School of Computer Science & Software Engineering Bachelor of Software Engineering Honours Research Proposal, Semester 1, 2003 Evolving Non-Standard Models of Computation Supervisor: A. Prof David Dowe

Stephen Jackel,
Student ID: 12782548,
Monash University, Clayton, Australia
Email: stjac1@student.monoash.edu.au
Computer Science & Software Engineering
http://www.csse.monash.edu.au/

April 30, 2003

1 Introduction

1.1 Is there life on other planets?

At present we are aware only of life on planet Earth. Many people argue that Earth-like planets may be extremely rare. Thus, assuming that life requires Earth-like planets, life on Earth may well be unique. Others propose that in a large enough universe, possibly with many Earth-like planets, that life elsewhere is inevitable. But even then, what form will this life take? Will other life display signs of intelligence? We do not know, and given the massive distances to even our closest neighbour stars, we may never know.

Life forms as we know them are based in the medium of carbon chemistry. Assuming that the Darwinian theory that life on Earth evolved through the process of natural selection is correct, we imagine that life on other planets would be similar. But we have only see an instance of life on Earth; life may take many forms. We should be able to apply the principle of evolution to create life in other media, life that may be very different to that which we are used to. This project looks at evolving and "teaching" digital organisms. It is hoped that insights gained from the field of artificial life will help us understand life on our planet and perhaps be a guide to the possibility of life elsewhere in our universe.

1.2 Search For Non-Standard Models of Computation

Hand-held calculators are able to perform mathematical computations. Humans, too, are able to perform many different computations, some of which are the same as a hand-held calculator. However, the method that humans use, especially when done without the use of paper and pens or other tools, is somewhat haphazard and the same calculation may even produce different results on different occasions. Clearly different models of computation are being used. While some methods are much faster and more efficient than others, some are also more powerful and are able to compute a larger range of problems. Models can be tested for Turing Completeness; if a model is Turing Complete then it is able to compute any computable function. It is possible that other models of computation are much more efficient or even more powerful than those currently [1]. Research into quantum computing and biomolecular computing is attempting to find new models of computation [2][3][4]. This project looks to digital evolution as a source of new models and questions whether we may find practical applications for evolving digital life.

2 Research Context

2.1 Definition of life?

What is life? Still noone is able to give a complete answer to this age old question. The question arose again in a slightly different form in the early 1950's as the rise of digital computing prompted the question, "What is intelligence?" Could computers be capable of displaying intelligence? Could we create "artificial" intelligence? Would that intelligence be considered alive? In 1950, the Imitation Game, later known as the Turing Test was suggested by Alan Turing as a way of defining intelligence [5]. Intelligent behaviour was defined as being the ability to act in a (restricted) manner such that an observer could not tell if the entity was human or not. Since then, the Turing Test has been used as a guide to both life and intelligence in the field of computer science [6]. But life and intelligence, while perhaps related, are very different. It is easy to imagine or find living organisms that exhibit intelligence but would fail the Turing Test. Even humans who speak another language may have great difficulty. Many believe that the Turing Test is more useful as a benchmark and definition only of human intelligence [1]. Therefore, we must find other means of defining and testing for life and intelligence when searching in the digial realm.

Biology and an attempt to understand life on our planet helped to spawn the field of artificial life where computer programs attempt to simulate evolution and/or grow artificial life with the general goal of achieving life of comparable complexity to our own. The digital medium as implemented on computers is an attractive place to simulate life as countless variables and values of experiments can be measured and analysed, and experiments can be repeated exactly, unlike in biology where slight variations in conditions or hidden variables may cause different results. Vast improvements in computational power and storage space have enabled researchers to watch evolution in "fast-forward".

While many simulations have been run to gain insights into evolution in our world, the results are only useful as far as the model that implements them will stretch; the more detailed and realistic the model, the more accurate the results. But forcing the digital medium to imitate our carbon based world of chemistry and physics becomes increasingly difficult as more detail is added. Although much insight may be gained through such experimentation, organisms that evolve on such systems will never be the same as in the real world because they are not in the real world; they are influenced by the medium in which they live.

In fact, the digital medium is almost completely independent of the carbon based one in which we live. It has its own rules, its own chemistry and physics. Organisms that evolve in the digital medium will adapt to and belong in the digital medium. This new environment is likely to produce very different life and intelligence to that with which we are familiar. Thomas Ray believes that we will never be able to guide evolution into producing a particular form due to lack of foresight - the “algae to corn” problem [7]. For example, having never seen anything more complex than single celled algae, it would be virtually impossible to have the insight to devise a fitness function that would guide the organisms through many levels of complexity to produce corn. Transferring this across to the digital medium, we have no idea where digital intelligence may go or what forms it will take and thus we should not try to impose too many fitness functions or other restrictions, but rather let evolution find its own path.

2.2 Using Digital Evolution to Solve Problems

Biologists have been attempting to recreate biological life in digital form for a number of years. Nevertheless, there have been other attempts to harness the power of evolution. Genetic algorithms (GA) have been used to find optimal solutions to problems [4]. However, the form of the solution must already be known; GA’s merely attempt to find the best values of the solution. This method is thus not useful in discovering new forms of intelligence or computation. An improvement comes in the form of Genetic Programming (GP) where the form of the solution is subject to change according to a pre-defined fitness function. Genetic programming appears to be more likely to find new forms of intelligence but, as predicted by Thomas Ray, it is limited to the fitness function. A fitness function may provide a strong guiding force for evolution to begin with. However, unless it is extremely well thought-out, it will become weaker as the organisms grow towards it until the fitness function becomes a ceiling beyond which evolution cannot pass.

Fitness functions that change as organisms evolve have been used to counter this problem, the most successful being coevolution [4]. Coevolution involves the fitness function also being subject to evolutionary processes rather than being fixed or embedded in the medium. The organisms grow towards satisfying the fitness function while the fitness function tries to maximise the gap between itself and the level of the evolving organisms. Effectively, instead of the fitness function being a fixed bar, the bar is raised as the organisms evolve closer. However, this is only a successful method if the fitness function stays ahead (but not by too much) of the evolution of the organisms. Finally, at the far extreme is Thomas Ray’s idea of letting the medium dictate the selective forces, and observing what happens. Thomas

Ray has attempted to implement this idea in Tierra [8][7].

2.3 Tierra

Tierra is an Artificial Life (AL) simulator written by Thomas Ray that attempts to use evolution in the "digital medium" to bring forth life (usually inoculated with a simple reproducing program) [8].Written in C, Tierra implements a virtual computer complete with "Darwinian" operating system and architecture that allows machine code to be evolved through randomly changing bits, "mutation", and interchanging sections of code, "recombination". The system supports evolving programs or "organisms" and allows for the inevitable lines and sections of code that are not executable by ignoring them. Over time, beneficial changes are made to the code, which enable the digital organisms living in the system to evolve and increase in function and complexity.

There are a number of control and monitoring tools built into the simulator; Tierra records all births and deaths and keeps a "genebank" of all successful genomes (sequences of creature code) and is able to monitor and record creature interactions. Tierra is an extremely useful tool both to observe, measure and experiment with evolution in the digital medium.

Organisms in the digital world of Tierra face the obstacles of various inherent natural limitations and finite resources. The RAM available on the computer running the simulation forms the space or "soup" that the organisms must occupy and share. Likewise, in the digital medium, the CPU is likened to energy as it is the only process that allows creatures to change state, reproduce and perform other actions and is therefore in high demand [7]. Competition for these resources creates evolutionary pressures for the organisms to adapt to their environment and outperform their competitors.

The instructional set used by the CPU forms the "physics and chemistry" of the digital medium together with the methods used to allocate resources. In Tierra, the instructional set is deliberately designed to occasionally introduce errors in calculations, feeding evolution. Organisms in memory may read or execute instructions from anywhere (or in some implementations, within a certain proximity of an organism) but may only write to memory that it "owns". These rules help to define the ways in which digital organisms may interact with others.

Thomas Ray has produced an array of different organisms in Tierra by inoculating the simulation with a simple creature that reproduced itself [8]. Eventually, mutations were not required as the creatures learnt to sexually reproduce, having offspring that were a mixture of the parents. Thomas Ray, pointing to the Amazon rain forests as another example, argues that as organisms increase in complexity, the interactions between the organisms often form a much stronger evolutionary force than the medium upon which they are implemented [7].

2.4 Extensions to Tierra

Tierra is being extended to "Network Tierra" through the use of a global "Digital Reserve" [9]. Organisms will be able to roam over a form of the Internet, migrating between different computers on the network in search of resources. People will be able to donate CPU time in a similar manner to SETI [10]. It is hoped that this expansion in the digital universe and introduction of further, complex interactions will set about an explosion of multi-cellular digital life that is equivalent to the explosion of multi-cellular life on earth thought to have occurred during the Cambrian period [11]. Tierra is amoung the most widely known Artificial Life simulators, though there are a number of others such as Avida. Avida is a Tierra derivative that allows the introduction of fitness functions in addition to the natural ones intrinsic in the medium and implementation [12]. All artificial life simulations may be viewed as tools that allow us to gain an insight into our own world, the life around us and make educated predictions about life elsewhere in the universe.

3 Research Plan and Methods

3.1 Proposed work

The project will modify Tierra or Avida to allow a special sort of fitness function to regulate the allocation of CPU time to the digital organisms. The goal will be to try to nurture intelligence so that the organisms are able to perform computations, quite possibly in a manner that we have not seen before. Thus the project attempts to find a balance between allowing the digital organisms enough freedom in their natural digital medium to evolve and display complex behaviour, and causing the organisms to display behaviour that is useful to us in the real world. That is, we want to be able to see and measure what computations the digital organisms are able to perform. We would like to be able to gain some insight into their "intelligence" by seeing how they react to new calculations that are similar to those that they can already solve. This project seeks to evolve digital organisms that are capable at least of the most basic mathematical computations.

3.2 Modifying the Simulator

The main source of energy for digital organisms is the CPU. Artificial life simulators usually allocate CPU time evenly among organisms. In this project, the method of allocating CPU time to the creatures will be changed through the use of a type of fitness function. Creatures will be penalised by losing CPU cycles for not satisfying the fitness function or "teaching function".

3.3 How the Teaching Function Works

Each organism normally gets 1/(number of organisms) of CPU time. Organisms will be penalised for wrong answers and will therefore get (1-(%wrong/100))/(number of organisms). This means that the entire cycle through all organisms will normally last less time but this way it is "fair" in terms of rewards - i.e. if all organisms get 50% wrong then although each time period that they get is less than normal, they all get the same amount less. Thus they get the same percentage of CPU time each, just as if it was 1/(number of organisms).

The fitness function or teaching function can be a computation/predicting test. This will involve creatures being given a (long) binary string and the organism being required to compute/predict the next few bits. For example, given 96 bits, the creature must predict the next 56. The exact length of the bit strings will be decided at a later stage. The teaching function will test each creature before allocating CPU time. The test may be chosen from a range of tests, including simple pattern matching, compliments, counting, addition, subtraction, multiplication and possibly division, sequences, e.g. random numbers, Fibonacci sequence, squares, powers and other mathematical functions. The more basic functions will be tested first - simple pattern recognition, addition, subtraction and multiplication. However, as organisms learn to count or add, the teaching function will move onto new problems, while regularly going back to test old skills. There will be a good variety of different functions of increasing difficulty such that the organism may eventually be required to remember previous series/answer.

3.4 How is this teaching function different to a normal fitness function?

The teaching function is a fairly regular fitness function and will initially be a "fixed bar" type at that. The reason why I have called it a teaching function rather than a fitness function is because I hope that this will prove to be a successful method of teaching the organisms. After inoculating soup with simple program that reproduces and copies last x bits to output, the teaching function will teach the digital creatures to count, add, multiply, etc. But once they learn these things, the teaching function no longer has an effect - all the organisms will be getting the same amount of CPU time. Thus the organisms will be again free to evolve, but this time with the extra knowledge/skills/intelligence. And they are unable to slide back down the evolutionary ladder without being penalised. This method of teaching may be the elusive "building block" technique that others have tried or wish to try to implement [13][14]. Rather than "cheating" or playing god in deciding which building blocks to keep, this teaching function will ensure that the organisms keep their skills intact while evolving upwards. The digital creatures may find some interesting uses for their new skills. Adding new teaching functions once the organisms have mastered the first one may be a way to loosely guide evolution or at least give it a helping hand in progressing. However, it is very difficult to test the intelligence of these organisms. The teaching function will serve a dual purpose as it will also test the abilities of the organisms.

3.5 Proposed Thesis Chapter Headings

1. Introduction
2. Creating Artificial Life in the Digital Medium
3. Expanding Tierra/Avida
4. Methods of Computation
5. Creating Artificial Intelligence
6. Conclusion and Future Directions
7. Bibliography
8. Appendix: Software Produced

3.6 Special Facilities Required

The facilities offered to Software Engineering Honours students should be sufficient provided that a computer may be left on for many hours without being reset.

Date	Uni Week	Activity/Task
10th March	Sem 1, w2	Background Reading
25th April	Easter hols	Draft of Research Proposal
30th April	Sem 1, w8	Research Proposal
5th May	Sem 1, w 9	Begin reading literature
15th May	Sem 1, w 10	Draft of Research Methodology Assignment
22nd May	Sem 1, w 11	1st Draft of Literature Review
28th May	Sem 1, w 12	Research Methodology Assignment
13th June	Mid Sem Exams 1	2nd Draft of Literature Review
16th June	Mid Sem Exams 2	Begin Modifying Simulator
25th June	Mid Sem Exams 3	Literature Review
7th July	Sem Break w1	Run modified simulator
14th July	Sem Break w2	Finish modifying simulator
15th July	Sem Break w2	Begin running simulations
6th August	Sem 2, w3	Begin writing up findings
5th September	Sem 2, w7	Submit Dynamical Hierarchies Paper
10th September	Sem 2, w8	1st Draft of Thesis Due
1st October	Med Sem Break	Website updated
22nd October	Sem 2, w12	Thesis concluded, including software
22nd October	Sem 2, w13	Research Log Book, finish web site
25th September	Exams w1	Final talk preparation

Table 1: Proposed Project Timeline

3.7 Timetable

4 Relevance of the Research

This project will be beneficial in a number of different areas:

1. The quest to understand life on Earth and how intelligence might evolve.

2. Useful in the search for finding non-standard models of computation. Is it possible to guide evolution into performing computations? This project will attempt to discover the potential of finding new models of computation through evolution in a digital world.

3. Using a teaching function as a method of finding the building blocks to move up the evolutionary chain towards a Digital Cambrian explosion of complexity. Will the digital organisms be able to use their new knowledge for new purposes? Is this a way of evolving intelligence?

4. Are such teaching methods are a useful way of guiding evolving organisms towards an end goal or at least speeding up the evolutionary process.

References

[1] C. S. Calude and M. J. Dinneen. Breaking the turing barrier. Technical report, 1998.

[2] Rajarshi Das. Emergent computation in cellular automata, 1998.

[3] Melanie Mitchell, James P. Crutchfield, and Peter T. Hraber. Dynamics, computation, and the "edge of chaos": A re-examination.

[4] Melanie Mitchell and Charles E. Taylor. Evolutionary computation: An overview. *Annual Review of Ecology and Systematics*, 20:593–616, 1999.

[5] A. M. Turing. Computing machinery and intelligence. *Mind, 59 pages 433-460*, 1950.

[6] Graham Oppy and David Dowe. The turing test. *The Stanford Encyclopedia of Philosophy (Summer 2003 Edition)*, 2003. Edward N. Zalta (ed).

[7] Thomas S. Ray. Frontiers of life, volume one the origins of life. *Academic Press*, pages 107–124, 2001. Renato Dulbecco, David Baltimore, Francios Jacob, Rita Levi-Montalcini (eds).

[8] Thomas S. Ray. How i created life in a virtual universe. *Originally published in French*, 1992.

[9] Thomas S. Ray. A proposal to create a network-wide biodiversity reserve for digital organisms. Technical Report TR-H-133, 1995.

[10] The search for extraterrestrial intelligence. http://setiathome.ssl.berkeley.edu/ David Anderson was the founder of Seti at Home.

[11] Gabrielle Walker. Ice magic. *New Scientist*, 12 April 2003:30–34, 2003.

[12] C. Ofria, C. T. Brown, and C. Adami. Avida user's manual. http://nemus.dllab.caltech.edu/avida/manual/manual.pdf.

[13] Carlo C. Maley and Eilis Boudreau, editors. *Seventh Artificial Life Conference*, 2000. Pages 158 - 161, Thomas, S. Ray, Aesthetically Evolved Virtual Pets.

[14] Brooks and Maes, editors. *Artificial Life IV Proceedings*, 1994. Pages 28-39, K. Sims, Evolving 3D Morphology and Behaviour by Competition.