous agents for behavioral animation, a technique which allows movie sequences showing behavior in synthetic agents to be produced with the human animator giving only broad “choreographic” commands, rather than detailed frame-by-frame pose specifications. Subsequent related projects, such as that by Terzopolous et al (1994), where faithful kinematic simulations of fish are modeled with impressive visual accuracy and considerable biological plausibility in the behavioral control, have shared with Reynolds' original work a reliance on skillful manual design of the agent's physical morphology, behavioral control mechanism, or both. This can often require a significant investment of skilled labor.

Faced with the difficult task of designing lifelike synthetic agents for entertainment applications, several researchers have drawn inspiration from biology. For example, Blumberg (1994) has developed a behavioral control mechanism inspired by findings in ethology (the science of animal behavior) which is used to control a synthetic dog that inhabits a 3D software environment, interacting with a human user and with other virtual agents and objects in the environment.

Other researchers have worked on developing techniques that reduce the reliance on skilled labor by incorporating some type of automatic adaptation or learning mechanism in the agent software. Reynolds (1994) has explored the use of genetic programming (Koza 1992), a technique related to genetic algorithms (Goldberg 1986), to develop control programs for synthetic agents moving in 2D worlds with simplified kinematics. Sims (1994) has employed similar artificial evolution techniques to develop both the physical morphology and the artificial neural network controllers

[1] ***Creatures*** was developed by Millennium Interactive Ltd and is published by Warner Interactive and Inscape. The core Artificial Life techniques developed by Millennium for use in ***Creatures*** are referred to as CyberLife®. The CyberLife Web site is http://www.cyberlife.co.uk/

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

for synthetic autonomous agents that inhabit a 3D world with realistic kinematics. There is also a large body of work on learning in artificial neural networks (see e.g. Rumelhart and McClelland 1986).

2.2 Other Relevant Entertainment Products

Publications in the scientific literature describing commercial interactive entertainment software products are rare, so the citations in this section are to promotional material available on the world-wide-web.

One of the first pieces of entertainment software explicitly promoted as drawing on Artificial Life research was ***SimLife*** by Maxis, released in 1993 (Maxis, 1995). In essence, SimLife allowed a user to observe and interact with a "simulated ecosystem" with a variable terrain and climate, and a variety of species of plant life, herbivorous animals, and carnivorous animals. The ecosystem was simulated using cellular automata techniques, and so makes little use of autonomous agent techniques.

More recent products have had stronger links to autonomous agent research. Another Maxis product, ***El-Fish***, was presented as an "electronic aquarium" where users could design and breed virtual tropical fish which could then be observed swimming in a virtual fishtank. The similarities between this product and the work of Terzopolos et al (1994) are manifest.

It should also be noted that Maxis pioneered the concept of "software toys" as opposed to "computer games". The metaphor of "toy" rather than "game" is intended to highlight a different style of interaction: a game is usually played in one (extended) session, until an "end condition" is reached, and a score or high-score is awarded; in contrast, use of a toy does not imply a score or an aim to achieve some end-condition, and interaction with a toy is a more creative, ongoing, open-ended experience.

Subsequently, PFMagic Inc. (PFMagic 1996) has released two products, ***Dogz*** and ***Catz*** which give users on-screen animations of virtual pets based on dogs and cats. Users can interact with their virtual pets and train them to perform simple tricks. There is some similarity between these products and Blumberg's work mentioned above. Two products announced but not released at the time of writing are ***Fin-Fin*** by Fujitsu Interactive Inc. (Fujitsu 1996) and ***Galapagos*** by Anark (Anark 1996). Of the two, ***Galapagos*** has stronger links to Artificial Life, involving a 3D kinematically realistic model of a six-legged agent in a 3D maze-like environment with an adaptive neural-network-like controller based on Anark's proprietary "NERM" (Non-stationary Entropic Reduction Mapping) technology. The ***Fin-Fin*** product involves 3D rendering of a hybrid dolphin/bird creature which the user can engage in simple interactions with via a specialized input device combining a proximity detector with a microphone which detects amplitude and pitch of nearby sounds (e.g. voices).

All of ***Dogz***, ***Catz***, ***Fin-Fin*** and ***Galapagos*** are presented as involving Artificial Life technologies, but none of them (yet) employ genetically encoded neural network architectures or artificial biochemistries as used in ***Creatures***, which we describe in Section 3. Nor do they allow for the development of "culture" in communities of artificial agents, a possibility with ***Creatures*** which we discuss in Section 4.

3. Creatures

The creatures inhabit a "two-and-a-half dimensional" world: effectively a 2D platform environment with multi-plane depth cueing so that objects can appear, relative to the user, to be in front of or behind each other. On a typical Windows95 system, the world measures approximately 15 screens horizontally by 4 screens vertically, with the window scrolling smoothly to follow a selected creature. Within the world there are a number of objects which the creature can interact with in a variety of ways. The system has been written using object-oriented programming techniques: virtual objects in the world such as toys, food, etc. have scripts attached that determine how they interact with other objects, including the creature agents and the static parts of the environment. Some objects are "automated", such as elevators which rise/fall when a button is pressed. Other objects and environments may be added later. A screenshot showing a view of part of the world is shown in Figure 1.

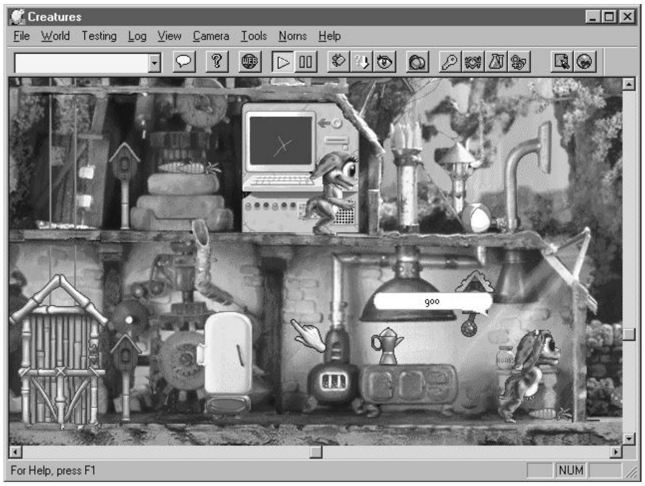

Figure 1: Screenshot showing a view onto part of the *Creatures* world

When the user's mouse pointer is anywhere within the environment window, the pointer changes to an image of a human hand. The user can move objects in the environment by picking them up and dropping them, and can attract the attention of a creature by waving the hand in front of it, or by stroking it (which generates a positive, "reward" reinforcement signal) or slapping it (to generate a negative, "punishment" reinforcement signal).

A typical creature is shown in Figure 2. All creatures are bipedal, but minor morphological details such as coloring and hair type are genetically specified. As they grow older, the on-screen size of the creature increases, up until "maturity", approximately one third of the way

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

through their life. The life-span of each creature is genetically influenced: if a creature manages to survive to old age (measured in game-hours) then senescence genes may become active, killing the creature. The creature has simulated senses of sight, sound, and touch. All are modeled using semi-symbolic approximation techniques. For example, the simulation of vision does not involve a simulation of optics or processing of retinal images. Rather, if a certain object is within the line of sight of a creature, a neuron representing the presence of that object in the visual field becomes active. Such approximations to the end-result of sensory processing are fairly common in neural network research. Sounds attenuate over distance and are muffled by any objects between the creature and the sound-source. An object can only be seen if the creature's eyes are pointing in its direction. There is also a simple focus-of-attention mechanism, described further below.

Figure 2: Close-up of two creatures

Creatures can learn a simple verb-object language, either via keyboard input from the user, or by playing on a teaching-machine in the environment, or from interactions with other creatures in the environment. On typical target platforms, up to ten creatures can be active at one time before serious degradation of response-time occurs. The following sections describe in more detail the neural network, biochemistry, and genetics for the creatures.

3.1 Neural Network

Each creature’s brain is a heterogeneous neural network, sub-divided into objects called ‘lobes’, which define the electrical, chemical and morphological characteristics of a group of cells. Cells in each lobe form connections to one or more of the cells in up to two other source lobes to perform the various functions and sub-functions of the net. The network architecture was designed to be biologically plausible, and computable from the ‘bottom-up’, with very few top-down constructs.

The initial model contains approximately 1,000 neurons, grouped into 9 lobes, and interconnected through roughly 5,000 synapses. However, all these parameters are genetically controlled and may vary during later phylogenesis.

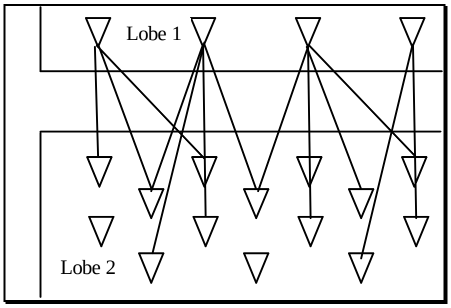

Figure 3: Sample interconnections between lobes

3.1.1 Structure

The structure of the neural architecture was designed to comply with several criteria:

- it must be very efficient to compute (a world with ten creatures requires the processing of some 20,000 neurons and 100,000 synaptic connections every second, in addition to the load imposed by the display and the rest of the system).
- it must be capable of supporting the planned brain model, i.e. the neural configuration which controls the first generation of creatures.
- it must be capable of expressing many other possible neural models, besides the planned one.
- it must not be too brittle—mutation and recombination should have a fair chance of constructing new systems of equal or higher utility than those of the parents.

Neurons. All the neurons within a single lobe share the same characteristics, but these characteristics can vary over a wide range of possible behaviors. Some aspects of the neurons’ dynamics are simple parameters, while others are defined as expressions. All of these factors are controlled genetically during the construction of a lobe. The parameters of a neuron are as follows:-

Table 1 Neuron Parameters

Type 1/2 inputs	Each cell may possess 0, 1 or 2 classes of input dendrites, each feeding signals from a different source lobe
State	Internal state, computed from genetically defined expression
Threshold	Output = (State>Threshold) ? State : 0
Relaxation Rate	Exponential recovery rate from current State towards Rest State
Rest State	Natural State value when unperturbed
Input gain	modulates inputs
State Function() (SVRule)	Expression to compute new State from input signals

A neuron’s internal state is computed via a genetically defined function known as a State-Variable Rule, or SVRule. SVRules are composed of interpreted opcodes and operands, and are also used to control several aspects of synaptic behavior. An SVRule expression is designed to be interpreted extremely rapidly, and also to be non-brittle and fail-safe—genetic mutations can never cause syntax

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

errors. SVRules can compute new state values in many ways (see Table 2). Many of these possible functions go well beyond the present needs of the 'brain model', but are provided in order that a powerful tool-kit is available for future man-made or evolutionary improvements to the system.

Table 2 SVRule examples

state PLUS type0 Sum of inputs is added to previous state
state PLUS type0 MINUS type1 Type0 inputs are excitatory and type1 are inhibitory
anded0 State is sum of type0 inputs or zero if not all inputs are firing. Previous state is ignored
state PLUS type0 TIMES chem2 State is raised by current input modulated by chemo-receptor

After a neuron's State is computed, a 'relaxation' function is applied to it, which exponentially returns it towards a definable 'rest state'. One important use of this relaxation function is to act as a damping mechanism, since the further the neuron's state gets from equilibrium, the faster it relaxes, and so the harder it becomes to disturb it further. This tension between input and relaxation not only keeps the system reasonably stable, but can also provide an integration of input signals, such that the state of the neuron reflects both the intensity and the frequency of the stimuli.

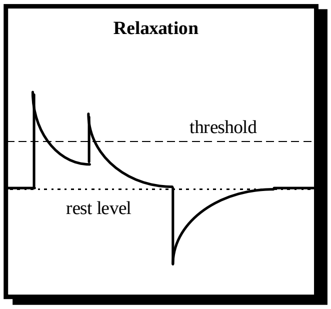

Figure 4: State Relaxation

Dendrites. Each neuron is fed by signals from one or more dendrites. Each cell may carry one or two different classes of dendrite, each with different characteristics and source lobes, thus allowing for the integration of different types of data. The main parameters for a dendrite/synapse are as follows:-

Table 3 Dendrite Parameters

STW	Short-term weight, used to modulate input signals
LTW	Long-term weight. Acts as rest state for STW and provides statistical response to reinforcement
STW relaxation rate	Rate at which STW relaxes back towards LTW
LTW relaxation rate	Rate at which LTW rises towards STW
Susceptibility	Current susceptibility to reinforcement
Susceptibility relaxation rate	Half-life of Susceptibility parameter
Strength	Controls dendrite migration
Reinforcement SVRule	Expression to compute changes in STW
Susceptibility SVRule	Expression to compute changes in sensitivity to reinforcement
Strength gain SVRule	Expression to compute Strength increase
Strength loss SVRule	Expression to compute atrophy

The signal arriving at the synapse is modulated by the Short-term Weight to provide an output value. A rise in STW can be triggered by a reinforcement SVRule usually in response to activity at a chemo-receptor. After disturbance, both the STW and the LTW relax exponentially towards other, with the LTW being the slower. The STW therefore reacts strongly to individual reinforcement episodes, while the LTW effectively computes a moving average of many STW disturbances: if a creature meets with situation X and finds that its chosen course of action was undesirable, then it should immediately be strongly disinclined to repeat the action, especially as many of the incentives to do so may still be present. However, situation X may not always be as bad as first experience suggests, and so the creature's long-term interpretation should be less sweeping.

Dendritic Migration. The initial wiring is defined at birth according to a small number of genetic rules. Generally, neurons attempt to connect from one lobe to another in a direct spatial mapping, with multiple dendrites fanning out in a specified distribution to either side of the optimum source cell (see Figure 1). After birth, however, individual dendrites may migrate and form new connections (always within the same source lobe). Periodically, a Strength value change is computed for each synapse using SVRules, often in response to chemical changes. If the Strength falls to zero, the dendrite disconnects and follows the appropriate rule about how to find a new connection. These migration rules were chosen in order to fulfill the requirements for the initial brain model. It was hoped that a more general scheme could be invented, but this was not possible in the time available. An extra migration function, involving a survival-of-the-fittest competition between cells for the right to represent a particular input pattern, was implemented as part of the model's generalization system, but has caused problems and so is currently left disconnected.

3.1.2 Brain Model

The above architecture is a generalized engine for neuron-like computation, whose circuitry can be defined genetically. This section describes the specific model which has been superimposed onto the system to implement the first generation of creatures.

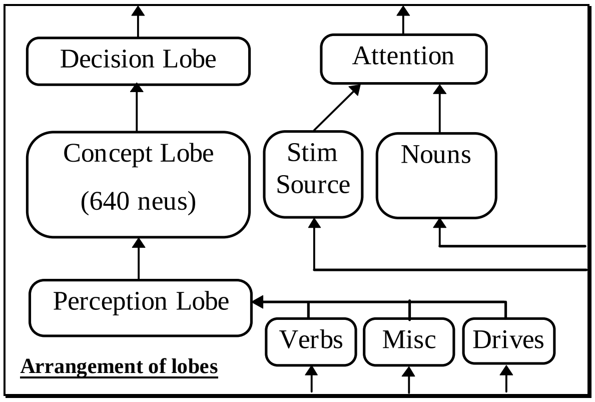

Figure 5: Brain Model

Attention. Some of the neural circuits are devoted to relatively minor tasks. For example, two lobes are used to implement an attention-directing mechanism. Stimuli arriving from objects in the environment cause a particular cell to fire in an input lobe (where each cell represents a different class of object). These signals are mapped one-on-one into an output lobe, which sums the intensity and frequency of those stimuli over time. Simulated lateral inhibition allows these cells to compete for control of the creature's attention. The creature's gaze (and therefore much of his sensory apparatus) is fixed on this object, and it becomes the recipient for any actions the creature chooses to take. Such a mechanism limits creatures to "verb object", as opposed to "subject verb object" modes of thought, but serves to reduce sensory and neural processing to acceptable levels, since the net need only consider one object at a time.

Decision Making. The bulk of the remaining neurons and connections make up three lobes: a 'perception' lobe, which combines several groups of sensory inputs into one place; a large region known as Concept Space, in which event memories are laid down and evoked; and a small but massively dendritic lobe called the Decision Layer, where relationship memories are stored and action decisions get taken. The overall model is behaviorist and based on reinforcement by drive reduction.

Cells in Concept Space are simple pattern-matchers. Each has one to four dendrites and computes its output by ANDing the analog signals on its inputs, which come via the Perception lobe from sensory systems. Each therefore fires when all of its inputs are firing. These cells are randomly wired at birth, but seek out new patterns as they occur. Once a cell has committed to a particular pattern, it remains connected until its dendrites' strengths all fall to zero. A biochemical feedback loop and two SVRules attempt to maintain a pool of uncommitted neurons while leaving 'useful' (i.e. repeatedly reinforced) cells connected for long periods. The Perception lobe has around 128 sensory inputs, and so the total number of cells that would be required to represent all possible sensory permutations of up to four inputs is unfeasibly large. This reinforcement, atrophy and migration mechanism is designed to get round this problem by recording only the portion of input space which turns out to be relevant. There are a number of problems associated with this approach, but on the whole it works.

The Decision layer comprises only 16 cells, each representing a single possible action, such as "activate it", "deactivate it", "walk west", and so on, where "it" is the currently attended-to object. The Decision neurons are highly dendritic and feed from Concept Space. The dendrites' job is to form relationships between Concept cells and actions, and to record in their synaptic weightings how appropriate each action is in any given sensory circumstance.

An SVRule on each dendrite decides the current synaptic 'susceptibility', i.e. sensitivity to modulation by reinforcers. This is raised whenever that dendrite is conducting a signal to a cell and that cell is firing (i.e. the connection represents both a 'true' condition and also the current action). It then decays exponentially over time. Synapses are therefore sensitized when they represent relationships between current sensory schemata and the latest action decision, and remain sensitive for a period in order to respond to any share of a more-or-less deferred reward or punishment.

There are not enough dendrites to connect every action to every Concept cell, and so these dendrites are also capable of migrating in search of new sources of signal. Again a biochemical feedback loop controls atrophy, while repeated reinforcement raises strength.

Decision cells sum their inputs into their current state (in fact they sum their type 0 inputs (excitory) and subtract the sum of their type 1 (inhibitory) inputs). The relaxation rate of Decision cells is moderate, and so each cell accumulates a number of nudges over a short period, based on the number of Concept cells which are firing, plus their intensity. The strongest-firing Decision cell is taken to be the best course of action, and whenever the winner changes, the creature invokes the appropriate action script.

Generalization. Because Concept Space seeks to represent all the various permutations of one to four inputs that exist within the total sensory situation obtaining at a given moment, the system is capable of generalizing from previously learned relationships to novel situations. Two sensory situations can be deemed related if they share one or more individual sensory features, for example situation ABCD, which may never before have been experienced, may evoke memories of related situations such as D, ABD, etc. (although not BCDE). Each of these sub-situations represents previously learned experience from one or more related situations and so each can offer useful advice on how to react to the new situation. For example, "I find myself looking at a big, green thing with staring eyes, which I've never seen before. I remember that going up to things with staring eyes and kissing them is not a good idea, and that hitting big things, particularly big, green things, doesn't work well either. So, all in all, I think I'll try something else this time." Of course, if the new situation turns out to have different qualities from previously experienced sub-situations (an 'exception to the

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

rule'), then both the new, total 'concept' and the previously learned sub-concepts will be reinforced accordingly. As long as super-concepts fire more strongly than sub-concepts, and as long as reinforcement is supplied in proportion to cell output, the creature can gradually learn to discriminate between these acquired memories and so form ever more useful generalizations for the future.

Learning. Delayed-reinforcement learning is provided by changes to Decision Layer short-term weights in response to the existence of either a Reward chemical (for excitatory synapses) or a Punishment chemical (for inhibitory ones). These chemicals are not generated directly by environmental stimuli but during chemical reactions involved in Drive level changes. Each creature maintains a set of chemicals representing 'drives', such as "the drive to avoid pain", "the drive to reduce hunger", and so on. The higher the concentration of each chemical, the more pressing that drive. Environmental stimuli cause the production of one or more drive raisers or drive reducers: chemicals which react to increase or decrease the levels of drives. For example, if the creature takes a shower by activating a shower object, the shower might respond by reducing hotness and coldness (normalizing temperature), decreasing tiredness and increasing sleepiness. Drive raisers and reducers produce Punishment and Reward chemicals respectively through the reactions:

DriveRaiser ⟶ Drive + Punishment
DriveReducer + Drive ⟶ Reward

Drive reduction therefore increases the weights of excitatory synapses while drive increase reinforces inhibitory ones. Of course, reducing a non-present drive has no effect, and so the balance of punishment to reward may reverse. Thus, many actions on objects can return a net punishment or a net reward, according to the creature's internal state at the time. Creatures therefore learn to eat when hungry but not when full.

The brain model is not an ambitious one, and severely limits the range of cognitive functions which can arise. It is also primitively Behaviorist in its reinforcement mechanism. However, it serves its purpose by providing a learned logic for how a creature chooses its actions, and doesn't suffer from too many non-life-like side effects: its in-built generalization mechanism reduces arbitrariness in the face of novelty, and the dynamical structure, albeit damped and close to equilibrium, produces a satisfactorily complex and believable sequence of behaviors, surprisingly free from limit cycles or irretrievable collapse into an attractor.

3.2 Biochemistry

Central to the function of the neural net is the use of a simplified, simulated biochemistry to control widespread information flow, such as internal feedback loops and the external drive-control system. This mechanism is also used to simulate other endocrine functions outside the brain, plus a basic metabolism and a very simple immune system. The biochemistry is very straightforward and is based on four classes of object:

Chemicals. These are just arbitrary numbers in the range 0 to 255, each representing a different chemical and each associated with a number representing its current concentration. Chemicals have no inherent properties—the reactions which each can undergo are defined genetically, with no restrictions based on any in-built chemical or physical characteristics of the molecules themselves.

Emitters. These chemicals are produced by Chemo-emitter objects, which are genetically defined and can be attached to arbitrary byte values within other system objects, such as neurons in the brain or the outputs of sensory systems. The locus of attachment is defined by a descriptor at the start of an emitter gene, representing 'organ', 'tissue' and 'site', followed by codes for the chemical to be emitted and the gain and other characteristics of the emitter. Changes in the value of a byte to which an emitter is attached will automatically cause the emitter to adjust its output, without the code which has caused the change needing to be aware of the emitter's existence.

Reactions. Chemicals undergo transformations as defined by Reaction objects, which specify a reaction in the form $_iA + [_jB] \rightarrow [_kC] + [_lD]$, where $_{i,j,k}$ determine ratios. Most transformations are allowed, except for 'nothing → something', for example:

A + B → C + D	normal reaction with two products
A + B → C	'fusion'
A → nothing	exponential decay
A + B → A + C	catalysis (A is unchanged)
A + B → A	catalytic breakdown (of B)

Reactions are not defined by immutable chemical laws but by genes, which specify the reactants and reaction products and their proportions, along with a value for the reaction rate, which is concentration-dependent and therefore exponential over time.

Receptors. Chemical concentrations are monitored by Chemo-receptor objects, which attach to and set arbitrary bytes defined by locus IDs, as for emitters. Receptor genes specify the locus, the chemical that the receptor responds to, the gain, the threshold and the nominal output. Many parts of the brain and body can have receptors attached, and thus can become responsive to chemical changes.

Biochemical structures. Attaching receptors and emitters to various loci within brain lobes allows widespread feedback paths within the brain, particularly in combination with reactions. Paths have been implemented to control synaptic atrophy and migration, and also to provide drive-reduction and learning reinforcement. Other neurochemical interactions are possible, such as the control of arousal. However, these have not been implemented, and we wait to see whether Nature can discover them for us.

As well as controlling vital neural systems, biochemistry is used to implement those systems which are

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

not actually necessary or compulsory within digital organisms, yet which would be expected by the general public. For example a simple metabolic system is simulated based on the following reactions:

starch ⇨ glucose ⇔ glycogen
↳CO_2 + H_2O + energy

Similarly, a selection of biochemicals and reactions produce the effects of toxins, which may be ingested from plants or emitted by the various synthetic ‘bacteria’ which inhabit the environment. These bacteria carry various ‘antigens’, which invoke ‘antibody’ production in the creatures, causing a very simplified immune response. The bacterial population is allowed to mutate and evolve, potentially adding a little co-evolutionary spice to the proceedings!

3.3 Genetics

As much as possible of the creature’s structure and function are determined by its genes. Primarily, this genome is provided to allow for inherited characteristics—our users expect their new-born creatures to show characteristics identifiably drawn from each parent. However, we have also gone to considerable trouble to ensure that genomes are capable of evolutionary development, including the introduction of novel structures brought about by duplicated and mutated genes.

The genome is a string of bytes, divided into isolated genes by means of ‘punctuation marks’. Genes of particular types are of characteristic lengths and contain bytes which are interpreted in specific ways, although any byte in the genome (other than gene markers) may safely mutate into any 8-bit value, without fear of crashing the system.

The genome forms a single, haploid chromosome. During reproduction, parental genes are crossed and spliced at gene boundaries. Occasional crossover errors can introduce gene omissions and duplications. A small number of random mutations to gene bodies is also applied. To prevent an excessive failure rate due to reproduction errors in critical genes, each gene is preceded by a header which specifies which operations (omission, duplication and mutation) may be performed on it. Crossing-over is performed in such a way that gene linkage is proportional to separation distance, allowing for linked characteristics such as might be expected (for example, temperament with facial type). Because the genome is haploid, we have to prevent useful sex-linked characteristics from being eradicated simply because they were inherited by a creature of the opposite sex. Therefore, each gene carries the genetic instructions for both sexes, but only the un-sexed and appropriately sexed genes get expressed in the phenotype.

Each gene’s header also contains a value determining its switch-on time. The genome is re-scanned at intervals, and new genes can be expressed to cater for changes in a creature’s structure, appearance and behavior, for example during puberty.

Some of our genes simply code for outward characteristics, in the way we speak of the “gene for red hair” in humans. However, the vast majority code for *structure*, not function. We could not emulate the fact that real genes code only for proteins, which produce structures, which in turn produce characteristics. However, we have tried to stay as true as we can to the principle that genotype and phenotype are separated by several orders of abstraction. Genes in our creatures’ genomes therefore code for structures such as chemo-receptors, reactions and brain lobes, rather than outward phenomena such as disease-resistance, fearlessness or strength.

4. Discussion and Conclusions

It is difficult to provide any “results” in this paper, since the project was essentially an exercise in engineering, rather than science. The overall objective was to create synthetic, biological agents, whose behavior was sufficiently life-like to satisfy the expectations of the general public. Sales figures will be our results, and at the time of writing, the product is still to be launched. However, in subjective terms, we have achieved most of our aims: the behavior of the creatures is dynamically “interesting” and varied and they do indeed appear to learn. Occasional examples of apparently emergent “social” behavior have been observed, such as cooperation in playing with a ball, or “chase” scenes resulting from “unrequited love”. However, it is very difficult to establish how much of this is genuine and how much is conferred by an observer’s tendency to anthropomorphism. The dynamical behavior of the agents and overall environment has been gratifyingly stable, and configuring a usable genotype has not been a problem, despite requiring approximately 320 interacting genes, each with several parameters. From that point of view, our belief that such a complex synthesis of sub-systems was an achievable aim appears to have been justified.

We believe that ***Creatures*** is probably the only commercial product available that allows home users to interact with artificial autonomous agents whose behavior is controlled by genetically-specified neural networks interacting with a genetically-specified biochemical system. As the creatures are responsible for coordinating perception and action for extended periods of time, and for maintaining sufficient internal energy to survive and mature to the point where they are capable of sexual reproduction, it could plausibly be argued that they are instances of “strong” artificial life, i.e. that they exhibit the necessary and sufficient conditions to be described as an instance of life. Naturally, formulating such a list of conditions raises a number of philosophical difficulties, and we do not claim here that the creatures *are* alive. Rather, we note that the philosophical debate concerning the possibility of, and requirements for, strong artificial life, will be raised in the minds of many of the users of ***Creatures***. As such, the “general public” will be engaging with artificial life technologies in a more complete manner when using

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra

Creatures than when using the other products mentioned in Section 2.2. Furthermore, if we assume that each user runs 5 to 10 creatures at a time, then after a few months of reasonable sales around the world, it is possible that there will be millions, or even tens of millions, of creatures existing in the “cyberspace” provided by the machines of the global ***Creatures*** user community. In this sense, the user community will be helping to create a “digital biodiversity reserve” similar to that advocated by T.Ray in his ongoing work on NetTierra, a major global Artificial Life research experiment (Ray 1994,1996). If we choose to, we can monitor the evolution of particular features in groups of creatures: on a local scale there may be little variation, but national or global comparisons may reveal divergent evolutionary paths. Also, because the creatures can learn within their lifetimes, both from humans and from other creatures, it should be possible to study the spread of “culture” or the emergence of “dialects” as creatures, moved from machine to machine via electronic mail, teach each other behaviors or language variants. In this sense, ***Creatures*** users could be considered to be taking part in an international Artificial-Life science experiment. Hopefully, they will also be having fun.

Acknowledgements

Thanks to all at Millennium Interactive Ltd. for their involvement and permission to publish this work.

References

(Anark 1996) Website at http://www.anark.com/index.html

(Blumberg 1994) B. Blumberg "Action Selection in Hamsterdam: Lessons from Ethology" in (Cliff et al 1994) pp. 108-117.

(Brooks and Maes 1994) R. Brooks and P. Maes "ALifeIV: Proceedings of the Artificial Life IV Workshop". MIT Press.

(Cliff et al 1994) D. Cliff, P. Husbands, J.-A. Meyer and S.W. Wilson. "From Animals to Animats 3: Proceedings of the 3rd International Conference on the Simulation of Adaptive Behavior (SAB94)". MIT Press.

(Fujitsu 1996) Website at fttp://www.fujitsu.co.jp/hypertext/news/1996/May/14-e.html

(Goldberg 1989) D.E. Goldberg "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison Wesley.

(Koza 1992) J. R. Koza "Genetic Programming: On the programming of computers by means of natural selection" MIT Press.

(Maes 1995) P. Maes "Artificial Life Meets Entertainment: Lifelike Autonomous Agents" Communications of the ACM 38(11):108-114,

(Maxis 1996) Website at http://www.maxis.com/

(PFMagic 1996) Website at http://www.pfmagic.com/

(Ray 1996) T.S.Ray "Continuing Report on the Network Tierra Experiment" unpublished document available from http://www.hip.atr.co.jp/~ray/tierra/netreport/netreport.html

(Ray 1994) T.S.Ray "A Proposal To Create Two BioDiversity Reserves: One Digital, and One Organic" unpublished document available from http://www.hip.atr.co.jp/~ray/pubs/reserves/reserves.html

(Reynolds 1987) C. Reynolds "Flocks, herds and schools: A distributed behavioral model". Computer Graphics 21(4):25-34.

(Reynolds 1994) C. Reynolds "Evolution of Corridor Following in a Noisy World" in (Cliff et al 1994).

(Rumelhart and McClelland 1986) D.E. Rumelhart and J.L. McClelland "Parallel Distributed Processing, Volume 1: Foundations" MIT Press.

(Sims 1994) K. Sims "Evolving 3D Morphology and Behavior by Competition", in (Brooks and Maes 1994) pp.28-39.

(Terzopoulos et al 1994) D.Terzopoulos et al. Artificial fishes with autonomous locomotion, perception, behavior and learning, in a physical world. In (Brooks and Maes 1994) pp.17-27.

Copyright © 1996, S.Grand, D.Cliff, A.Malhotra