Template Based Evolution

Chris Headleand
School of Computer Science
Bangor University
Bangor, Wales
chris@chrisheadleand.com

William J. Teahan
School of Computer Science
Bangor University
Bangor, Wales
w.j.teahan@bangor.ac.uk

ABSTRACT

This paper describes a novel approach to multi-agent simulation where agents evolve freely within their environment. We present Template Based Evolution (TBE), a genetic evolution algorithm that evolves behaviour for embodied situated agents whose fitness is tested implicitly through repeated trials in an environment. All agents that survive in the environment breed freely, creating new agents based on the average genome of two parents. This paper describes the design of the algorithm and applies it to a model where virtual migratory creatures are evolved to survive the simulated environment. Comparisons made between the evolutionary responses of the artificial creatures and observations of natural systems justify the strength of the methodology for species simulation.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Genetic Algorithms, Evolutionary Simulation

Keywords

Template Based Evolution, Subsumption, Grammatical Herding, Problem Solving, Multi-Agent Simulation, Implicit Fitness

1. MOTIVATION

Evolutionary algorithms, although inspired from real-life evolutionary mechanisms, diverge substantially from nature in the way that populations are replaced each generation, in the way that genomes are represented (as bit strings, for example), and in the way that selection, sexual reproduction, crossover and mutation are performed. A primary aim of the research described in this paper was to explore an alternative evolutionary method where populations are able to survive and breed freely in a multi-agent simulation within a virtual environment in a way that more closely echoes the way natural evolutionary systems develop over time. The motivation behind this research is similar to the work of John Holland [14] concerning the need to understand further the interplay between evolutionary and ecological processes and how changes in population dynamics affect ecosystems.

One application area for such research is in computer games where Artificial Intelligence is playing an increasingly important role [23]. With this in mind, consider a game set on an imaginary virtual world. In this world, there are various creatures that the player may interact with. We already know what each creature will look like, what senses they have, and various attribute information. How could we structure their behaviour to provide each player with a subtly different experience? Or ensure that if a player decides to replay the game from the start, the experience will remain unique? This paper describes a new technique that can readily be applied to these problems.

The primary objective of the research described in this paper was to develop a lightweight method to allow the simulation of an evolving, embodied situated virtual species. A secondary objective of the study was to move away from the classic fitness function paradigm and instead test the agent's fitness implicitly through their ability to survive within the virtual environment. A third objective was to see if a multi-agent simulation combined with evolutionary computation to derive behaviours based on a subsumption architecture could be effective at creating reactive virtual creatures that exhibit emergent phenomena.

2. BACKGROUND

The concept of evolving individual agents implicitly within a multi-agent simulation is not new and has been successfully used in a range of novel applications including product design [21] and species simulation [26]. In this section, we will look at the major ideas that inspired our algorithm and related ideas within the field of artificial life and multi-agent simulation.

2.1 Sexual Reproduction

One foundation of evolutionary computation is sexual reproduction, the production of offspring that inherit the qualities of the parents. Commonly this is achieved by taking the genome of parents, splitting them into components and recombining to form unique children. Several variants to this standard model exist affecting the crossover or the number of swapped genome components, but also the number of par-

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.

ents [32]. However, natural and artificial genetic recombination are to the most part significantly different and therefore research with artificial approaches would still benefit from investigating alternatives. In our algorithm, we wanted to focus on global population trends and so decided to take inspiration from swarm based methods such as Grammatical Herding (GH) [10]. Grammatical Herding uses a weighted average to move agents within the search space and this inspired the crossover method for the TBE algorithm. This alternative method of reproduction by generating the average of two parents has been applied before to other studies [19, 20, 24]. In some benchmarks, these methods have produced 'better' solutions than standard Genetic Algorithm (GA) aproaches.

2.2 Implicit fitness evaluation

As one of the aims of the algorithm was to simulate species evolution, we wanted to move away from explicit fitness functions and instead test fitness implicitly within the environment. Several methods have been proposed which implement implicit fitness evaluations. For example, [3] describes an experiment where agents are placed within a world with their fitness tested by predation and the competition for a mate. Implicit fitness has also been applied to the field of robotics where implicit evaluations were used to reduce the human constraints placed on autonomy [2]. This work formed part of a multi disciplinary study into artificial creativity. Another study [34] discusses the implementation of virtual predators with a limited visibility able to only perceive the bearings and relative distances of other predators or prey that they can see. The results showed that complex behaviour can emerge from these implicit, proximity-based, social interactions. The later of these two studies formed the primary inspiration for the design of the algorithm presented here and the proof of concept implementation.

2.3 Evolving multi-agent systems

As the algorithm is focused on simulating a population of agents, some inspiration was taken from other evolutionary multi-agent systems. One interesting area evolutionary multi-agent systems have been applied to is the modelling of human resource management [5]. In this model, each individual agent's genome is directly mapped to a phenotype, which represents its characteristics and behaviours. This formed part of the the inspiration for our attribute to behaviour mapping system. Additional inspiration was taken from Grammatical Evolution (GE) and Constituent Grammatical Evolution (CGE) [31, 9] . The authors of the work also discuss advantages of this type of simulation by noting that whilst it is difficult to perform studies of the interactions between people and their environment in the real world, evolutionary multi-agent system simulations can provide a test bed for controlled experiments. Another multi-agent system (Amalthaea) features two classes of agents co-evolving as part of an artificial ecosystem to preform search and filtering of information, with the environment this has been applied to being the World Wide Web [22]. Reproduction within the system is based on the relationship of the agent to the user. Useful agents are selected for breeding while under performing agents are destroyed. This form of fitness evaluation through the destruction of under preforming agents is conceptually similar to the 'environmental trial' fitness evaluation we use in our design.

3. ALGORITHM DESIGN

In this section, we will cover the design of the new algorithm focusing primarily on how a TBE simulation is designed. We will also discuss the genetic and subsumption components as well as the embodied situated approach to fitness evaluation.

3.1 Nomenclature

Various terminology is used frequently within the following sections in relation to the algorithm's design and implementation. They are listed here with an aim of adding clarity to further sections.

Environment This is the virtual world the agents are situated within.

Species This is the population of agents being evolved. The species specification contains the species attributes, the subsumption template (species template) and the possible behaviours available to each individual within that species.

Agents These are the unique members of a species each containing values representing the evolved actions and individual attributes.

Fitness This is how suitable an agent (i.e. individual member of a species) is for the specific environment it is situated within.

Trials These are the methods of testing an agent's fitness through environmental challenges.

Attributes These are characteristics associated with each agent being evolved. They are specified by unique values that can be considered as the equivalent of GE codons. Attributes include actions and traits.

Actions These are actions an agent can perform such as turning 180 degrees and mating.

Traits These are further attributes such as speed, lifespan, colour etc.

Behaviours These are procedures that are accessed through a mapping process.

Species Template This is the subsumption architecture 'brain' that defines the decision process of each individual agent within the simulation. This specifies all the possible sensory inputs, such as hearing specific noises or seeing specific objects and specifies the actions of the agents within the simulation.

3.2 Design

In the following sub-sections, we will discuss the design of the TBE algorithm. For the purpose of overview, the TBE algorithm is structured as follows:

1. Each agent has a unique genome made up of a series of attributes.

2. Behavioural responses to the environment are stored numerically in action variables that form part of the agent's genome. These numeric values are mapped during run-time to a specific action.

3. Each agent executes a virtual 'brain' known as the species template. This design is based on a subsumption style architecture.

4. Fit parents are able to breed if they survive in the environment. Weaker agents die out before they are able to breed offspring that have their inherited traits.

5. Offspring are produced with a genome that represents the average attribute values of two parents.

3.2.1 Genome

The genome in the TBE algorithm is a collection of numeric variables known as attributes, each of which can vary in length or numerical type (integer, double, float, etc.). Each attribute relates either to a trait (such as speed, vision or colour) or a possible behaviour stored numerically as an action. The TBE genome differs from that of traditional genetic algorithms (GAs) [1] in the way they are represented, using real values rather than binary strings to represent the genome. A TBE genome is made up of attributes that can contain variables of any numerical type or length, this approach is taken directly from the concept of "real coded genetic algorithms" [11].

3.3 Species Template

In TBE, numerical action variables are evolved and these are used to determine the behaviour of each agent using a pre-defined subsumption-based architecture [4] known as the species template. The template used is identical within each individual agent with only the values of the agents' actions varying. These attributes are used directly to decide which actions should be performed (see the procedure `runAction` in Algorithm 1). This does mean that an agent within a TBE simulation will always be limited in the possible ways they can respond to an environment. However, each agent will always execute a valid response, a limitation of some genetic evolution systems [25]. This has the subsequent advantage of reducing bloat and improving the efficiency of a simulation. A strength of the method is that it allows for conditional behaviour switching allowing various responses to be evolved in parallel. This is demonstrated in the pseudocode example (see Algorithm 1). In the example, if the agent sees a predator, it will send the Action 1 attribute to the runAction function, which will result in the agent turning 180 degrees.

3.4 Crossover

Classical generic algorithms work by selecting two components of a parent's genome and combining to form the offspring. In contrast, the TBE crossover method works by taking the average between the attributes of two parents (a TBE attribute can be considered equivalent to a codon for the sake of this example). This is known as an Aggregation-Based Crossover Operator (ABCO). ABCO methods dispose of traditional methods in favour of leading the population towards possible zones of high fitness [12] rather than explicit values. The aim was to reduce the probability that a child may be of lower fitness than the parents by taking the average of potentially high fitness characteristics of both parents during crossover. To ensure a strong level of diversity and a diverse search, each pairing of two parents may only produce one child.

Genome of Attributes: - Example

[Action1 = 0.7]
[Action2 = 2.3]
[Action3 = 1.1]
[Action4 = 3.9]
[Trait1 = 3.1] (Vision Length)

Species Behaviour: - Example

```
if predator within vision-length(Trait1) then
    runAction(Action1);
else
    if obstacle ahead then
        runAction(Action2);
    else
        if Mating ground reached then
            runAction(Action3);
        else
            runAction(Action4);
        end
    end
end
```

runAction (attribute) - Example

If attribute > 0 && <= 1 [Turn 180 degrees]
If attribute > 1 && <= 2 [Move forward]
If attribute > 2 && <= 3 [Move backward]
If attribute > 3 && <= 4 [Mate]

Algorithm 1: Species template example.

The inspiration for this choice of crossover came from swarm based methods where individual candidates trend towards higher fitness socially through movement within the possible search space. Grammatical Herding does this through a weighted average method between two agents [10]. The crossover method we use in our algorithm calculates the average attribute value of both parents in a similar way to how GH takes average positions. This method also allows for a large range of possible candidates to be bred and tested, as it does not rely on the attribute values present in the first, randomly generated population. However it could be considered potentially destructive, as the fit qualities of parents are possibly "forgotten" in the offspring.

3.5 Mutation

The crossover method is susceptible to bias. By taking the average of all attributes, the population will trend towards the mean of the possible range available within the population. As some agents will not survive the environment, the bias should be weighted towards higher fitness. However, without any mutation, the genetic diversity of the species will be limited as offspring could not exceed the maximum range of values present in the first generation.

As each attribute could potentially contain values of differing length and type, the traditional GA bit manipulation method is not appropriate for this algorithm. The new method implements a random value shift based on the range of possible values. The following equation defines how this is implemented:

$M = V + (Random(R/D) - Random(R/D)$.

where M is the resulting value of the mutated attribute, V is the value of the un-mutated attribute, R is the numerical range available for the selected attribute about to be mutated and D is a divisor which determines the size of the mutation. Behaviour attributes were bounded between specific lower and upper bounds with any mutation outside these boundaries not allowed.

4. MIGRATORY BIRDS SIMULATION

To test the suitability of the framework, an initial model was designed in NetLogo [37] to simulate the evolution of a migratory bird species. In this section, we will discuss the design of the simulation and the emergent properties of the agents that arise. Finally, we compare the results of the simulation to observations of real migratory birds to demonstrate the strength of the approach.

4.1 Inspiration and Background

In this section, we will discuss observations of migratory birds in nature. This background is used to both guide the design of the simulation and validate the quality of the end results.

Our simulation will focus on a species migrating through flight. Understanding how birds escape from predatory attack is a significant consideration as part of the design as it has been noted that in most birds, escaping through flight is the most important means of escape from predators [35]. One review of a century of ornithological observations categorised flight based escape tactics into speed based tactics, aerial dodging and socially coordinated escapes [18]. Speed based and aerial dodging escape techniques concern outrunning and outmaneuvering, respectively. In the latter, it is not clear why a predator would terminate an attack instead of hunting until successful. However, it has been proposed that the hunter may lose sight of the target or conserve energy to target less capable prey. Socially coordinated escape tactics refer to the possible defensive advantages of flocking [27]. Despite lack of understanding as to why this tactic is specifically effective, it has been observed that hunting raptors rarely dive into a large flock of birds [30].

Another important consideration when simulating a migratory bird species is how birds in nature migrate. It has been shown that birds have an attachment to the same site and will return to it seasonally [16, 13]. Research has also shown that birds may possess a magnetic compass [29, 38].

4.2 Design

In this section, we will discuss the design of the simulation relating it to the TBE methodology.

4.2.1 Environment

The environment created was a non-wrapping world large enough to allow the agents enough space to effectively flock. Within this environment, three adversities were devised to encourage evolution:

Predators

The first adversity was designed in the form of non-evolving 'dumb' agents known as predators. These agents were constrained to one third of the total space in the centre of the world known as the hunting ground. They followed two simple behaviours 'hunt' and 'patrol'. If the predators were able to see a bird within their field of vision, they would chase it. If they caught up to it, they would then attack and 'eat' it. If no birds were visible, they would patrol. In this mode, they would randomly wander within the hunting ground. If they reached the boundary of the hunting ground, they would turn 180 degrees and move back into the hunting ground.

Exposure

A second adversity was included to force the birds to migrate to survive. This was designed in the form of simulated exposure. If the agents did not make it to the appropriate mating ground within a fixed timeframe, they would die. The migration time was determined by four seasons – two mating seasons and two migratory seasons.

Limited Resources

Finally, a limit was placed on the amount of individual agents that could be bred each mating season. The aim of this was to simulate the competition for food resources.

4.2.2 Species

In this section, we discuss the species of migrating birds and the components common to each agent. This includes the species template, the available actions and the traits common to all agents.

The Species Template

The subsumption architecture was designed in two layers. The first layer, known as the instinctive layer, determined what the birds would do uniquely in each of the four seasons. The second layer, known as the reactive layer, allowed the agents to respond to predators, the arrival at the mating ground or the presence of another agent during mating. Each behavioural response within the species template was assigned two possible action attributes; this, plus a random selection function which chose which of the two actions to respond with, provided some variance in how individuals responded.

Attributes (genome definition)

Several action and trait attributes were set within the species as the genome definition. The trait attributes are defined below:

- Max-lifespan – Initial agents are created with a lifespan of 4000 time steps plus or minus a random value between 0 and 1000.
- Vision-Length – This specifies how far ahead the agents can see from their perspective.
- Vision-angle – This specifies how wide the vision of each agent is.
- Speed – This specifies how fast the agent can move within the environment.
- Instinctive – As mentioned in section 4.1, it has been hypothesised that birds in nature can navigate magnetically. The instinctive trait was a percentage chance that the birds would turn towards the heading of the next mating ground.

- Colour – This is a trait that specifies a numerical representation of a bird's colour, used for observation purposes.

- Friendly – This trait represented how close the agent would flock to other agents. If whilst flocking, another agent was within its "friendly" range, it would turn away.

Actions

The agents were provided with the following actions, with only the flee and move actions resulting in any forward movement:

- Flock – This results in the bird agent producing a flocking behaviour. How close the agents would flock to each other is governed by the 'friendly attribute'. This was adapted from the boids system proposed in [28].

- Flee – The bird turns 180 degrees and moves forward at 1.5 times the current speed value.

- Migrate – The bird turns towards the approximate heading of the current mating ground.

- Wander – The bird varies its heading randomly (left or right) up to a maximum of 20 degrees.

- Move – The bird moves forward one step, the distance of the step determined by its 'speed' attribute.

- Spin – The bird rotates by a random number of degrees.

- Mate – If the bird is within a mating ground and can see another bird, then it will move towards it. If it catches up with its selected partner, one new bird is bred.

- Freeze – The agent performs no action and remains stationary.

A limit was set on the possible speed and lifespan each bird could achieve to prevent mutation evolving unrealistic traits within the species. Additionally, a minimum age was set for sexual maturity. Agents younger than a season were unable to breed.

The way each agent was able to react to their environment was a combination of 12 decisions within the species template, each with 8 possible actions, the possible behaviour search space (ignoring the traits) is 8^{12} or 68719476736 possible combinations. As some of the trait attributes (friendly, vision-length and vision-angle for example) alter the resulting behaviour, this adds to the complexity of the possible search space.

4.2.3 Agents

500 individual agents were initially bred into each simulation run, and their genome was initialised with a set of random attributes. When first created, they are placed randomly within an initial mating ground facing a random heading. Each agent starts at age 0 and their age increases by 1 at each time step of the simulation.

4.3 Experiments and observations

125 experiments were run, each experiment comprising of 5,000 years, each year containing 2 mating and 2 migratory seasons which alternate, with each season equal to 1200 timesteps. The experiments were split into 5 sets. Set 1 contained no predators; Set 2 contained 50; Set 3 contained 100; and Set 4 contained 200. A final set (Set 5) had a steady increase of predators from 0, increasing at a rate of 1 predator every 32 seasons to 625 predators by the end of each experiment.

The agents in the Set 1 experiments all evolved to quickly migrate from one mating ground to the other, flocking closely, the only adversity they had to face being the competition for a mate. Other than this, there was no correlation in their behavioural responses. Eyesight traits that evolved were generally quite poor, the agents not requiring good vision to survive in this environment.

In Sets 2, 3 and 4, all the agents evolved high speed as a common tactic to avoid predators. There was also a direct correlation between vision length and width and the number of predators in the environment. In the Set 2 experiments, the birds evolved narrow forward facing vision with a long range. However, for all the Set 3 experiments, the birds evolved wide vision with a relatively poor range.

The Set 5 experiments showed that the birds were able to evolve multiple strategies to cope with environmental change. The same vision trends observed in Sets 2, 3 and 4 were observed over the time frame of the simulations. Additionally, the birds in this experiment evolved relatively short lifespans (2.5 seasons long) which allowed the species to quickly respond to environmental changes.

4.4 Emergent Responses

In this section, we will look at the emergent qualities the agents evolved in response to the environment.

4.4.1 Parallel Flocks

One interesting response was the parallel evolution of multiple flocks, usually two and occasionally as many as four. As the mating function was proximity based, this allowed small clusters of agents to create local mating grounds with groups of agents with unique traits. The offspring that came out of these separate mating zones would have their genome defined by a limited set of parents.

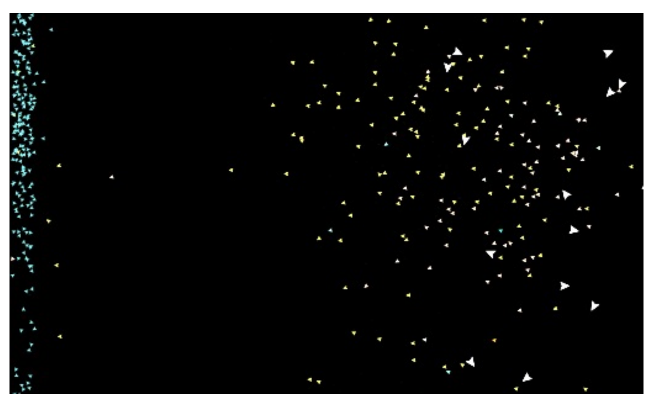

Figure 1: In this example, the right pack has learnt to migrate fast; however, the left pack have yet to leave the mating ground.

Through several generations, the two sub-species might evolve very different responses to the environment (see Figure 2). Interestingly, despite having no explicit mechanism that prevented it, the sub-species would rarely inter-breed. Instead, eventually the weaker pack would die out or be absorbed by the larger dominant pack. However, the two sub-species would often evolve in parallel for up to 7 seasons.

These early arrivals could soak up all/most/many of the available births for that season. In fact, this is probably why eventually one pack of birds would become dominant. However, this is somewhat mirrored in nature, in the connection between early arrival at a breeding site and sexual reproductive performance in long distance migratory birds [33].

4.4.2 Pulsing

An interesting behaviour occurred very occasionally at the end of a mating season. Instead of migrating across the environment as a group, the agents would move in pulses. After a roughly equal break, a line of agents would move in unison across the environment, slowly dispersing the further away they moved.

Figure 2: In this example, the agents can be seen clustering at the mating ground on the left. Two pulses of agents can be seen traversing the environment from left to right. The furthest pulse (right) has started to disperse; but can still be seen clearly.

This behaviour appeared like a wave when observed at high speed (see figure 3). This kind of pulsing behaviour is often observed in cellular automata; but we believe it to be quite unique for this type of simulation. Interestingly, the pulses generally contained agents of equal speed (with some exceptions) even though the range of speeds in the population could be quite large. At first, it was assumed that this was due to the flocking behaviour; however, the majority of the pulsing agents had not evolved to follow that behaviour. Also, flocking would have caused them to move towards a coherent group instead of dispersing as with the agents mentioned in [28] and the NetLogo flocking model that inspired the code responsible [36].

4.4.3 Decoys

A social response to the threat of predators was the behaviour of the decoys. In this behaviour, the agents would flock very closely. On approaching a predator, the lead agent would break off with the predator in pursuit whilst the other agents would remain in the flock moving quickly past. By tracking the number of birds killed by predators each season, this behaviour proved highly efficient in protecting a large number of agents.

4.4.4 Wide Eyed Birds

The 'wide eyed birds' behaviour occurred in simulations with a large number of initial predators. The behaviour involved the agents evolving poor forward eyesight, but with a wide vision angle and a loose flocking formation. On leaving the mating ground, the agents would flock, and would ignore the predators until they were within close range. At this point, the birds would scatter and move erratically through the environment making several turns, freezing and wandering. These quick changes of direction and velocity made it difficult for the predators to track the agents allowing a large group to escape.

By evolving poor eyesight, the agents did not react too early to the threat of predators and instead maintained a close flock. As soon as the lead agents were able to see the the predators, they scattered; any flocking agents were able to follow the leaders to safe routes or themselves scatter. If the agents evolved long range vision and a wide vision angle, it is perceivable that the agents would not have been able to face a forward direction without encountering an agent somewhere within their vision limiting their ability to progress through the environment.

4.5 Observations

A subset of the experimental runs were observed to note occurrences of the emergent phenomenon discussed above. The data in Table 1 details the amount of time these behaviours were visibly evident as a percentage of the total migration timesteps (mating season time steps excluded).

For each experimental set, 4 runs were observed in full, with the table detailing these specific observations. Each time a behaviour was observed, a start and end timestep were recorded, noting when the bahaviour first became apparent and when the behaviour had concluded.

	Set 1	Set 2	Set 3	Set 4	Set 5
Wide Eyed	0	1.34	11.30	15.31	14.28
Pulsing	0.06	0.09	0.02	0	0.05
Decoys	4.83	8.19	9.31	5.12	4.66
Parallel Flocks	3.23	2.01	2.20	3.63	5.11

Table 1: Percentage of non-mating time steps that a specific emergent behaviour was observed in the experiments.

4.6 Validation

Several observations were made that correlated closely to studies of birds in nature. In this section, we pick some of the more interesting to discuss. It is important to note that whilst these results are interesting, and although this simulation has been inspired by nature, conclusions about real birds cannot be taken from this relatively simple two dimensional model.

In the background to the initial investigation, we discussed studies of birds and their responses to predation. Specifically, we mentioned three types of aerial tactics that migratory birds may use to avoid attack. These were classified as speed, dodging and socially coordinated techniques. During the observations, we saw examples of all of these behaviours, and whilst the initial behaviour search space biased the possible responses, complex behaviour emerged despite the relative simplicity of the behaviour primitives programmed into the system. Specifically interesting were the socially coordinated responses (for example, the wide eyed birds response) which produced some realistic behaviour from the predators. As the agents were moving erratically, the predator followed a single objective ('hunt the closest bird in vision') and was often unable to catch any prey. This was because the closest bird changed direction regularly forcing the predator to make many turns. This operation would often slow the predator down enough that the migrating birds were able to escape, correlating with one hypothesis proposed in [18].

Another interesting emergent response was the parallel flocks, where multiple sub-species were evolved in parallel. This process of evolution constrained by geographic isolation (or Allopatric speciation) has been observed in many bird species [8, 6] most famously described in Darwin's "The Origin of species" [7]. Without any primitive behaviour that pre-disposed the agents to this action, it was still observed in several simulations. We correlated this behaviour with the arrival of the first birds able to breed in isolated areas of the mating ground, an affect mirrored in nature [17].

Also noted was the vision characteristics of the birds. In the simulations with many predators, they evolved wide vision. This is echoed in nature where a common trait of birds that are preyed upon are eyes on opposite sides of the head to give the widest possible field of vision [15].

5. CONCLUSIONS AND FUTURE WORK

A new attribute-based evolutionary algorithm has been described. For the simulation of artificial life, the algorithm has proven that it is capable of evolving agents by testing fitness implicitly within an environment. However, further research is needed to establish whether the new algorithm's crossover and mutation methods provide an advantage over methods used in traditional genetic algorithms.

Also, as the algorithm relies on a reactive subsumption based approach to structure the behaviour of the agents, there are currently no benchmark tests that would provide a fair comparison. The true benefits of this algorithm are the possible applications in species simulation or decision engines. If combined with a method such as Grammatical Evolution or Herding to continually develop unique behaviours within a reactive agent, the ability to respond to a challenge could be significantly improved. This could provide a further development to the traditional Brooks reactive paradigm by creating a reactive system with the ability to learn and evolve.

5.1 Possible Applications

In this section, we will discuss some possible applications of the TBE algorithm, which would suit its particular strengths.

5.1.1 Evolutionary Robotics

This work extends the work of Brooks subsumption architecture with the addition of an evolutionary component. With that in mind, the algorithm could be applied to robotic decision processes. As well as being able to react to an environment, a robot utilising a TBE system would be able to learn, evolving the best affective strategies to react with.

5.1.2 Virtual Creatures / Artificial Ecosystems

An objective of this initial investigation was to develop a lightweight method to allow the simulation of a virtual species. The algorithm has demonstrated that various emergent behaviours can be evolved through this method, indicating suitability in the field of artificial life.

The algorithm could be applied to games to allow NPCs and virtual animals to react and respond to changes in the environment caused by the player, creating a unique experience each time the game was played. It would allow the player (as an agent within the system) to have an affect on the development of the world around them.

6. ACKNOWLEDGMENTS

Chris Headleand's contribution to this research was supported by the Access To Masters project funded directly by the European Social Fund via the Welsh Government.

7. REFERENCES

[1] T. Bäck and H.-P. Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary computation*, 1(1):1–23, 1993.

[2] J. Bird, P. Husbands, M. Perris, B. Bigge, and P. Brown. Implicit fitness functions for evolving a drawing robot. *Applications of Evolutionary Computing*, pages 473–478, 2008.

[3] N. Brodu. Environmental fitness for sustained population dynamics. In *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, volume 1, pages 343–350. IEEE, 2005.

[4] R. A. Brooks. Planning is just a way of avoiding figuring out what to do next. Technical report, MIT, 1987.

[5] J.-C. Chen, T.-L. Lin, and M.-H. Kuo. Artificial worlds modeling of human resource management systems. *Evolutionary Computation, IEEE Transactions on*, 6(6):542–556, 2002.

[6] R. T. Chesser and R. M. Zink. Modes of speciation in birds: a test of lynch's method. *Evolution*, pages 490–497, 1994.

[7] C. Darwin and W. F. Bynum. *The origin of species by means of natural selection: or, the preservation of favored races in the struggle for life*. AL Burt, 2009.

[8] S. V. Edwards, S. B. Kingan, J. D. Calkins, C. N. Balakrishnan, W. B. Jennings, W. J. Swanson, and M. D. Sorenson. Speciation in birds: genes, geography, and sexual selection. *Proceedings of the National Academy of Sciences of the United States of America*, 102(Suppl 1):6550–6557, 2005.

[9] L. Georgiou and W. J. Teahan. Constituent grammatical evolution. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Two*, pages 1261–1268. AAAI Press, 2011.

[10] C. Headleand and W. Teahan. Grammatical herding. *J Comput Sci Syst Biol*, pages 043–047, 2013.

[11] F. Herrera and M. Lozano. Gradual distributed real-coded genetic algorithms. *Evolutionary Computation, IEEE Transactions on*, 4(1):43–63, 2000.

[12] F. Herrera, M. Lozano, and A. Sánchez. A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study. *International Journal of Intelligent Systems*, 18(3):309–338, 2003.

[13] J. B. Hestbeck, J. D. Nichols, and R. A. Malecki. Estimates of movement and site fidelity using mark-resight data of wintering canada geese. *Ecology*, 72(2):523–533, 1991.

[14] J. H. Holland. Echoing emergence: Objectives, rough definitions, and speculations for echo-class models. In *Complexity*, pages 309–342. Perseus Books, 1999.

[15] M. P. Jones, K. E. Pierce, and D. Ward. Avian vision: a review of form and function with special consideration to birds of prey. *Journal of Exotic Pet Medicine*, 16(2):69–87, 2007.

[16] E. Ketterson and V. Nolan Jr. Site attachment and site fidelity in migratory birds: experimental evidence from the field and analogies from neurobiology. In *Bird Migration*, pages 117–129. Springer, 1990.

[17] H. Kokko. Competition for early arrival in migratory birds. *Journal of Animal Ecology*, 68(5):940–950, 2002.

[18] S. L. Lima. Ecological and evolutionary perspectives on escape from predatory attack: a survey of north american birds. *The Wilson Bulletin*, pages 1–47, 1993.

[19] S. Ling and F. H. Leung. Real-coded genetic algorithm with average-bound crossover and wavelet mutation for network parameters learning. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, volume 2, pages 1325–1330. IEEE, 2005.

[20] S. Ling and F. H. Leung. An improved genetic algorithm with average-bound crossover and wavelet mutation operations. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 11(1):7–31, 2007.

[21] H. Liu and M. Tang. Evolutionary design in a multi-agent design environment. *Applied Soft Computing*, 6(2):207–220, 2006.

[22] A. Moukas and P. Maes. Amalthaea: An evolving multi-agent information filtering and discovery system for the www. *Autonomous agents and multi-agent systems*, 1(1):59–88, 1998.

[23] A. Nareyek. Review: Intelligent agents for computer games. *Computers and Games*, pages 414–422, 2001.

[24] T. Nomura. An analysis on linear crossover for real number chromosomes in an infinite population size. In *Evolutionary Computation, 1997., IEEE International Conference on*, pages 111–114. IEEE, 1997.

[25] M. Oltean and C. Grosan. A comparison of several linear genetic programming techniques. *Complex Systems*, 14(4):285–314, 2003.

[26] L. Pagliarini, A. Dolan, F. Menczer, and H. Lund. Alife meets web: Lessons learned. In *Virtual Worlds*, pages 156–167. Springer, 1998.

[27] W. K. Potts. The chorus-line hypothesis of manoeuvre coordination in avian flocks. *Nature*, 1984.

[28] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH Computer Graphics*, volume 21, pages 25–34. ACM, 1987.

[29] C. T. Rodgers and P. J. Hore. Chemical magnetoreception in birds: the radical pair mechanism. *Proceedings of the National Academy of Sciences*, 106(2):353–360, 2009.

[30] G. Rudebeck. The choice of prey and modes of hunting of predatory birds with special reference to their selective effect. *Oikos*, 2(1):65–88, 1950.

[31] C. Ryan, J. Collins, and M. Neill. Grammatical evolution: Evolving programs for an arbitrary language. *Genetic Programming*, pages 83–96, 1998.

[32] S. Sivanandam and S. Deepa. *Introduction to genetic algorithms*. Springer Publishing Company, Incorporated, 2007.

[33] R. J. Smith and F. R. Moore. Arrival timing and seasonal reproductive performance in a long-distance migratory landbird. *Behavioral Ecology and Sociobiology*, 57(3):231–239, 2005.

[34] I. Tanev and K. Shimohara. On role of implicit interaction and explicit communications in emergence of social behavior in continuous predators-prey pursuit problem. In *Genetic and Evolutionary Computation – GECCO 2003*, pages 201–201. Springer, 2003.

[35] P. J. Van Den Hout, K. J. Mathot, L. R. Maas, and T. Piersma. Predator escape tactics in birds: linking ecology and aerodynamics. *Behavioral Ecology*, 21(1):16–25, 2010.

[36] U. Wilensky. Netlogo flocking model. *Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL*, 1998.

[37] U. Wilensky. Netlogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 1999.

[38] R. Wiltschko and W. Wiltschko. The magnetite-based receptors in the beak of birds and their role in avian navigation. *Journal of Comparative Physiology A*, 199(2):89–98, 2013.