UNIVERSITY OF CALGARY

On the Origin of Robotic Species

by

Matthew Templeton

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTERS OF SCIENCE

DEPARTMENT OF DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE

CALGARY, ALBERTA
September, 2011

© Matthew Templeton 2011

Abstract

This thesis investigates the possibility of robots forming species as defined by biology. This is tested through the use of physics-based 3D simulations in which populations of robots are subjected to various conditions which have been shown to cause speciation in nature. Tests include ecological speciation, anagenesis, mutation order, morphological and non-morphological evolution, and genetic drift. Within the framework of evolutionary robotics, it will be shown that robots can evolve reproductive isolation and develop into species. At the same time, a novel approach and methodology for speciation simulations is also demonstrated. These results will help us better understand the population mechanics involved in artificial evolution as well as identify a potential tool for evolutionary algorithms.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In 1835, Darwin arrived at the Galapagos Islands to study as a naturalist. He was certainly not the first visitor to the island; a small population of birds had beaten him by roughly 2.3 million years[39]. The now famous Galapagos finches, a staple topic of most evolutionary textbooks, had immigrated from somewhere in the Americas. Eventually the birds were able to populate multiple islands in the chain, though the journey between landmasses was long enough to substantially separate them. Then began the slow evolutionary process of environmental adaptation. Some of the birds evolved larger beaks were better able to consume the large seeds on their island. Others became adapted to a different niche, and developed smaller beaks for smaller seeds. The original population began to diverge as differing groups evolved to exploit differing environmental resources. Despite their physical separation and dissimilar interests, small-beaked finches and large-beaked finches did inevitably come in contact and attempt to mate. The product of such a union, a hybrid offspring, found itself poorly equipped for either food source. Its mid-sized beak was unable to break large seeds and inefficient at consuming the small seeds as well[94]. These low-fitness hybrids did not perform well in terms of survival. In their inability to crossbreed successfully, the large and small-beaked finches had become separate species. Scientists have observed this same phenomenon of hybrid inviability through many different mechanisms in many different species; in fact most hybrids have lower rates of survival than either parent[41].

How might this be relevant to computing and robotics? The field of artificial intelligence has for many years been increasingly adoptive of biologically inspired processes such as evolution[104], neural networks[78], and even predator-prey relationships[108].

While much of the research has invested itself in optimizing these techniques, a macro-perspective has proven fruitful as well. For example, understanding how population mechanics change group behaviours is critical to the operation of swarm systems[114]. Platforms such as the Symbrion project [4] focus on how entire populations of robots, rather than individuals, adapt and perform. Small, simpler robots such as these have been shown to operate rather robustly and evolve together to suit their environment and task[114]. How then, is the evolutionary path of these robots impacted when presented with some of the environmental dynamics the Darwinian finches observed?

Robotic swarms are well recognized for their ability to remain functional as a group, despite failures amongst individuals[98]. This makes them ideal for hazardous environments, so space exploration has been proposed numerous times [117]. Imagine then, that a population of evolvable robots is sent to map and explore a distant planet or asteroid. Upon their arrival, some of them may find themselves in dissimilar environments. They will adaptively evolve as groups, optimizing their functionality for their respective circumstances, just as the finches did. But what will happen if these two divergently evolved groups come back together? This all leads naturally to the question 'Can robots form species'?

The goal of this thesis is to investigate this concept and to attempt to demonstrate the very first speciation event within a robotic system. This will be accomplished using the Framsticks simulation software and will further examine the impact of environmental differences, selection pressures and genetic drift to robotic speciation. Experiments will also look at the possibility of mutation order speciation, species arising from both morphological and non-morphological evolution, and speciation as it can occur from either branching events or within a single lineage. The most common method of delineating species, by reproductive isolation, will be used with consideration also given to more traditional taxonomic measures.

Should speciation within robotic populations be possible, a number of intriguing possibilities present themselves. Currently a number of methods, such as island models, are used to assist in maintaining diversity within evolutionary robotic systems. Speciation represents an emergent, self regulating approach for performing the same task. This allows multiple viable solutions to exist within a single population without requiring unnecessary barriers. An understanding of how species groupings form may also show a path to joining these divergent populations, gaining the most from the genetic diversity. This concept may even apply outside of the field robotics having application in a much broader study of artificial evolutionary systems.

On the other hand, a speciation event can also be thought of as the amount of divergence at which a crossover operator fails. This essentially blocks entire groups within a population from exchanging genetic information slowing evolutionary progress. Knowledge about the mechanisms that underlie this counter productive situation will help system designers in avoiding it. Finally, speciation events can be used to trace important evolutionary changes in robotic development. Existing methods of quantifying changes with measures such as genetic distance are lacking in context, whereas speciation denotes a functionally relevant divergence in a population. These events may prove to be important markers in the overall development and deserve further study.

As a general breakdown, the information is presented as follows: Chapter 2 will explore the concepts used throughout this thesis, particularly those relating to artificial evolution, speciation, and evolutionary robotics. Chapter 3 examines the research leading up to this work, while Chapter 4 discusses how this question has yet to be answered. Chapter 5 presents the details of the experimental setup and the simulation tool used. Chapter 6 has the final results and Chapter 7 concludes with a short summary and ideas for future work.

Chapter 2

Evolutionary Systems and Speciation

2.1 Evolution

Our world teems with life; it has invaded nearly every corner on this planet. Evolution, it seems, can find a solution to nearly any problem in time. The basic principles of evolutionary theory[24][69] have been well documented in innumerable texts, so it does not serve us to review them again here. For further reading please see [6][95][41]. However there are many nuances to the process of evolution that are of interest to this work; those will be briefly discussed.

A common misconception about evolution is that it is progress driven, but nature is not always so pragmatic. It is much more accurately described as a system of change, and to that end, not all change creates progress. And while we perceive evolution to be acting on the living organisms around us, it is often the genes of those organisms that were impacted first. This leads us to an important distinction, that all organisms can be considered from the view of their genetics (genotype space) or their physical bodies (phenotype space). Clearly these two concepts are inextricably linked as it is the genotype that encodes for the phenotype, but evolution sometimes acts upon them in very different ways.

Natural selection[24], a term that is often misunderstood as meaning the same thing as evolution, describes a force that operates largely in phenotype space. It describes the reproductive benefit attained by higher fitness individuals in a population, and is therefore governed by the behaviour and success of the phenotype.

This is in contrast to the concept of genetic drift [125], another important feature of evolution. When genes are duplicated to be passed on to the next generation, there inevitably exist copy errors, which are referred to as mutations. They are inherently random, and to the original point, do not necessarily progress the organism. Many of these mutations are fitness neutral, so while they do constitute change in genotype space, they do not create meaningful change in phenotype space [56]. This idea becomes relevant over long time scales, over which the accumulation of small mutations can amount to a very large genetic change.

The complex interplay between micro and macro forces on evolution is a recurring theme in the literature[123][97][81][42]. Some stress the importance of the behaviours of entire populations and species when studying evolution, while others are more interested in how the DNA is performing. It is, in fact, still an ongoing debate between scientists as to which of these two mechanisms plays a larger role in evolution [105].

2.2 Evolution in Computing

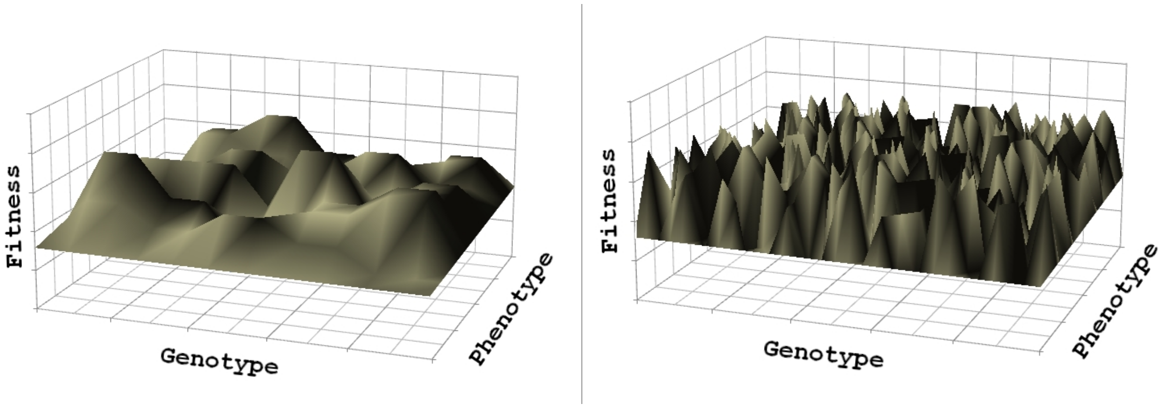
Though natural evolution may not have any intention of creating progress, its success in that scale has not been lost on computer scientists. For several decades, researchers have been developing methods called Evolutionary Algorithms (EAs)[127] that can be applied to engineering and computational problems. This is best understood from the ‘micro’ perspective of genotype space. Rather than the chemical DNA used in nature, EAs apply evolutionary forces to digital DNA.

2.2.1 Different overall approaches

There are many different methods of applying the concept of evolution to a computing environment; Evolutionary Strategies(ES)[8], Genetic Programming(GP)[61] and Genetic Algorithms(GA)[45] are some of the more popular classical designs. These will be cited for the purpose of providing examples of the differing techniques, but an in-depth exploration of their differences will not be made in this thesis. It is sufficient to say that no particular system is universally applicable, nor do all EAs conform to these specific procedural templates[127]. More precise definitions and comparisons of GP, GA and ES as well as discussions about appropriate uses can be found in [127][29].

Generally an EA works like a search algorithm, testing many different individual solutions in an attempt to find the best one. Its search space is often called a 'fitness landscape' and might be visualized as in Figure 2.1 for demonstration purposes. Every different sequence of digital DNA (or genotype) will map to a point on this surface showing how well it performed. One can imagine that very similar solutions may perform in similar ways, which is crucial for the efficient operation of an EA. By determining which genotypes have performed well, it can focus its search in the areas with the most potential. In Figure 2.1a the fitness landscape is smooth, and the EA may behave like a simple gradient accent algorithm. If, however, the landscape appears more rugged as in Figure 2.1b, the search will be more difficult because neighbouring solutions will perform less consistently. In order to effectively search a landscape, it is usually required that various parameters of an EA be tailored to that specific problem.

For example, if the purpose of the EA is to optimize an existing system, then implementing an ES may work well. This only evolves certain variables of a program while the overall structure remains constant. In this case it might make sense that the genotype simply consist of a series of real numbers. Alternatively, the entire program itself may be subjected to evolution. Just as DNA 'encodes' for organic bodies in nature, the digital genotype encodes for the digital phenotype, be it a computer program[61], neural network[78], or even an entire robot[104]. This can take the form of a bit string, integers, characters, or any data structure that is meaningful to the objective [127].



(a) Smooth Landscape

(b) Rugged Landscape

Figure 2.1: Fitness Landscapes

2.2.2 Fitness and Selection

Common, but not universal to all EAs, is the use of a fitness function. This formally quantifies the performance of an individual - a measure of how well it is doing in any particular environment. The ability of a fitness function to give clear directions to an EA largely defines how effectively it will progress. This information is usually used to determine which solutions should be allowed to progress forward, a process usually referred to as ‘selection’. It is another important parameter of an EA and can greatly affect the outcome.

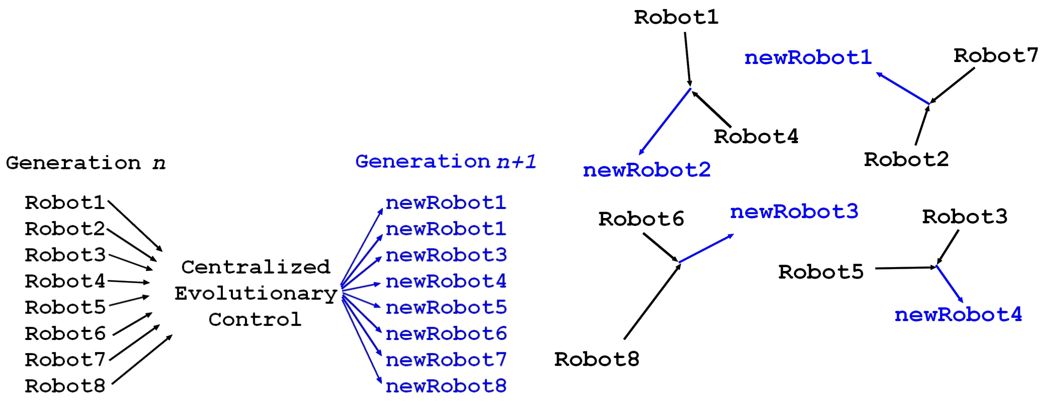
One method, as is often used in ES[8], is to only keep those individuals that have the best fitness in the population. This means the entire next generation may be derived from a single individual or some small percentage of the population. If the next generation does not differ sufficiently from the previous one, it may not effectively search the solution space. In this case, the EA will likely become trapped in a local optimum [4]. Not all systems of choosing individuals are deterministic, as many are stochastic in nature [127]. Selection can be made probabilistically based on their fitness, or even randomly. Whatever the case, it is important that the selection process ensure overall evolutionary progress while avoiding premature convergence. Some simulations do not explicitly calculate fitness and allow populations to mate/multiply to pass on their genetic information. This is common in Artificial Life (AL) systems which often rely on energy measures and death to control the population [7]. Once the selection has been made, EA must pass the current solution(s) from one generation to the next.

2.2.3 Control

A system that lacks a fitness function will usually lack centralized control as well, another important variation in artificial evolution. Centralization/decentralization is not so much a parameter of artificial evolution as it is a description of how the parameters operate together. The difference lies in where the evolutionary algorithm is located or processed; in a central hub or in a distributed fashion (Figure 2.2a). Many evolutionary systems are run on a single computer so the non-distributed method tends to be the norm, though both have advantages. Centralized evolution gives the experiment designer great control over the various parameters such as selection and can also format the system into discrete generations. This is necessary for some of the examples in the previous section, such as generating the next generation from a single genotype. Decentralized control, on the other hand, represents how nature operates and is capable of generating unique and unforeseen solutions (Figure 2.2b). This also promises better scalability and is less compromised by issues of locality.

2.2.4 Passing on Genes

The method of passing on genetic information from one generation to the next is another critical parameter. The simplest method is to transmit the data directly; this could be thought of as cloning or copying. However this cannot be used every time or there would be no variation between generations and therefore no evolution could occur. Instead, a number of different genetic operators can be used to act upon a selected individual to



(a) Centralized Evolution (b) Decentralized Evolution

Figure 2.2: Controlling Evolution

create variances in the data. The two most common methods are mutation and crossover.

2.2.5 Crossover

Crossover, also called recombination, finds its inspiration in the sharing of genes through breeding as is found in nature. A simple example might include two parent genomes that have been selected from the population to pass on part of their genetic information. A segment of each genome is copied and when the segments are combined, they form a new 'offspring' genome (see Figure 2.3). In this case only one new genotype is created, but commonly there are two offspring generated [119]. This is done in the hopes that the new genome will show progress by outperforming its parents. For many years, researchers of EAs have debated over how, and in what situations, crossover should be applied [50]. It is usually considered essential in GAs and GPs but less so in EP and ESs [107]. It has also been argued that the operation can be damaging to genetic information and may do more harm than good. This would be particularly true if the algorithm ignores structure inherent in the data when choosing where to 'cut' the genome[68]. So, depending on the application, some programs will make random cuts while others are more delicate when splicing the data. Another parameter of a recombination operation is the number of cuts made to the parents' genomes before reassembly. Theoretically this is only restricted by the length of the genetic information though it is traditionally limited to one or two cuts [107]. EAs are also typically either asexual or sexual systems, by having one parent or two parents, respectively, pass genetic information to offspring genomes. These are not fixed values and any number of parents can be used in a crossover operation, though the use of more than two is not as common [28].

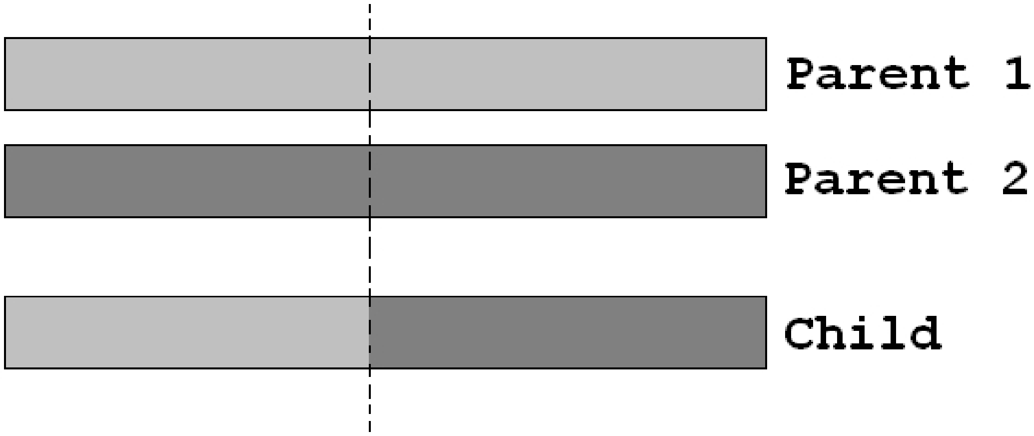


Figure 2.3: Crossover operation

2.2.6 Mutation

Mutation operations are widely used in EAs of all forms when transmitting genetic information between generations. It differs from crossover in that it can create new genes and can add fresh genetic material to the genepool [29], as opposed to only drawing from the existing genetic data. For simplicity, consider the binary genome described by Figure 2.4. The mutation operation selects random positions and changes them, in this case by flipping the bit. The method of selection is often based on some probability that any given position will be mutated. Again the hope is that the new genome, though very similar to the parent genome, may have a higher fitness. Reversing bit values works well for binary systems, but as the complexity of genome data increases, the complexity of the mutation operation often needs to match as well.

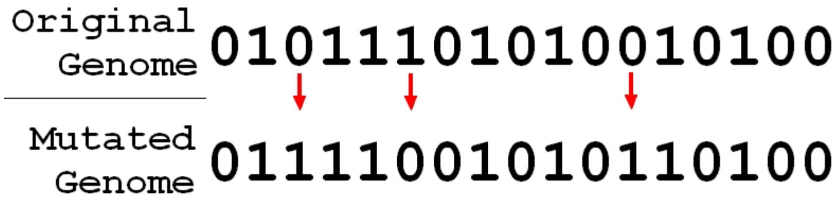


Figure 2.4: Mutation operation

The rate at which mutations occur also has a great impact on evolutionary progress.In humans mutations occur at a rate of only 10^{-10} [6] (per base pair per replication). This is very slow but a necessary consequence of avoiding high variation mutant genotypes. For organisms in the real world, producing or raising low fitness offpsring comes at a great cost. Computers on the other hand can delete poor offspring without moral consequence, so a higher mutation rate is reasonable. The tradeoff between generating low variation and high variation genotypes is critical to the selection of genetic operators. These opposing values are often referred to as Exploitation and Exploration respectively. Returning to the fitness landscape concept, in Figure 2.5 three different genotypes have been mapped to their respective points. A lower mutation rate will likely yield lower variation between generations, an example of exploitation. This allows an EA to 'exploit' the potential of current solutions by more effectively searching the nearby fitness landscape. In contrast, exploration through higher variation will search more distant genotypes looking for increased fitness. Both are important to the overall success of an EA.

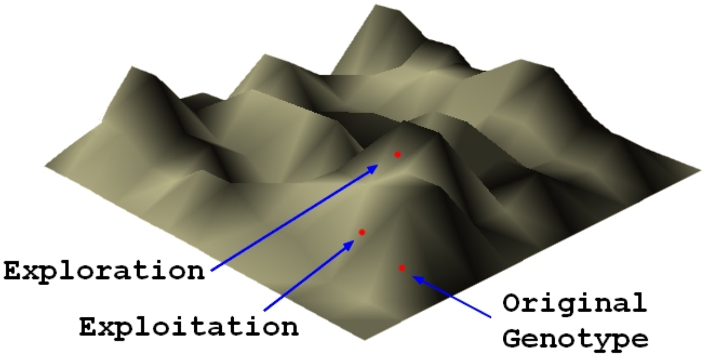


Figure 2.5: Exploration vs Exploitation in EAs

2.3 Evolutionary Robotics

At the most basic level Evolutionary Robotics (ER) uses the methods of Artificial Evolution, such as Evolutionary Algorithms(EA), and applies them to robots. Robots are, in the simplest sense, defined as "a machine capable of carrying out a complex series of actions automatically" [82]. This encompasses many different types of devices from industrial arms such as Figure 5.2a to the androids of science fiction novels. The majority of the robots considered in this work are Autonomous Robots, i.e. "robots that move freely and without direct human supervision"[119]. The general goal of ER is to explore ways for these robots to develop and adapt without the help or guidance of humans - in a word, to evolve. Broadly, this encompasses every aspect from controller design and social structure to the physical design of the robot itself. It also teaches us about the mechanics of evolution and biology, giving us insight into a natural phenomenon that occurs at a rate too slow to observe in the real world.

The earliest experiments into genetic algorithms did not use robotic systems, so why has ER become so popular? Robots provide an interesting platform on which to test development techniques for two reasons. Robots with a neural net controller use a sensorimotor loop to control their behaviour. This describes the cycle of information as the controller instructs actuators, actuators impact the environment and the environment is perceived back through the controller. This setup can draw on very simple solutions to solve for more complex problems[36][119]. Robots also present an opportunity to work with a complete system [108], one that can actually be used and tested outside of a computer if necessary.

Akin to many other ER experiments, this thesis will make use of multiple robots operating in the same space [122][55][74]. Robotics literature has long recognized the value of using many simpler robots over an individual complex robot in performing certain tasks [15][84]. For example, multi-robot systems are more fault-tolerant - they continue to function even if some of the robots are damaged or destroyed. Populations of robots can also be comprised of simpler and more cost-effective robots, rather than a single expensive one. Individual robots may also lack the scalability of multi-robotic systems- though one single complex robot may ably perform a task, adding more complex robots may not improve performance. Finally, they also allow for parallel evaluation as discussed in Section 2.3.2.

This field of ER began as an attempt to develop controllers without undue interference or influence of the programmers. The hope was that this would allow robots to generate unique solutions to problems that humans might not have considered. These experiments generally attempted to optimize the robotic controller towards a specific goal or behaviour. Many studies have noted the ability for evolutionary algorithms to find and exploit niche solutions to general robotics problems ranging from locomotion [66][12]to vision systems [5] [91].

2.3.1 Current uses of ER

This area of study has developed with a much more robust and complicated set of problems. These could be summarized as generally belonging to three areas of focus: optimization, emergence, and studying life through ER. Much of the foundational work in ER focused on demonstrating that functional controllers could be generated without human guidance [74][62]. Many hoped that this would eventually save time and effort on the part of programmers [5][62][93]. It might also allow robots to dynamically adapt to changing environments, or simplify the programming in multi-robot systems[60].

Those researchers aimed to create specific behaviours in robots using ER, while others study how the evolutionary process reacts to the rules and environment it is presented with. Emergent and unexpected behaviours are a common and fascinating aspect of the work. In several ways, ER could be compared to a form of developmental psychology for robots; rather than creating robot minds, we are allowing them be created, while trying to understand how to best work with them. Studies that focus on this are often of interest to the AL community as well. For example, ER experiments of predator-prey interactions such as [108][37] may teach us about similar systems in nature.

2.3.2 Parameters of ER

The experiments performed in ER are varied, but generally follow the same pattern. An initial population is formed, each with a genome. These genomes can be used to encode only a portion of the robot such as the controller, or potentially the entire robot. These robots begin to interact with their environment in such a way so as to allow the researcher to grade them based on a fitness function. Individuals are then selected from the population based on their fitness and new genomes are generated from theirs. In the case where the new genomes are generated from a single parent genome (asexual) some form of genomic mutation is usually performed. If two parents contribute to the new robot (sexual), a crossover operation is also used to join the genomes. These new genomes define the next generation. The experiment can run for a specific number of steps, or until a certain fitness is achieved, or can be run indefinitely.

There are a number of variables and methods that can be manipulated to change the behaviour of an evolutionary robotics system. We will explore those most relevant to this thesis, specifically controller choice, morphology, initialization, place of execution, use the simulations and distal/proximal considerations. For a more complete overview of other variations in ER see [108][119][106].

Controller

The controller, or the brains of the robot, can be implemented in a number of different ways. Researchers have tried tree-structure programming [65][62] and unstructured chips are discussed by [106] and implemented by [38][116] to name a few, but overwhelmingly, they have used neural networks [106, 122, 35, 90, 124]. Researchers have discovered that robots with neural network controllers are more insensitive to noise [106]making them more robust. Neural networks are also capable of ontogenetic learning [78], that is, they can develop behaviours and functionality that they did not directly inherit.

Morphology

A common practice in ER experiments, particularly in early studies, was to begin with a functional robot body but a non-functional controller. Later, research progressed and began to incorporate certain physical features on the robot, such as the placement and design of the sensors [85][103] along with the controller. Other experiments were able to demonstrate that the co-evolution of the brain and body resulted in operational improvements especially as task complexity increased [11]. However, the physical evolution of robots need not be constrained to the location and function of its sensors. "Intelligent behaviours arise out of the coupled dynamics of mind body and environment. It follows that the complexity of the controller and morph must match the complexity of the environment"[3]. With this in mind, some researchers have taken morphological development a step further and have allowed the entire structure of the robot to be slowly altered [70][109]. Much of this work can be likened to Karl Sims early work[104] developing physical creatures with a focus on locomotion. The co-evolution of body and brain "in lockstep" [106] has exciting potential, but the evaluation of results in real robots is challenging. Lipson and Pollack [66] famously demonstrated how morphologically evolved robots could be exported to the real world. The robotic bodies that were designed by the experiment were exported to a rapid prototyping machine that manufactured them out of plastic. Motors and a microcontroller programmed with the evolved neural network we connected, and the robots were shown to function. While this did effectively demonstrate the viability of evolved robot bodies, it was also very expensive (using today's technology). Once again the 'cost factor' showed that real world trials, while possible, were mostly impractical.

Initialization

The researcher can also manipulate the initial setup of the experiment to achieve different results. For example, as mentioned earlier, most experiments begin with a functional robot body but a non-functional controller [11][60][35][75][103]. However, initializing the controller without any inherent functionality or structure may lead to the bootstrap problem [119]. This describes the well established issue in EAs wherein the minimal solution is too complex to be generated without assistance. Also, if the controller is generated randomly, it might result in destructive behaviours, whereas a predesigned initial controller might restrict or at least influence the evolutionary development. The design of the body need not be fixed either, and this carries with it some of the same decisions. The robot might begin as a functional creature [85][4], or both body and mind may be evolved from scratch [66].

Place of execution

Much of the earlier work in ER used a centralized computer to perform the evolutionary algorithm [35][72][120]. However this is not the only option; the EA can also be executed in a distributed fashion. This subset of experiments in ER has arisen, rather pragmatically, from early work described by Watson et al. In [33] it is argued that under certain circumstances, a real population of robots might not have the ability to evolve using a centralized network. Under those circumstances, it is unrealistic to require that all information, including the fitness and genotype of every robot, be run through a single computer. If, for example, the robots were spread across a large area and communication was localized, a central computer would be ineffective.

This technique, called Embodied Evolution (EE) allows the robots to be freed from the restraints of a centralized system. This methodology promises both scalability and parallelism, but more inspiringly, simulates greater functional independence in its robots. It is also argued that this is more similar when compared to a natural evolutionary system [122] [108]. It is not necessarily more efficient though, as a centralized system may be able to better calculate a path through solution space.

Real vs Simulated

Robotics tends to be an application-focused area of study - if results cannot be applied to real robots, then researchers may become disinterested. Taking this pragmatism into account, several important developments have impacted the methodologies of ER. Almost inevitable in an ER paper is a discussion regarding real versus simulated experiments. This topic has been reviewed many times, so only a brief overview of the points will be made here. It is often cited that “the world is its own best model”[14], summing up the basic criticism of simulations - that they are not a completely accurate representation of reality. There is a fidelity to the real world which cannot be recreated, or it is at least intractable. The ”reality gap” [49] describes an issue where robots that learn in a simulation may not have sufficiently accurate sensory data to develop meaningful behaviours.

Despite these very important points, the merits of simulation have persuaded many researchers. Primarily, the argument in favor of simulation is that it is far less expensive in both time and money. Many papers propose experiments that require only a few hours using simulators but estimate that they would take years to complete in the real world [76][13][83].

Using real robots also carries a host of problems. Simple wear and tear has been reported to require maintenance [122], and physical damage is commonly cited as a reason to not use them [121] [120][106] especially in earlier stages. The absence of any requirement to supply spare parts or mechanically maintain the robots makes simulated work very appealing. Simulation testing also allows researchers to test many different variations quickly without undertaking time-consuming and repetitive setup procedures. Experiments that study the behaviors of robotic populations, such as this one, tend to prefer large numbers of agents, which further increases the costs. Other researchers have used a single robot and tested the genomes in serial order [36] but this deprives the robots of any interaction with each other. In an effort to assuage the concerns of other researchers, [65][74][120] describe running the majority of the experiments in simulation, and then at the end, porting the controller out to a real robot as a proof of concept.

In a boon for those working in simulation [49] effectively demonstrated that the use of artificial noise, applied to the sensorimotor loop, could negate some of the reality gap. Prior to this, researchers had pointed out the unrealistic feedback received by simulated sensors- it was always perfect data. [65][72] attempted to use actual sensor data for simulated training, but this proved difficult and painstaking. [76] later proved that excessive noise could also harm the evolution, but overall the addition of noise helped evolve more robust controllers. This study might benefit from the use of real world robots, unfortunately morphological evolution is prohibitively expensive using current technology.

Distal / Proximal

Another important consideration in judging an experiment's realism/portability to the real world is the information used in the fitness function. [77] refers to this as distal/proximal and [36] as external/internal. These authors argue that the experiment should consider what information is likely to be available in a real world trial. For example the use of a human to identify and select against pathological behaviors (such as becoming stuck [74]), that the robot itself is not able to identify. Other information that is easily accessible in a simulation is difficult or expensive to derive for real robots such as their exact location. Even the velocity of the robot at any given point might be difficult to calculate with any precision. This largely constrains the variables that can be used in a fitness function. A fitness function which has elements that are difficult for real robots to independently calculate such as [53][74] is 'distal' or 'external' and is perhaps less meaningful for an ER experiment. If the test results are meant to be applicable to robot development, it is important to maintain a proximal approach such as [122].

2.4 The Formation of Species

In biology, evolution can be seen as creating bio-diversity. Organisms are constantly adapting and changing to meet the needs of ever varying environments. The basic unit of measurement [9] [1], indeed the very cause of this diversity, is denoted by the term species[42].

2.4.1 The Macro View of Speciation

To begin a discussion on speciation, it makes sense to first look at what a species is. The most common definition [6] is "groups of actually or potentially interbreeding populations" [69]. So in order for two organisms to belong to the same species, they must be capable of producing viable offspring (this is explored more thoroughly in Section 2.6). Speciation, quite simply, is the process by which a species is formed. The term 'speciation event' is rather misleading, as only on a geological time scale would one consider it to be an 'event'. In reality, speciation is believed to occur gradually over thousands or even tens of thousands of generations [48] and can take millions of years [95].

Classical Scenario

The classical scenario [39] of speciation follows a hypothetical population that is split geographically, forming two different groups. This could be the result of a mountain range forming through the middle of their habitat range, or perhaps by chance, they may populate a distant island. It can then be imagined that the two islands they inhabit differ ecologically in some way. The predators may be different, or the available food sources may not be the same. Perhaps a larger body size is beneficial in one environment but detrimental in the other. This puts a unique evolutionary pressure on either population, forcing them to evolve and adapt along potentially divergent paths. If enough time passes and enough of these differences accumulate, they may come to behave or look dissimilar.

These changes and adaptations could prevent them from breeding with each other, or a hybrid offspring born from a parent of each group would be inviable.

It has been over 150 years since Darwin first presented this model for the formation of a new species and we have only recently begun to understand just how accurate it is [101]. Darwin's basic premise, that selection towards adaptation drives evolution and therefore speciation, has been contended with for many years. Indeed, over a hundred mathematical models have been used to address this very question [123], and to understand what other forces might be shaping evolution. For some time, there was some agreement that it was theoretically possible for selection to be the driver behind speciation, but that it might be rare. However, recent studies and a refinement of our models have changed this opinion, and it is currently held that this classical version of speciation occurs frequently [95][101]. This is sometimes referred to as Ecological Speciation to clarify that it depends on factors in the surrounding environment to create change[80].

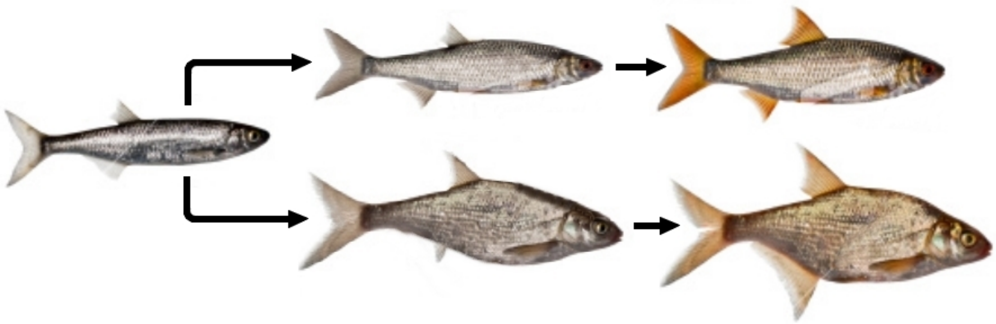


Figure 2.6: Speciation: The Development of New Species[0]

Barriers to Gene Flow

A critical concept to speciation is that of 'Isolation Mechanisms' that exist between groups. The simplest example was the one used in the Classic Scenario - a geographic barrier. A mountain range or large body of water prevents the separated groups from

---

[0] For demonstration purposes only - does not describe a specific speciation event

breeding, effectively isolating them from one another. Isolation mechanisms can take many different forms though, and not all are so direct. For example, two species may have different mating seasons so they will never attempt to breed; this is a form of behavioural isolation. Differences in body size or sexual organs that prevent crossbreeding are an example of mechanical isolation. Any one of these may not be enough to completely stop gene flow between the two populations (e.g. the geographic barrier may only partially block transit between ecosystems). So it is often the case that a combination of two or more of these isolation mechanisms act on a population during the course of speciation [105]. In the end, the defining element in speciation is the development of some Reproductive Isolation. This describes any set of isolating mechanisms that combine to prevent viable breeding.

The isolating mechanisms can interrupt various processes of mating in order to prevent breeding. In the examples of behavioural and mechanical isolation, both act to prevent mating from happening in the first place. In various forms of genetic isolation (discussed in Section 2.4.2), mating occurs but does not generate an embryo. These cases are referred to as pre-zygotic isolation - when no offspring is born. Post-zygotic isolation is the case where two species can produce 'Hybrid' offspring but they are considered inviable for other reasons. Typically they are either infertile, or the offspring may be deformed and/or otherwise deemed 'low-fitness' [41].

Completeness

Speciation is often thought of as a binary event, as either having occurred or not having occurred. This simplistic view ignores the continuous nature of speciation - it is a very gradual process and has varying degrees of completeness [80]. The fact that the process of speciation has begun is no guarantee that the two species will remain stable and separate. Even though it has been initiated, it may still require additional isolating mechanisms to progress, or possibly revert back to a homogeneous population. Figure 2.7 describes the overall pattern and stages wherein mid-speciation may be identified by ongoing hybridization and similar genomes.

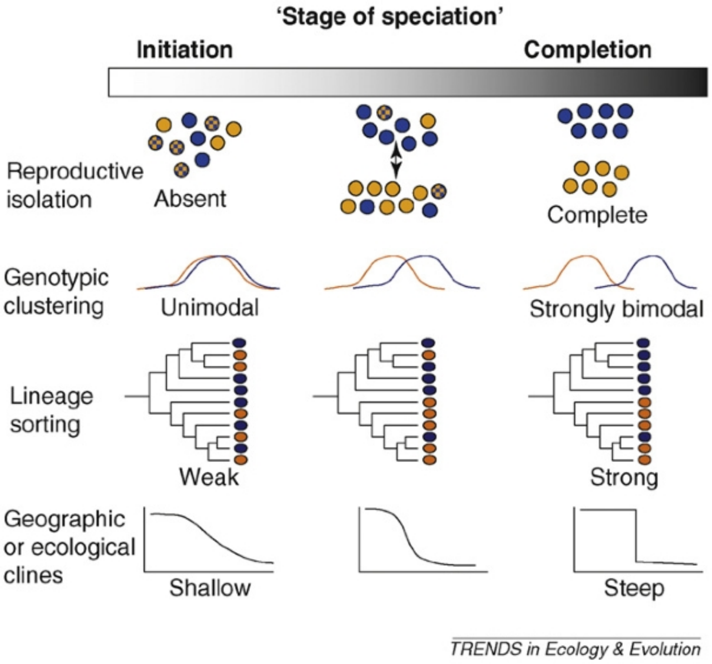


Figure 2.7: Speciation continuum [80]

Reinforcement

It has also been shown that speciation can be perpetuated through mate selection preferences. Consider a case of ecological speciation where only post-zygotic isolation has become fixed, and assume that two populations have diverged significantly enough so as to only produce low-fitness hybrids. Intuitively, these hybrids are less likely to survive to maturity and are therefore less likely to breed. Even those that do reach maturity may not be selected for mating by partners, further decreasing the chances of a hybrid genome passing to the next generation. Selection against hybridization, a form of 'reinforcement', creates an extra barrier to gene flow between the two populations and is thought to play an important role in the rate of speciation. As an isolation mechanism, mate selection occurs before breeding and is therefore pre-zygotic. This illustrates the manner in which post-zygotic isolation such as ecological adaptation can lead to pre-zygotic isolation as well [123].

Alternative Theories

Few evolutionary biology textbooks are complete without a discussion of alternative models for speciation. The definition chosen for this thesis is, again, the most influential but not the only one. Defining species groups based on viable offspring is called the Biological Species Concept (BSC). The BSC does not address all forms of speciation and so, a number of competing theories, no less that 22 [52], have sprung up. Most other speciation routes are not relevant to this work; however there are two that may prove to be of interest. The BSC requires that members of two populations be able to, at the very least, attempt to mate. By definition this only applies to sexual reproduction - asexual reproduction requires only one individual. Intuitively, this also requires that both individuals be alive so it cannot be used to differentiate between species that are extinct. The classical scenario involves a single population that splits and forms two different species, neither of which becomes extinct. This process can be more generally referred to as 'cladogenesis' and is the focus of this thesis. 'Anagenesis' on the other hand describes evolutionary change within a single lineage - it does not involve a branching of species[22]. It occurs slowly within one species as it accumulates enough change and gradually becomes a new species. It is in this case that the original species has typically gone extinct in which case it is not possible for the BSC to be applied. In this situation, scientists have typically made use of a taxonomic systems such as the Morphological Species Concept(MSC), or its more current version the Phenetic concept [41]. This is how one would imagine species were classified before the concept of evolution had been proposed - by their physiological traits. The basic definition is"assemblages of individuals with morphological features in common and separable from other such assemblages"[47]. While this may be more intuitive it is also considered to be more subjective as a method of delineation[39]. In the case of Anagenesis, the division made between an ancestral species and a new species is often somewhat arbitrary. The solutions to these problems are numerous and not well agreed upon; for more information see [6][41].

2.4.2 The Micro View of Speciation

A very different way of looking at how speciation works is to take the perspective of microbiology and understand what role it plays. This largely focuses on the underlying genetics and how it both affects and is affected by speciation. These theories are not competitive with the concepts found in ecology; it is only through an understanding of both that we can see a clear picture of how speciation works.

Depending on who one talks to, it might be suggested that we understand the micro view best [123] or that we still know very little about it [101]. Regardless, the volume of information we do have is beyond the scope of this work; for a recent survey of the literature, see [81]. Our short summary begins with the Dobzhansky-Muller theory of genetic incompatibility [25]. Consider again the classical model in which two populations begin the process of divergent evolution as in Figure 2.8. In the second step both populations have seen an advantageous mutation arise in their respective gene pools. Gene A mutated into a in one environment and B mutated into b in the other. With these new genes giving some adaptive advantage, they eventually become fixed in the population. The crux of the Dobzhansky-Muller model shows that while the a and b genes are beneficial on their own, they are incompatible when joined together. Therefore a hybrid having a genome of ab will see some phenotypic disadvantage, possibly sterility or generally lowered fitness.

Therefore, the post-zygotic isolation and hybrid inviability can actually be the result

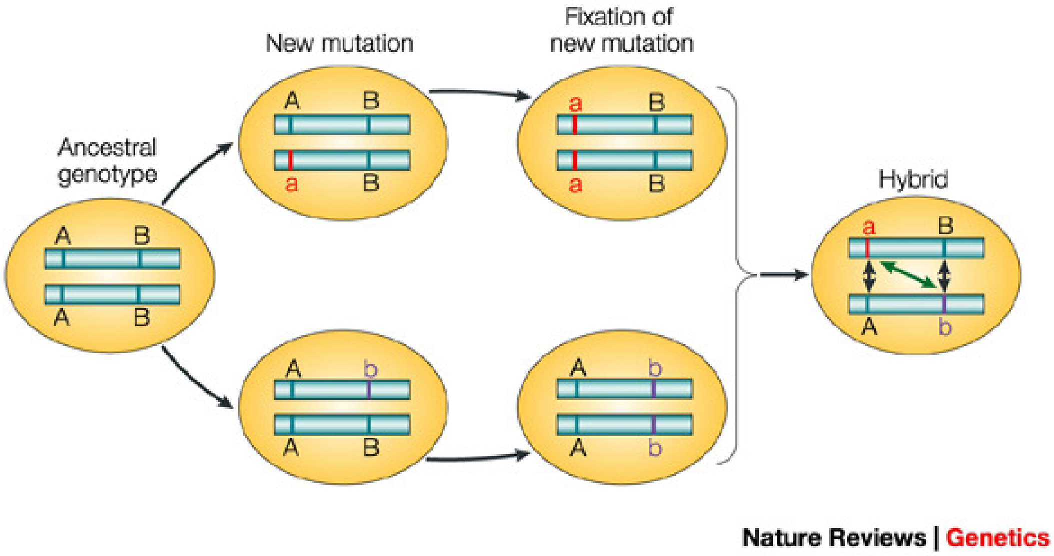


Figure 2.8: The Dobzhansky-Muller Model [126]

of genetic incompatibility. But not all isolating barriers are considered to be genetically derived, for example, behavioural isolation may not stem from a Dobzhansky-Muller incompatibility [47]. Additionally, morphological differences may point to genetic differences, but there is no guarantee that they will translate into reproductive isolation. This has led researchers to look for 'Speciation Genes' - areas in the genome "that reduce hybrid fitness"[81]. Many experiments are ongoing, and the results are contentious, but science is slowly unraveling this puzzle [126].

Chapter 3

Current Research and Other Systems

This chapter outlines some of the research and experiments that have led up to the work performed in this thesis. First, will be a summary of various evolutionary robotics systems, both physical and simulated. Then a look at some of the research that has been done on speciation from the biology and ecology perspectives.

3.1 Evolutionary Robotics

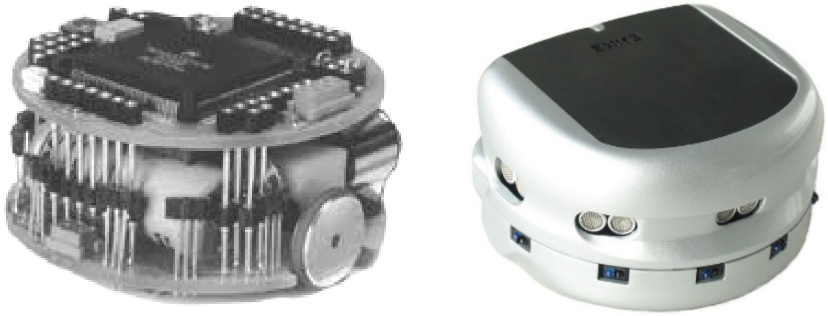


Figure 3.1: Khepera robots [51]

The Khepera series of robots (see Figure 3.1) are an excellent example of a popular platform for ER experiments and have been used for over a decade[36]. They provide an array of IR and ultrasonic sensors and can be upgraded with vision or radio modules. Some of these experiments trained the robots to perform non-trivial behaviours such as box-pushing [65] or obstacle avoidance [72] [88] in order to test various optimization techniques. They have also been used to test predator/prey scenarios [35][124] for both Artificial Life research as well as to learn about co-evolutionary systems. Unfortunately the Khepera robots are relatively expensive so some researchers do not use the actual robots for testing. Webots, as well as other software tools such as Matlab [51][111], can be used to simulate the khepera robot to save time[93]. Some experiments are run completely in simulation [124][88], while others argue that a combination of both simulated and real Khepera robots is best [72].

Another physical robotic system used in ER is the S-Bot. This is notably different from the Khepera in the addition of a gripper device that allows the robots to physically attach themselves to each other to and perform functions as one large ‘Swarm-bot’. The system is designed to explore the dynamics and potential benefits of swarm systems. While some still use it for testing evolutionary optimization techniques [18], it is well suited for evolving biologically inspired multi robot behaviours such as aggregation [113], coordination [114], and signaling [115]. Again evolution in a physical system is slow, and some experiments opt to simulate the S-bot [26].

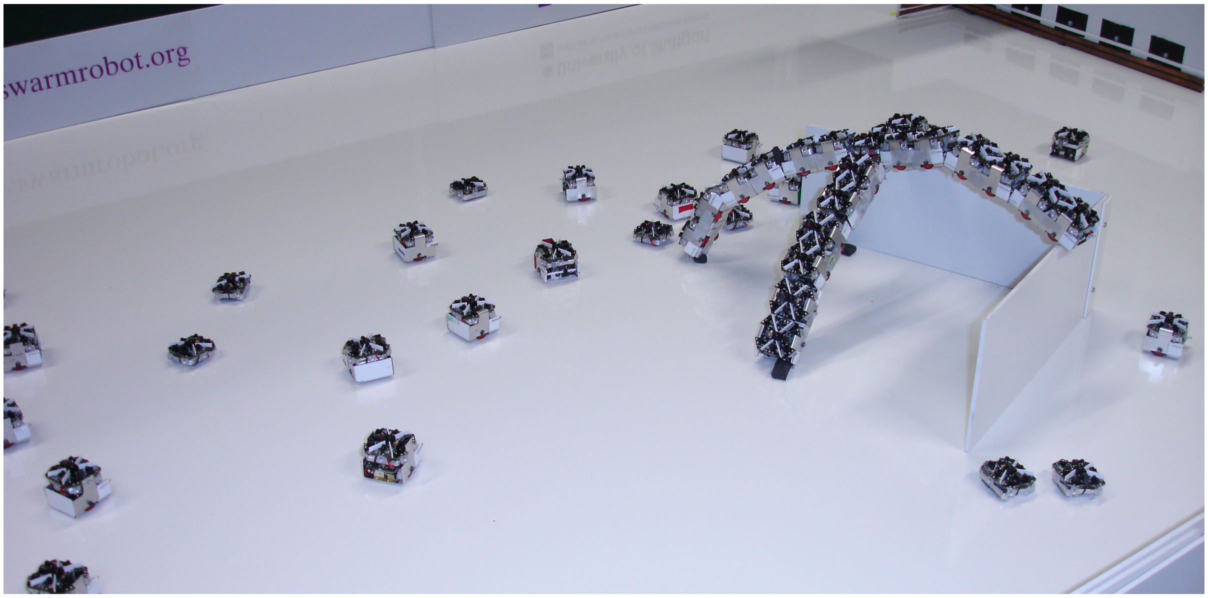


Figure 3.2: Symbrion robots [32]

The Symbrion and Replicator [55] projects provide an exciting robotics platform, that unlike Khepera and S-bots, is both open hardware and open software. It is a modular system (see Figure 3.2) with many small independent robots capable of forming larger coordinated bodies. In this way the Symbrion conglomerate robot is able to evolve morphologically - a change in the genome of the small robots can change how they assemble and therefore change the shape of the overall structure. This is similar to other modular robotic systems such as Polybot **??** and M-TRAN**??** though some are programmed by a human rather than encoded by a genome. A simulation tool can be used for Symbrion experiments to shorten the time required for complex evolution. The system has been used to perform Artificial Life experiments such as a test into the role of sexuality in robot populations [4], but is also concerned with complex locomotion behaviors [54].

Expanding upon the early work of Karl Sims [104], several more simulation-only platforms have been developed. Morphids [90], is morphological evolutionary system and has been used in testing vision systems [91]. Bongard's system [12] tested the value of behaviour chaining by evolving robots capable of both locomotion and object retrieval [10]. A number of other similar software tools are also available, for a short survey see [109]. All of these are able to evolve both the morphology and the controller of an embodied agent. By and large, any of these systems would likely be capable of demonstrating speciation as it relates to this thesis.

3.2 Speciation

The study of speciation modelling is dominated by mathematical abstractions- attempts to formalize Darwin's theories. Over 100 of these have been developed in the last century alone [123] and they continue on today. Traditional models have focused on the genetics in a sort of bottom-up approach often catagorized as Population Genetics. With a view to the 'micro' forces, these models usually ignore many macro forces in ecological details and interactions [123]. This is in contrast to a relatively new and popular modelling technique called Adaptive Dynamics [44]. This attempts to take a more balanced approach by modelling ecological forces as well, although its treatment of genetic processes may be too simplistic [123].

On the other hand, there seems to have been a recent surge of studies modelling ecological speciation which is the focus of this thesis as well. Some examples include [112] which looks at how to identify early stages of speciation from divergent natural selection. [123][63] both examine the interplay between natural and sexual selection in speciation while [79] and [100] analyze how mutation-order speciation can occur when the two populations are subjected to similar selection pressures. The topics vary greatly however a quick review of this work does point to one trend that is worth noting - a widespread use of the Biological Species Concept. Other simulations study alternative paths for speciation such as sympatry [34] [67] or ring species [2]. Like any study involving models there is some inevitable discussion about underlying assumptions and overall accuracy - for more information on this topic see [123].

A number of game simulators have also become available that allow an exploration of evolutionary development. 'Spore' [30] allows users to manually guide the development of an individual organism thereby generating unique and novel creatures. Simlife, also developed by Maxis, takes a broader approach and simulates an entire ecosystem with the user manipulating the genetic information of populations in order to effect change. Though not strictly about speciation, these simulators provide an entertaining way to investigate how evolution works.

Beyond simulations, speciation is a topic of intense study in the lab as well, with the overall number of citations increasing exponentially [105]. Much of this effort is directed at understanding the genetics behind speciation and searching for speciation genes, often in reference to a specific organism. A blending of these two approaches is the application of mathematical models to systems that are already well understood in nature.

Chapter 4

Speciation in Robotics

4.1 Why speciation should be applied to robots

As the field of evolutionary robotics continues to grow and solve increasingly complex problems, the robots and their controllers will also grow in complexity. It seems entirely plausible that a large enough population may begin to diverge through evolution. This could happen for any number of reasons: Perhaps the robots discover multiple solutions to a single problem in such a way that both genotypes are maintained in the population. Or if locality of communication creates geographically distinct populations they may naturally take different evolutionary paths. If the populations should diverge significantly enough it is possible they may begin to display characteristics and behaviors that biology would refer to as speciation. More specifically, one group of robots may become so different that they are no longer able to breed with another group. While it has not been necessary for this emerging field to create a system of categorization, biology's continued interest in speciation may point to future research in ER. Consider that "there is a general consensus among biologists that species are real and important units of biological diversity" [9].Identifying and classifying the occurance of speciation is ultimately the first step in understanding its implications. On one hand, speciation could be viewed as the systematic failure of the crossover operation - something to be avoided. In other circumstances, it may prove valuable as a self-regulating diversity preservation technique or a method for multiple solutions to develop in parallel much as they do in island models.

4.1.1 How others have not addressed this

Many works have demonstrated speciation in artificial simulations, but none in a manner that describes a robotic system. To a large extent this could be attributed to the lack of simulations performed in 3D, and only a handful having been performed in a physics based 2D setup. This is integral to robotics as a field concerned with the study of physically embodied agents. To avoid unnecessary repetition, the more in-depth discussion of the various non-embodied simulations is made in the next section (4.2).

The work performed by Earon [27] was very close to this study in terms of goals and approach. His interest lay in more complex genetic structures that included protein production and cell development. While much of the background research came from the ER community, the project looked at multi cellular organisms in 2D rather than robots.

So speciation simulations do not utilize robotics, and similarly, robotics experiments do not usually focus on speciation events. For those using a centralized evolutionary algorithm, it is less likely that speciation would even occur as the system may explicitly/implicitly select against divergent populations. However some researchers have discovered the developmental value of species in their evolutionary systems and have created it artificially [89] [23]. One of these ‘speciation’ techniques, also known as fitness sharing and diversity preservation, seeks to discourage an evolving population of robots/code from remaining too similar or converging too early [19]. By penalizing individuals for having genomes that are too similar, in the fitness function for instance, diversity in a population can be maintained. The similarity measurement is often genotype distance, but phenotype characteristics such as behavior can also be used [31]. Other researchers simply keep the two populations apart, never attempting to inter-breed them [92] [16]. This demonstrates a practical use in evolving using species, but is not technically speciation. The division between the species is simply maintained through the fitness sharing algorithm to promote diversity. However the organisms are not actually incapable of producing offspring- they are simply prevented from doing so. A demonstration that robots can naturally form into species groups has yet to be made.

4.2 Phenotype testing for speciation simulations

While the focus of this thesis is to demonstrate speciation in robotics, it also sheds light on a largely overlooked technique for performing speciation simulations. The actual process of creating a new species as it occurs in nature is long, so long that we are unable to observe it [64]. Much of our knowledge has come from the fossil record which is incomplete [17][47], and the data we've been collecting during the limited time earth has had scientists. To better understand the mechanisms of speciation we have turned much of our attention to simulations. Whether statistical or physical, these simulations can help us watch the process of creating bio-diversity.

Many of these studies have focused on demonstrating that speciation can happen in different ways. The mechanics of these events are largely ignored in favor of demonstrating the statistical plausibility of a theory - they don't test the phenotypes in a real world environment. Instead they resort to much simpler methods to groups organisms rather than according to reproductive isolation.

In [99] the experiment deals with the evolution of strings, but wrapped inside a cellular automata approach. They define two genotypes will 'belong to the same species if they differ in less than K_m genes'. This amounts to using a simple Hamming Distance measure to delineate the species. The use of a Hamming calculation further requires that the strings all be of the same length - The genotypes cannot change size.

In [43] a numerical simulation again uses the variable 'K' to define an arbitrary phylogenetic line. The authors note at the end the importance this K factor has both on the number of species and the rate at which species turnover, and show it to be $\frac{1}{\sqrt{k}}$.

[71] has a slightly different approach, although the experiment is never actually performed. Rather than calculating the difference across the entire genome, there is a specific area (or variable) that is checked. If the variables between two agents do not meet an arbitrary similarity measure, then the agents are defined as different species.

The treatment of speciation as a discrete process with clearly defined boundaries between the species is erroneous. This is especially true early in the speciation event when differentiating between species is much more difficult [118]. In more formal terms a species is a 'fuzzy set' [1]. In order to accurately model a species this should be considered. These algorithmic shortcuts are unfortunately necessary when dealing with certain isolating barriers. For instance, gene incompatibility leading to hybrid sterility is not even fully understood and is likely to prove intractable once it is. Attempting to model this is unfeasible. Other elements of the speciation process are not so difficult to work with and present an exciting opportunity for study. Inviability of hybrids from low-fitness, for instance, can be modeled and simulated. A phenotype could be tested in a physical environment to determine its fitness. This would not have been possible in the aforementioned studies, as the genotypes did not encode to a phenotype. Some researchers have taken the next step and designed an evolutionary system that considers both genotype and phenotype.

For instance [21] is an ecological simulation of the breeding patterns of a specific fish. Although not a genotype in the strictest sense, each individual fish has a number of variables (such as color) which are passed on during breeding. Breeding preferences and patterns are then a function of these characteristics. So the simulation includes phenotypic parameters in the evolution, but the phenotypes are never directly modeled. The fish are not actually simulated in an environment to be tested for viability, their behaviors are determined by a statistical model.

In another experiment [27] created a complex Artificial Life simulation in which organisms begin as cells and develop into multi-cellular creatures. The evolution was genomic in nature with the organisms as the phenotype expressions. Despite the possibility for phenotype viability testing, this experiment also used the method of calculating a distance measure.

[67] used the Framsticks 3D simulation platform in an attempt to create a speciation event, but also did not test the phenotypes for viability. Like [71] they differentiated between individuals based on a single gene/variable. Unfortunately the experiment was not able to successfully create two divergent species.

An opportunity presents itself to demonstrate that viability testing can be explicitly modeled in a three dimensional environment in a speciation simulation for the first time.

Chapter 5

Implementation

5.1 The Basic Idea

Imagine two large flat areas, each filled with a population of 60 robots. These robots are all homogeneous - they have all directly descended from a single genome and differ by only 1-2%. The two areas are completely separated; robots cannot move from one area to the other.

This is the basic setup for the thesis experiments and it is modeled using the 'Framsticks' platform. Framsticks is a 3D simulation tool capable of evolving both 'mechanical structures' and 'control systems' [57]. Each robot in the simulation has a genotype, which encodes for the neural network and the physical body. The system is a powerful tool with a C-based language that gives the programmer control over almost every aspect of the environment and evolution. Many of the tools and libraries were not used in this study, but there were several features which made it ideal for a speciation experiment.

The most critical parameter is the physical expression of a phenotype capable of interacting with its environment. The Framsticks system allows the robot to actually be tested for viability and fitness rather than estimating it based on the genome. As previously mentioned, the ability to evolve both the morphology and the controller simultaneously may also provide a more 'open' evolutionary system.

For the same purpose, it is also valuable for the system to have the capacity to accommodate functionally neutral changes to the genotype. For instance, it is common for the more complex genotypes to have one or more neurons that are not connected to a robot's central neural network. These neurons may play no role in the robot's behavior, so how could they be useful? In a later generation there may be a slight mutation or crossover that joins the neuron to another and instantly a complete node(s) has been added to the network. In this way, functional additions can be made to the robot that may not have been possible through a single incremental mutation.

5.2 The Framsticks environment

A more complete overview of the Framsticks system is available[57], so only a brief description of the relevant details will be made. In simulation, the body of a robot is composed of rigid posts or sticks that are joined at their ends. Mechanical actuators positioned at the joints allow the robot to move. Just as other simulators do not explicitly model the engines that drive the wheels of their robots, Framsticks does not explicitly model the force-generating mechanism of its robots. It does, however, enforce laws of physics on all movement as described by Figure 5.1. The mechanical devices used are ones common to real robotics systems today - a rotator (such as a motor) [Figure 5.2a] and bender (such as a hydraulic arm) [Figure 5.2b].

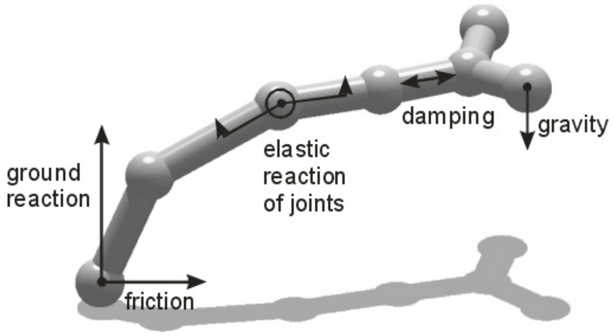
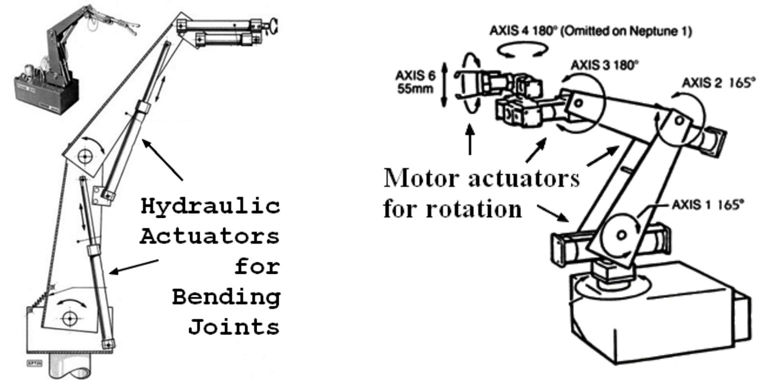


Figure 5.1: Framsticks Robot Body [57]

These actuators are controlled by a neural network comprised of both state neurons and sensory devices. Connections between neurons are weighted but not uniquely handled, so all neural inputs are aggregated weighted sums. The vast majority of the state neurons used by the evolved robots are Sigmoid Neurons, common to many ER



(a) Hydraulic Actuator (b) Rotation Actuator

Figure 5.2: Actuators on Robotic Devices [73]

experiments [35][46][91][86]. The other type of state neuron used is a Constant Neuron, which always has a normalized output value of 1. The Framsticks system provides a small library of different sensors that can be used for input into the neural network. For instance, there is a sensor that can determine the distance (but not direction) to the nearest target item. While it was likely that this would improve the functionality of the robot, it was intentionally left out to help ensure the simulation remained as realistic as possible. Designing a sensor to perform this task accurately in the real world would be both costly and difficult.

For this work only two of the available sensory devices are used: gyroscopic and contact. Both of these types of sensors are widely available and have been shown to work on a robotics platform [110]. The gyroscopic sensor is attached to one of the robot's structural posts; its output is defined by the orientation of the post relative to gravity. In the horizontal position, the sensor output is 0, but tilting the post vertically will result in output values reaching +/-1. The touch sensor is described best as a 'whisker' sensor, rather than as a button which may only have a discrete output. Non-activated output of this sensor is -1, but at very close distances to another object the output approaches 0. If force continues to be exerted once the sensor is already in physical contact with the ground/object, the sensor output becomes positive up to a value of 1. All inputs to the robot system include a measure of Gaussian noise (1% standard deviation) to help create a more realistic environment as demonstrated by [49]s.

5.3 The Genome

The entire robot including structure, mechanics, and neural network are all encoded by a genome. Framsticks provides a number of different encoding systems for genomes, a complete summary is available on the development website [57]. This thesis uses ‘f1’ encoding which has been shown to be both robust yet relatively easy to understand. It is a high-level recursive language that uses alphanumeric strings to encode the data. Additionally, experiments have shown f1 to produce better results than the low-level encoding [59].

The basic structure of a gene in f1 is ‘Modifiers Structure [neuron info]’. The symbols used for structural elements are the letter X, brackets and commas. An X defines a structural element and a bracket defines a branching point. Within the brackets the angle of branches is divided by commas. Figure 5.3 describes a simple exercise in building the body of a robot using branching.

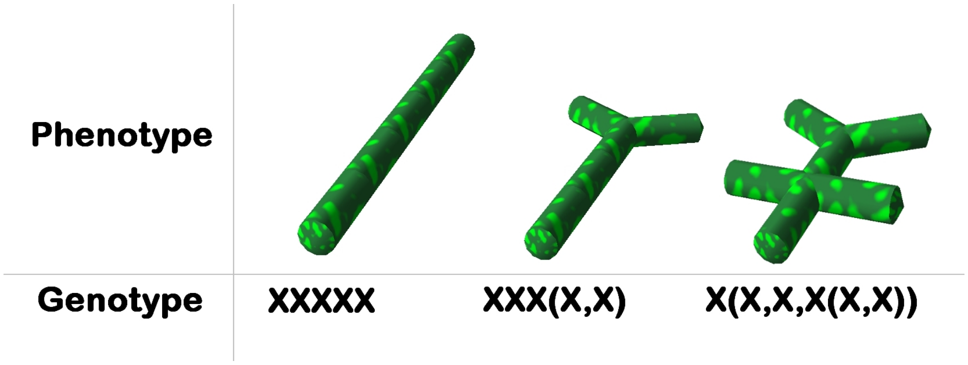


Figure 5.3: Body Genetics

Modifiers are special characters that affect various physical properties of a robot.

R	Rotates parts axially
Q	Twists the parts relative to each other
C	Adjusts curvature by adding a slight angle between parts
W	Affects buoyancy (discussed in Section 6.3.1)
L	Shortens or lengthens a structural element
M	Changes the amount of force exerted by corresponding actuators
F	Increases or decreases the friction coefficient of the part (e.g. sticky vs slippery)

Table 5.1: Physical Modifiers

They precede their structural elements having a lessening effect on each subsequent X. Figure 5.4 demonstrates the use of 'R', the rotation modifier. The first R rotates the terminal section of the robot 45 degrees and the effect is cumulative - two Rs rotate it 90 degrees. This work makes use of the seven modifiers shown in Table 5.1.

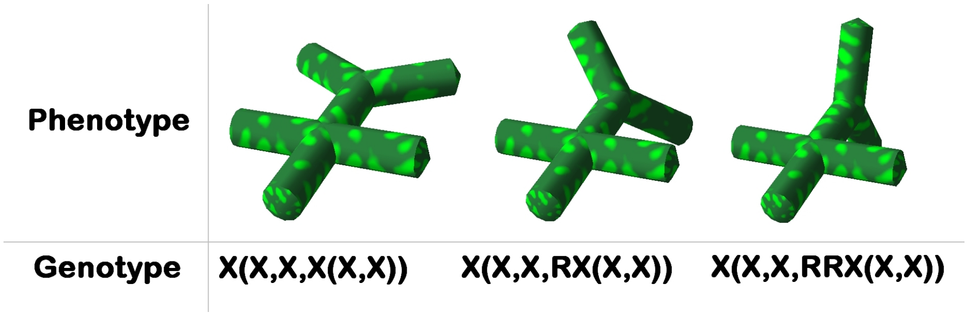


Figure 5.4: Genetic Modifiers

5.4 Neurons

The final element in the gene is the neuron information associated with the structural X, proceeding it in [ ] brackets. The general format is *[NeuronType, PropertyandInputList]* although in the case of a sigmoid neuron the *NeuronType* is absent. The remainder takes the form of *PropertyType:Value* or *NeuralInput:Weight*. There can be multiple properties and inputs separated by commas, as well as multiple neurons for each stick in the form *X[...][...][...]*.

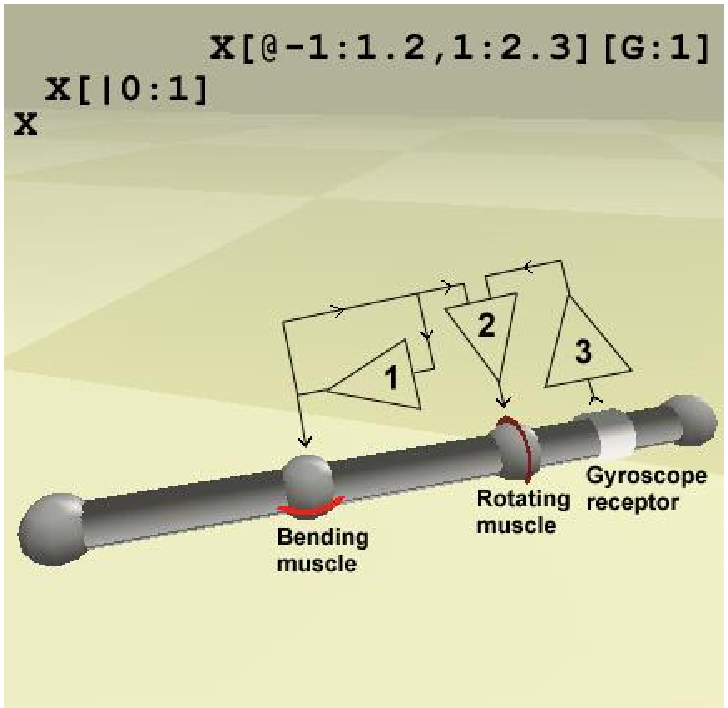
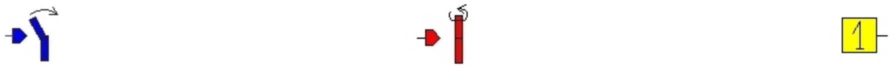


Figure 5.5: Example Neuron Map

All inputs to a neuron are denoted as relative locations with the following syntax *[location: weight]*. For example, *X[T][N,1: 0.5][G]* defines a stick with 1 touch sensor(T), 1 neuron (N) and 1 gyro sensor(G). In this case the neuron takes its input from the location 1 (meaning 1 to the right) which is the Gyro sensor, with a weight of 0.5. If, however, the genome was changed to *X[T][-1 : 0.5][G]* so the input location was -1 instead of 1, it would take its input from location -1 relative to itself which is the touch sensor. A slightly more involved example (courtesy of Framsticks) is Figure 5.5. In this case the first neuron *[|0:1]* signals the bend actuator and obtains input from location 0 (looped to itself). The second neuron *[@-1:1.2,1:2.3]* signals the rotation actuator. It takes input from locations -1 and 1, and neurons 1 and 3 respectively. The last neuron *[G:1]* takes input from the gyroscopic sensor with a weight of 1.

(a) *N* - Sigmoid Neuron (b) *G* - Gyro Sensor (c) *T*- Touch Sensor



(d) | - Bending Actuator (e) @ - Rotation Actuator (f) * - Constant Neuron

Figure 5.6: Neuron and Sensor Types

5.5 Crossover operation

An example of the crossover operation is shown in Figure 5.7, which describes the method of joining the genetic information from two different robots. Since an ‘X’ in the string denotes a structural element of the phenotype and is common to all genotypes, this makes it a reasonable position to splice. First, the halfway point for the string is located (shown as the arrow in step 2). The closest ‘X’ to the right of this mark is selected as the division point (step 3). The same procedure is followed for dividing the second genome (step4&5). The first half of Genome 1, and the second half of Genome 2 are combined to create the new genome. The syntax of the genetic encoding makes it possible for ‘invalid’ genotypes to be created accidentally. If the gene error is minor, the system can ignore it - essentially inhibiting that gene from expression. In Framsticks, this generally means that a defective (syntactically incorrect) neuron, joint or limb is ignored in the phenotype. It is possible for the error to damage the gene’s syntactic structure so badly that the expression system cannot adjust. In effect, the genotype is so flawed that it simply cannot be made. This is theoretically similar to the phenomenon of pre-zygotic isolation in biological speciation events - when a hybrid organism is not born due to genetic incompatibilities.

Figure 5.7: Crossover Operation

5.6 Mutation operation

Mutations are also an important part of the robotic evolution as they provide ‘fresh’ genetic material. Some experiments in ER use completely random mutation[72][35], but this would be impractical given the syntactic structure and would likely produce even more invalid genotypes. Instead the mutation operation in Framsticks is highly customizable both for morphology and for the neural network. Figure 5.8 displays those properties that were mutated in the majority of tests as well as their corresponding probabilities. In this setup a single connection weight is changed in the neural network 100% of the time, but a stick will be removed only 5% of the time.

Many have suggested that morphological evolution in conjunction with neuronal evolution helps us approach open-ended evolution. However given the current state of technology, the actual application of that theory outside of simulation is largely impractical. It seems the vast majority of ER experiments recognize this implicitly in their choice of

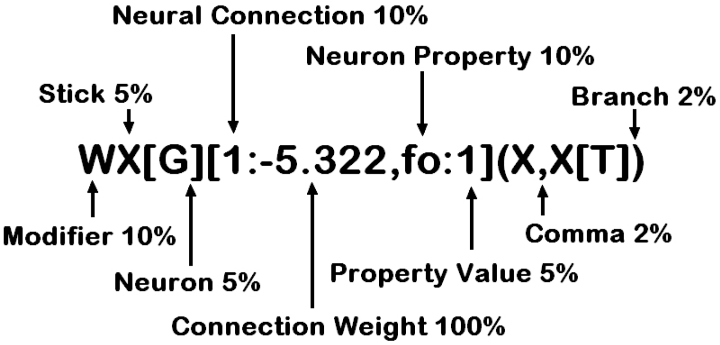


Figure 5.8: Mutation Operation

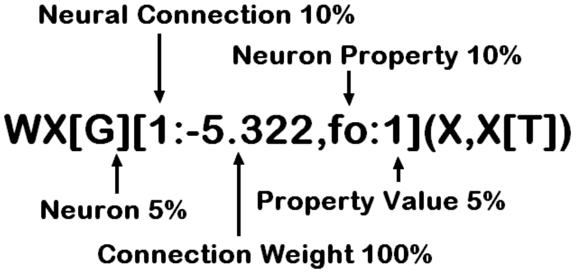


Figure 5.9: Non-Morphological Mutation Operation

neural network evolution only. So while the choice of Framsticks was made specifically because of its ability to perform morphological evolution, a series of experiments was also performed to attempt speciation through controller evolution alone. To achieve this all aspects of the robot were held constant except the neurons and their weights. The structure and mechanics of the very first robot was identical to that of the very last as well as the hybrids. No changes to the operation of the crossover function had to be made, only the mutation operator was adjusted using the parameters shown in Figure 5.9. The initial population also had to be morphologically homogenous.

5.7 Comparison Tools

On a fundamental level speciation describes a stable clustering trend, a convergence towards a local optima. Identifying low hybrid fitness addresses the stability but tells us little about the clustering itself. To accomplish this requires some quantification of the robot itself independent of its performance. As speciation is an inherently relativistic concept [22], using a relative metric for this only makes sense. The robots can be considered in two different dimensions, their genotype and their phenotype. For genotypic comparison the Levenshtein Distance calculation is used, which when applied against two genomes gives the number of single character changes required to turn one into the other. Some of the properties used in these experiments (see Figure5.8 on page 46) use multi-character mutations, so this gives a more approximate value to the evolutionary distance between genotypes rather than an exact one. It is also closely related to the Hamming distance calculation used in many other simulations, which performs the same function but for fixed length genomes.

Not surprisingly defining a measure of the difference between phenotypes is much more complicated, this thesis uses the method described by [58]. In this, the properties of the phenotype are divided into more manageable groups. First are the physical characteristics, the structural elements and the joints. This includes most of the modifiers in table 5.1 and more importantly the overall map of the body as it relates to each part's degree [1]. Once the two robots have been compared and their parts matched for similarities, the associated neural elements are compared. This is done first based on the number of neurons for the matched parts, and then for their parameters (connections, sensors, actuators . . . etc). The final summation value is an estimation of the dissimilarity between the robots.

5.8 Robot Movement in Framsticks

The locomotion of robots in Framsticks could be described as 'limb-driven' rather than 'wheel driven' as it is in many other ER experiments. Similar to the foundational work

---

[1] 'degree' as it is defined in graph theory: The number of connections at any given vertex

of Sims[104] and Lipson[66] these robots tend to evolve crawling, hopping or walking as their primary methods of movement. This is common in other ER simulations that allow complete morphological evolution [90][12]. Figure 5.10 shows an example of an evolved robot moving across the ground. It first reaches forward with two sections of its body and then pulls its entire frame along the surface. Repeating this action allows this robot to effectively transverse across the simulation environment.

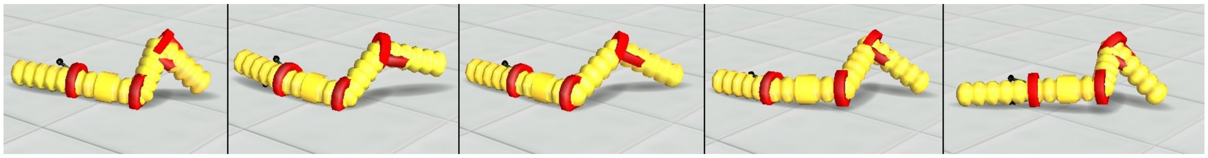
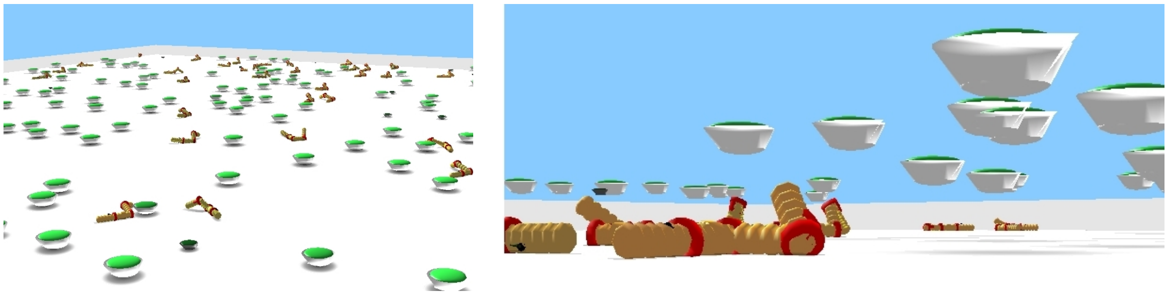


Figure 5.10: Basic Robot Movement in Framsticks

5.9 Experimental setup

As mentioned above, the experiments model two isolated populations of robots. While the initial populations of robots are identical in the two experiments, the environments are slightly different. Both possess the exact same rules such as physics, and both are filled with ‘target items’- but the placement of these items differs. Target items [figure] are discs that are scattered randomly throughout the environment. In the case of environment 1, these targets lie on the ground [Figure 5.11a]. In environment 2, called the ‘Low Hanging Fruit’ (LHF) environment, the targets hang in the air, just above the ground [Figure 5.11b]. In essence, environment 1 places the targets randomly in two-dimensions, while the LHF places them randomly in three dimensions.

These target items define the performance, or fitness, of the robot. When a robot touches a target item it receives points, each target only has a certain number of points to give after which it disappears. A new target will appear randomly in the environment to replace it.



(a) Environment #1 (b) Environment #2 (LHF)

Figure 5.11: Experimental Setup

These two environments were designed to provide nearly identical scenarios for robot development while also incorporating a controlled differentiation. The relative complexity of the Framsticks genetic system should also allow for multiple evolutionary paths to be taken in response to varying environmental cues. Coupled, these features create an excellent platform from which to experiment with, and hopefully generate robotic speciation.

5.10 Breeding process

Once a specified number of steps/cycles (usually 10000) have passed the robot enters a breeding mode. The robot looks to a random[2] member of its population and receives its fitness value (communication protocols are beyond the scope of this work). The fitness is calculated as points/lifetime - this is used rather than just points, accounting for the fact that the second robot will not have been alive as long as the breeding robot. Lifetime is calculated as the number of cycles the robot has been functioning for. The selection of what genotype will be used for the next 10000 cycles is determined as follows:

If the fitness of the second robot is lower than that of its own, the first robot will reset (lifetime and points =0) and continue using its current genome. If the second robot

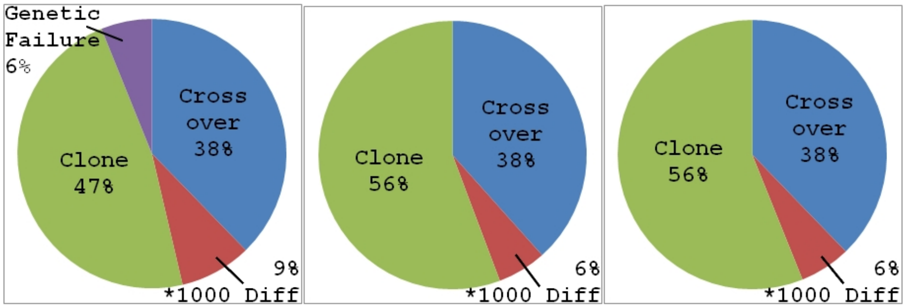
---

[2] Through selecting a random robot the communication is simplified to ignore locality, thereby controlling for the effects of dispersal as described by Payne [87]

Algorithm 1 Selecting which Genotypes will be Used

```
if $Robot1.fitness \geq Robot2.fitness$ then
    $ResetAndContinue(Robot1)$
else
    if $(Robot2.fitness * 1000) > Robot1.fitness$ then
        $Robot1.genotype \leftarrow Robot2.genotype$
    else
        $Robot1.genotype \leftarrow Crossover(Robot1.genotype, Robot2.genotype)$
    end if
end if
```

has a higher fitness value, the first robot will perform a crossover operation using both robots' genomes. If the product of the crossover function does not have any genetic errors it will also be mutated. A substantial gap in the fitness of the two robots could be an indication that one of the robots is low performing or the other is high performing (or both). In this case it would be beneficial for the first robot to duplicate the genome of the second robot rather than performing a crossover. This action is performed if the fitness values differ by a factor of 1000 or more (see Figure5.12 for rates). Allowing the robots determine their own breeding rather than determination through an omnipotent central system aims to simulate embodied evolution - a process more similar to evolution in nature.



(a) 1 Million Cycles (b) 25 Million Cycles (c) 50 Million Cycles

Figure 5.12: Rates of Crossover and Cloning at Different Times in Sample Test

5.11 Test procedure

The two populations both run for 50 million cycles, during which they make some evolutionary progress in adapting to their environments. At every millionth step a sample of the genepool is taken. Testing, for the purpose of data collection, is performed independently of the evolution and has no effect on the robot's adaption. Each of the samples contains the complete genome for all 60 robots in each population. Every one of these robots is individually tested for fitness in their native environment for 7500 cycles. With the timing held constant, fitness is taken simply as the number of points the robot acquired. This test is performed twice for each robot and the average of the fitness values is saved. This amounts to

$$2populations \times 50\frac{samples}{population} \times 60\frac{robots}{sample} \times 2\frac{tests}{robot} = 12000tests$$

The next step is to determine how effective hybrids robots generated from the samples would be. A random genome is selected from the first population, and another from the second. The same crossover function that is used in the evolutionary process is used to generate an offspring genome from these parent robots. This is repeated until up to 60 valid hybrid robots have been generated for each of the 50 sets of samples. Again, each of these hybrids are tested twice for fitness. The result is data at regular intervals during the evolution showing the relative viability of hybrids generated as the two populations diverge.

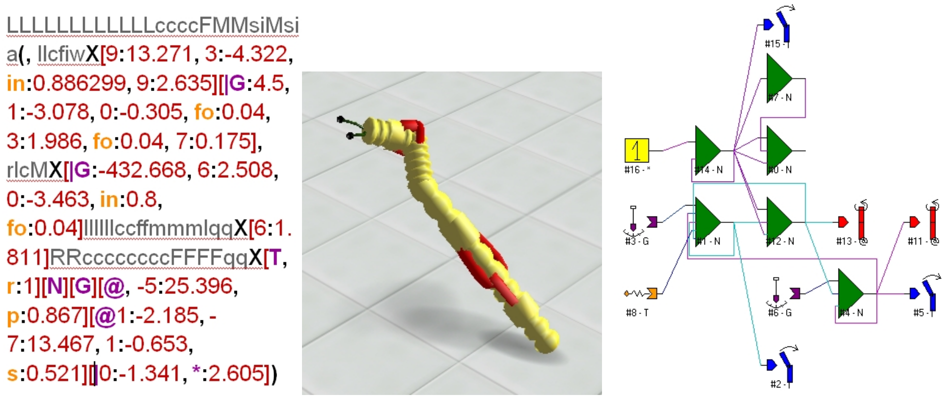
The following section will cover a number of individual tests performed using these systems. Additional parameter information and experimental setup details are available in the Appendices A & B.

Chapter 6

Results

6.1 The Classical Scenario

The first results presented are those that were tested in an Allopatric setting, the Classical Scenario as described in the previous section. The intial population is cloned from a robot called M5, which is a functioning, pre-evolved robot specifically created for these tests. Its body is composed of 4 sections, with 3 bend actuators and 2 rotational actuators as can be seen in the neural network (fig 6.1c) and genotype (fig 6.1a). The primary method of movement is best described as jumping; it is a combination of vertical and horizontal travel. It first compresses its joints and then releases them simultaneously, springing up and forward. As figure 6.1b shows, the robot lifts itself completely off the ground. M5 provides an excellent genome from which to generate a population for the allopatric test because it is already capable of acquiring both ground and LHF targets. Without an ability to achieve at least modest fitness, the robots are less likely to find a successful evolutionary path (bootstrap problem).



```
LLLLLLLLLLLLccccFMMsiMsi
a(, llcfiwX[9:13.271, 3:-4.322,
in:0.886299, 9:2.635][|G:4.5,
1:-3.078, 0:-0.305, fo:0.04,
3:1.986, fo:0.04, 7:0.175],
rlcMX[|G:-432.668, 6:2.508,
0:-3.463, in:0.8,
fo:0.04]lllllllccffmmmlqqX[6:1.
811]RRccccccccFFFFqqX[T,
r:1][N][G][@, -5:25.396,
p:0.867][@1:-2.185, -
7:13.467, 1:-0.653,
s:0.521][|0:-1.341, *:2.605])
```

(a) M5 Genotype (b) M5 Phenotype (c) M5 Neural Network

6.1.1 Course of Evolution in Ground Target Environment

The original M5 genotype was not particularly well suited to acquiring targets scattered on the ground, considering its tendency towards vertical travel (see Figure 6.1b). However the basic means of locomotion was not entirely ineffective, and after 10 million cycles, the robots have started to move along the ground rather than jump 6.1. They continue along this evolutionary path and optimize their motion for fast and efficient horizontal movement 6.2. The evolutionary progress is evidenced by the overall increase in fitness of the population, as shown in Figure 6.3. Recall fitness is defined in Section 5.11 as the number of point each robot acquires during testing.

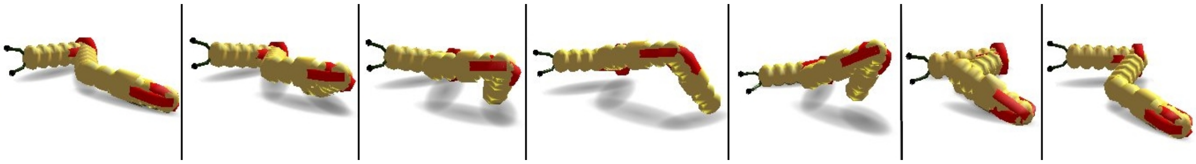


Figure 6.1: Locomotion in Ground Target Environment at 10 Million Cycles

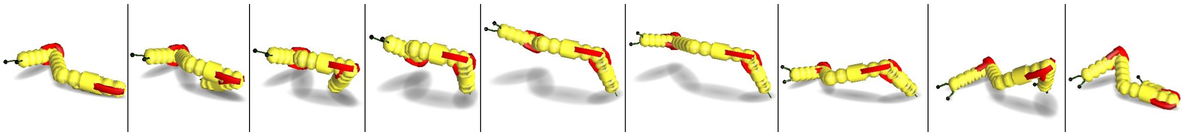


Figure 6.2: Locomotion in Ground Target Environment at 50 Million Cycles

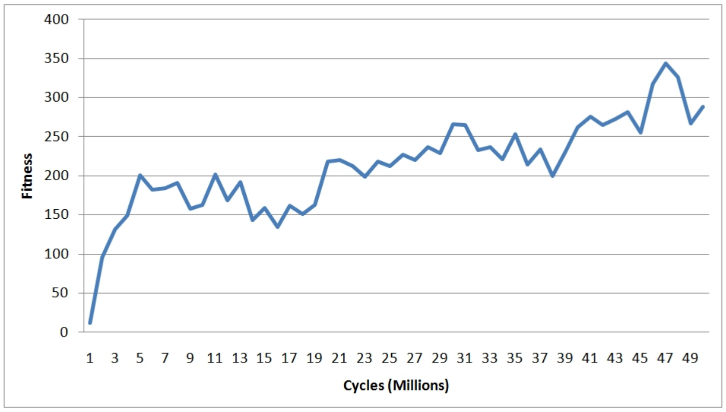


Figure 6.3: Fitness of M5 during Evolution in Ground Target Environment

6.1.2 Course of Evolution in LHF environment

The M5 robot requires little structural change in order to adapt to the hanging target items of the LHF environment and still looks very similar even after 50 million cycles of evolution (see Figure 6.4). However the original jumping behaviour, while effective, required some optimization. After the experiment is complete, the robots are able to collect approximately 1.5 times as many target items as the initial M5 phenotype as described by Figure 6.5.

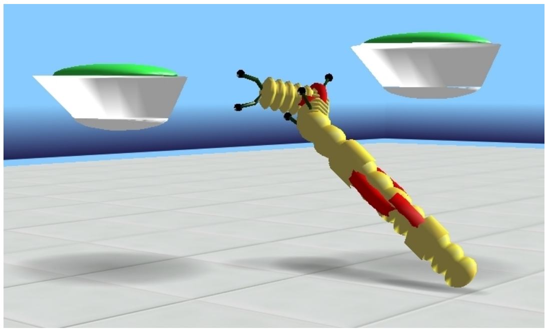


Figure 6.4: Evolved M5 Robot Jumping Towards LHF Target

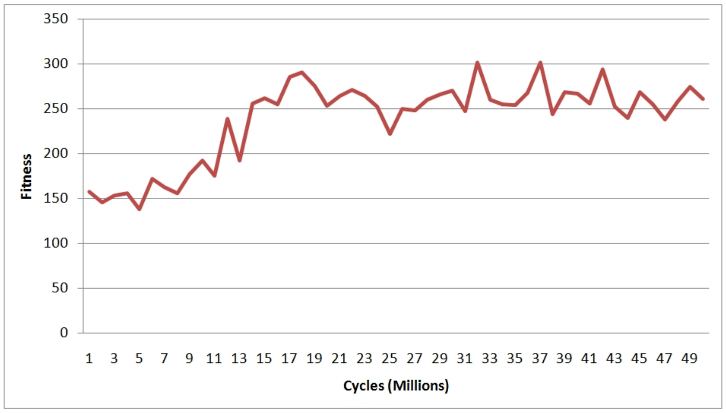


Figure 6.5: Fitness of M5 during Evolution in LHF Environment

6.1.3 Hybridization

Hybrid robots are generated using the crossover operation described in Section 5.5 using a parent genotype from both populations. Early in the experiment, before the populations diverge significantly, the hybrid robots perform reasonably well. However as the genotypes of the robots in the two environments become more dissimilar, the hybrid fitness levels fall away while the parent populations remain viable (see Figure 6.6).

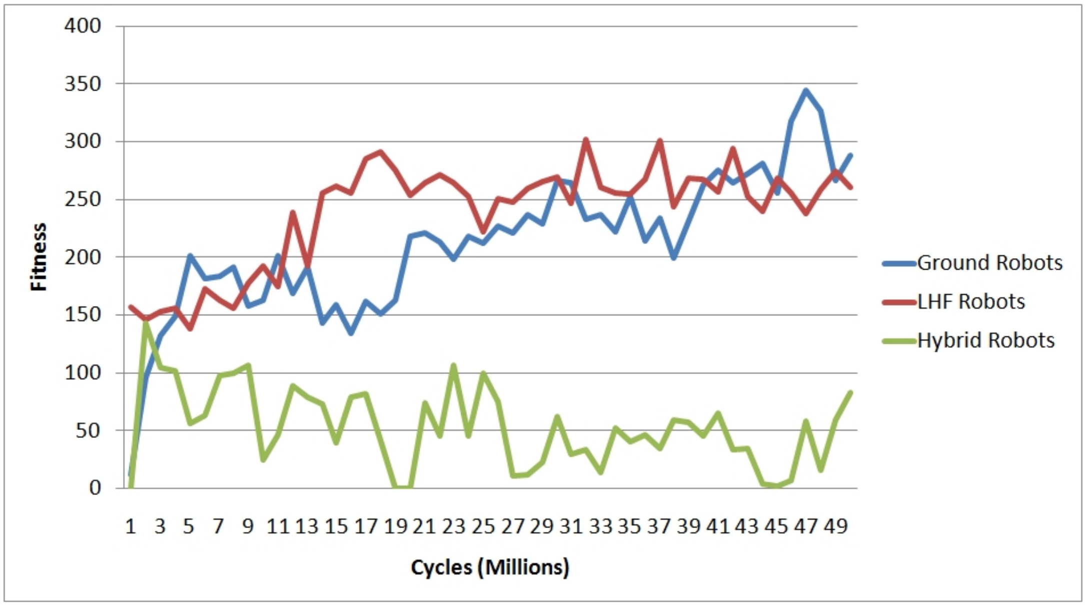


Figure 6.6: Hybrid Fitness During Evolution of M5 Robots

The hybrid may still have much of the same physical structure as its parents (see Figure 6.7), but it appears to operate without intention. The movements of the hybrids do not display the coordinated patterns required for locomotion. Any physical displacement that is attained lacks consistent direction - in general the hybrid could be described as ‘flailing’. This is likely the result of divergent evolution in the robot’s actuators and in its neural network (See Figure 6.8). The neural networks are presented here to display their overall topography, rather than for a detailed analysis of their operation. For simplicity, recognition of changes to neural network function are implicit in the recognition of changes in robot behaviour. The results demonstrate the emergence of reproductive isolation between the two populations as a result of their independent evolutionary paths. In losing the ability to produce viable offspring between the groups, they satisfy the definition of a speciation event.

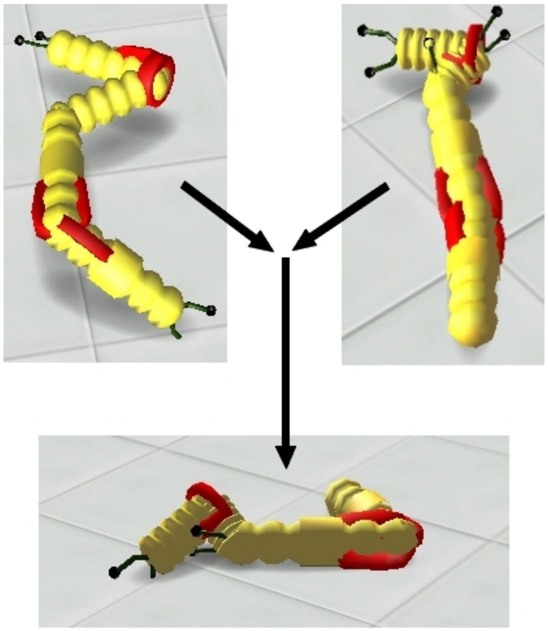
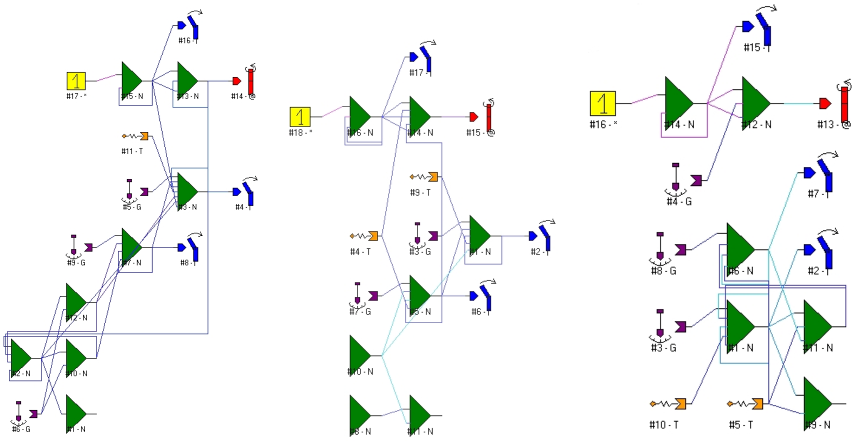


Figure 6.7: Hybrid Generated from Evolved M5 Robots

Note that this does not suggest that we cannot prevent speciation from occurring in these artificial systems, in fact we have the same power to disrupt normal evolution in the real world (which we do regularly through animal husbandry). Humans have created a number of creatures that nature surely would have been made extinct if it werent for their perceived value as pets or livestock. Instead this is a demonstration that speciation can happen naturally in robotics through many of the same mechanisms as it does in biology.

6.2 Morphological Testing

The average person, if asked to explain why dogs and lake trout are different species, is unlikely to refer to reproductive isolation. Instead they may point out various physiological or morphological differences, such as lungs vs gills, fur vs scales, etc. This more intuitive approach makes use of the Morphological Species Concept(MSC), which is similar to the taxonomic systems used in traditional cultures [39]. Other methods for defining



(a) Neural Net from Ground Population (b) Neural Net from LHF (c) Neural Net of Hybrid

Figure 6.8: Low Functioning Hybrid Neural Network

species also use some measure of phenotypic similarity such as the Phenetic and some forms of the Phylogenetic Species Concepts [94]. So while reproductive isolation, and therefore speciation within the context of the BSC, was demonstrated in the last section, further genotype/phenotype analysis may serve to reinforce and clarify the results.

As described in section 5.7, genetic differences will be measured as Levenshtein Distance which estimates the number of mutations required to change one genotype into another. The difference between phenotypes is calculated as a combination of the difference between body parts, neural networks, actuators and sensors, as also described in section 5.7. Morphological analysis need not be so formulaic though, a simpler and more subjective approach may be used as well.

Using these techniques to individually analyze each genome from the experiment discussed in the previous section yields interesting results. Early in the evolution, the two populations remain largely the same and occupy the same genotype-phenotype distance space (see Figure 6.9). However as the experiment progresses, and evolutionary pressures direct the two populations in different directions, a gap can be seen forming in Figure 6.10a. The relative differences increase over time and the two populations continue to move further away from each other. Just as important, though, is that while the populations are moving apart, the robots within those groups remain cohesive in both genotype and phenotype space. This means the robots are genetically and physically similar to the members of their own population but different from members of the other population, thereby satisfying the basic definitions of other species concepts as well.

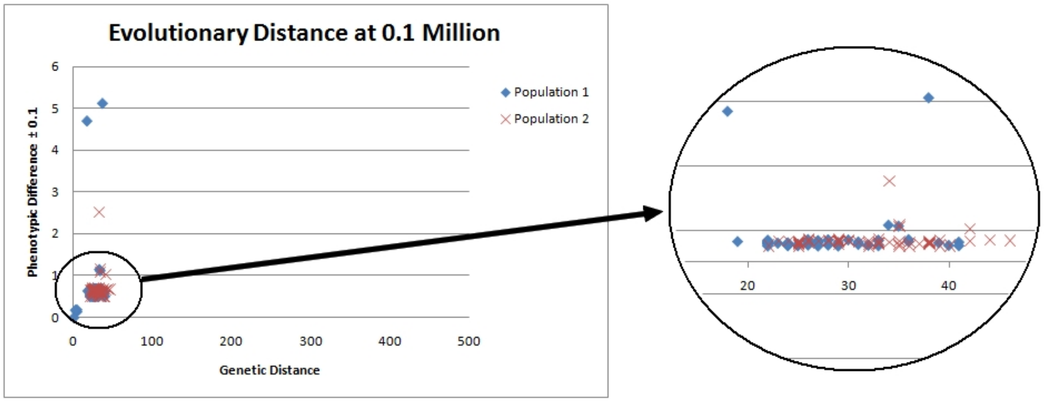
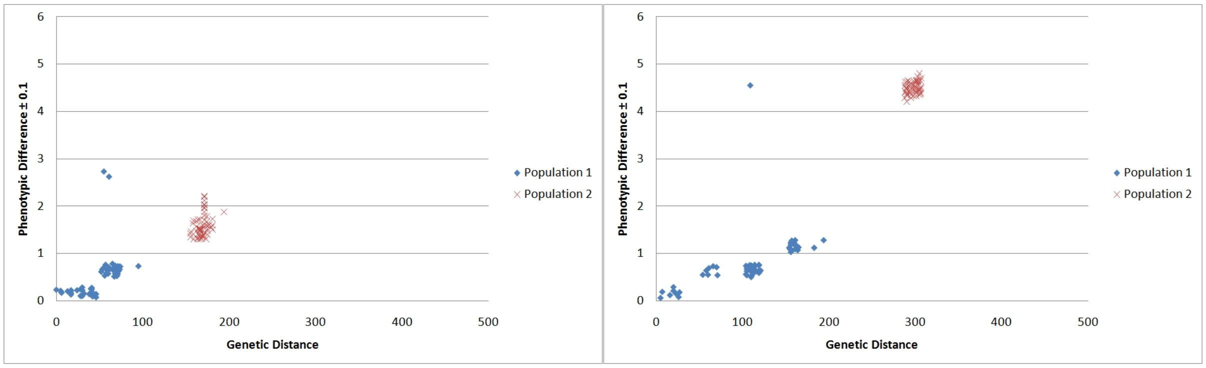


Figure 6.9: Genotype/Phenotype Distance before Evolutionary Divergence

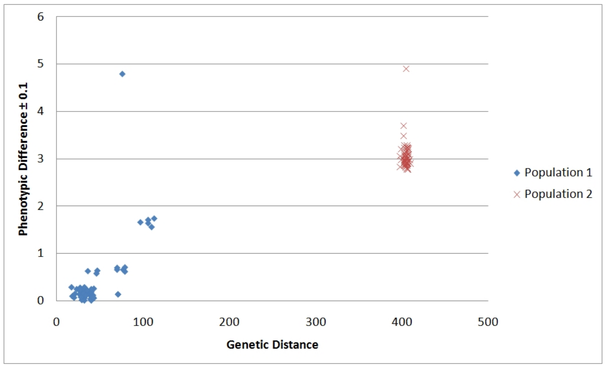
6.3 Environmental Impact on Speciation Rate

Ecological speciation is dependent on environmental differences to promote genetic divergence. If the two environments in question are not sufficiently different, speciation may not take place. Theory states that evolutionary divergence is linked to ecological differentiation [80][96]. This relationship can be tested in the robotic simulation as well by increasing the environmental differences to see if that promotes speciation as it does in biology [40].

The results demonstrated so far have used the original environments Ground Targets and LHF, as described in section 5.9. These differ only in the height of the target items; all other parameters are kept constant. To test if robotic speciation demonstrates a similar reaction to an increase in environmental differences requires a third simulation



(a) 1 Million Cycles (b) 25 Million Cycles



(c) 50 Million Cycles

Figure 6.10: Genotype/Phenotype Distance Throughout Evolution

setup. One might imagine these first two as air environments, though air is not explicitly modeled.

6.3.1 The Water Environment

Framsticks also has the ability to simulate a fluid / water environment for the robots to evolve in such as the one in Figure 6.11. The transition from ‘dry’ land to an environment that includes a surrounding fluid comprises changes that go beyond merely altering the positions of the targets. First is the way the robot can move about while acquiring targets. It allows locomotion to be generated via fluid propulsion (such as a kick would provide) as well as by friction. It also incorporates buoyancy into the physics engine; each stick element has an evolvable parameter for weight that determines how much it floats. This, as [102] points out, also requires less balance on the part of the robot.

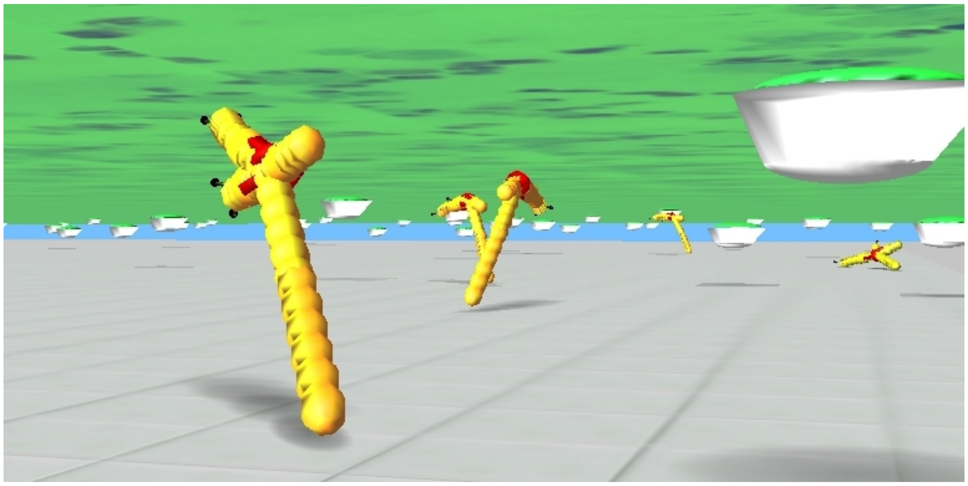
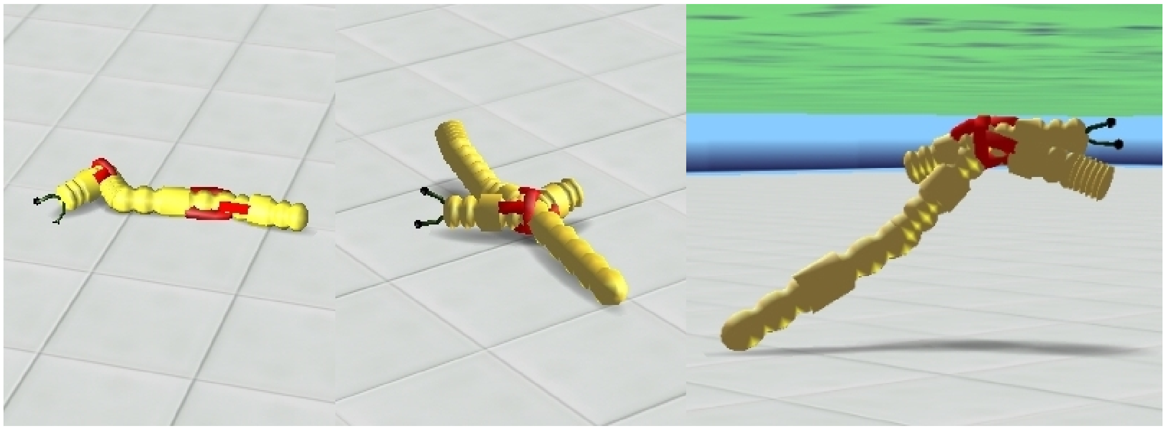


Figure 6.11: Water Environment

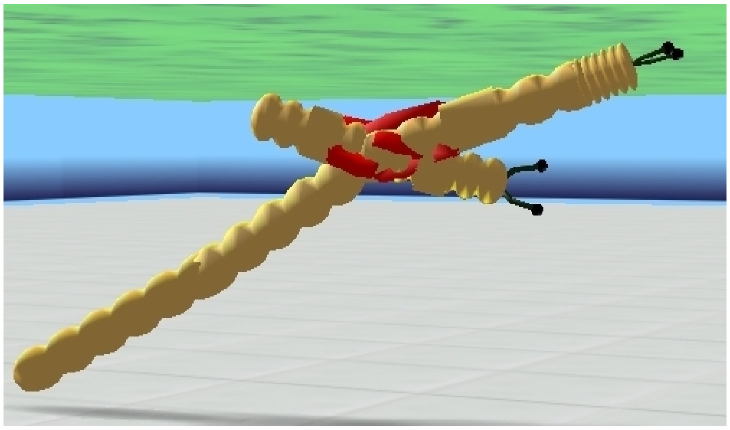
6.3.2 Evolution in Water

The same initial population as the tests in section 6.1.2 are used in this water evolution test. The original phenotype, while quite functional on land, is almost completely immobile in the water. After one million cycles, the structure has already changed dramatically (see Figure 6.12b). The robot no longer has a long straight body, but has taken a semi-quadruped shape with four limbs articulated from a central point. It still lacks buoyancy and moves along the ground in a similar fashion to a starfish, as described in Figure 6.12b. By five million cycles, the robot has become more neutrally buoyant which generally helps water-borne organisms move more effectively. It has kept the central articulation and has evolved to the general shape that it will keep for the remainder of the simulation.

Locomotion is primarily generated through a vertical paddling motion of the intermediate-length limb. This propels the robot through the water in a forward motion. The small limbs oscillate up and down with the robots movement through the water, in a similar fashion to the pectoral fins of a fish. It is likely that they provide some of the same lateral



(a) Original M5 (b) 1 Million Cycles (c) 5 Million Cycles

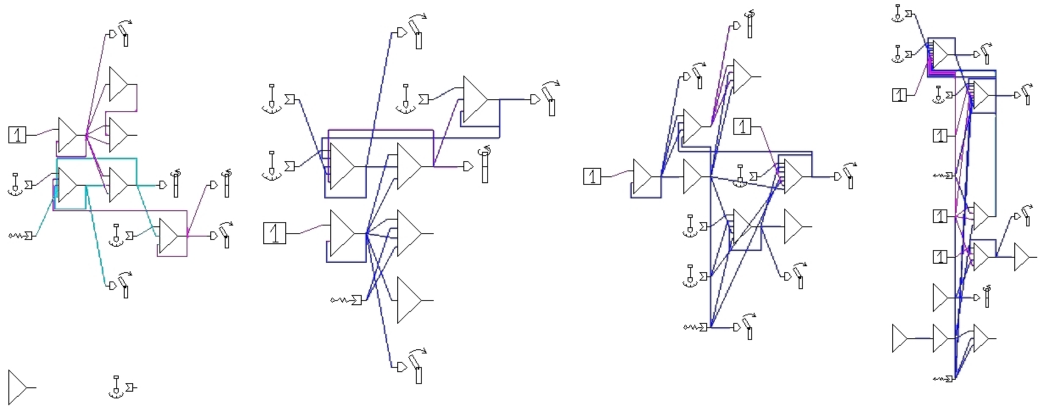


(d) 50 Million Cycles

Figure 6.12: M5 Evolution in Fluid Environment

stability by preventing undesirable rotation of the body. Finally the long limb appears to perform three different functions. First it drags along the bottom optimizing the collection of target items as it passes. It also assists in motion by re-stabilizing the robot after each 'kick', and by using some friction to create an additional pulling movement.

This adaptation towards swimming contrasts sharply with the land-based jumping techniques in Figures 6.2 and 6.4. In a comparison between the final products of both evolutionary paths, it is clear that the behaviours, morphology, and neural network have all become substantially different. From an MSC standpoint, the robots do appear to have diverged into different species.



(a) Original M5 (b) 1 Million Cycles (c) 5 Million Cycles (d) 50 Million Cycles

Figure 6.13: Neural Network Evolution in Fluid Environment

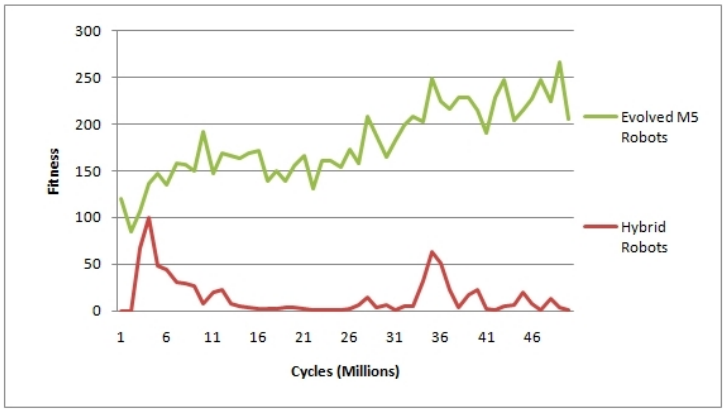


Figure 6.14: Low Hybrid Fitness on Land

The Biological Species Concept provides an even clearer picture. Hybrids generated from the two populations are tested for fitness in both environments. On land their fitness levels drop off within the first few million cycles (see Figure 6.14) and they remain low for the rest of the experiment. In the water environment, they fare even worse. Figure 6.15 shows the hybrids remaining in a low-fitness status for the entire simulation. These two groups of robots, though descended from a single ancestral genotype, have become unable to produce viable hybrid offspring.

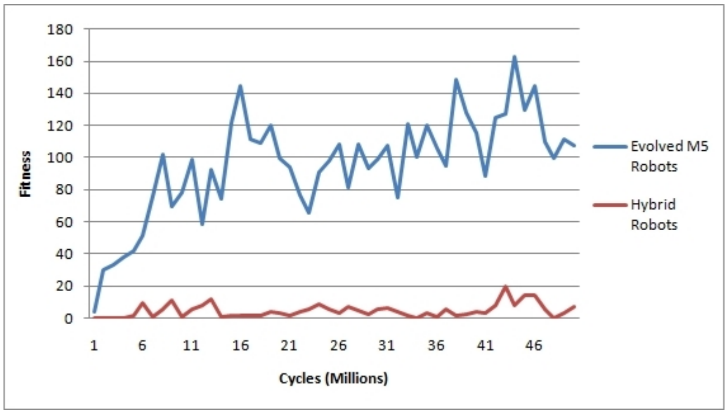


Figure 6.15: Low Hybrid Fitness in Water

6.3.3 Rate of Speciation

This is another example of speciation through the formation of post-zygotic reproductive isolation. However, it differs from the results from section 6.1.3 in its timing as well as in the extent of the isolation. These hybrids have even lower fitness levels and their performance decreased earlier - they appear to be *more* completely speciated. One begins to understand the continuous nature of speciation when reviewing these results. The completeness of the speciation points to a difference in the overall pattern of change between the populations. A comparison of all three experimental groups is shown in Figure 6.16. By the end of the test, the new water robots have in fact diverged further in both genotype and phenotype than the land robots did, when compared to one another.

The rapidity of the divergence also points to a difference in the rate at which the two populations were changing. Figure 6.17 shows a comparison of the rates of divergence in both the M5 land test and the M5 water test. This is measured as the product of the Genotype and Phenotype difference measurements as taken at increments of one million steps. There is not only substantially more difference between the robots, but it also happens much more quickly.

So in a computing system, an understanding of the difference between two environments may offer some predictive value of whether a speciation event could occur or not, as

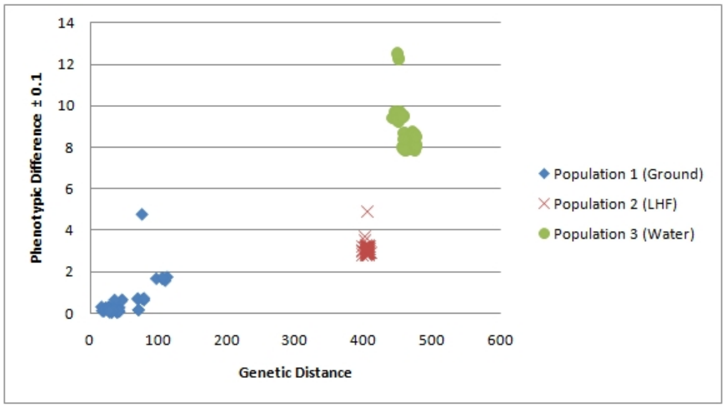


Figure 6.16: Difference in Overall Divergence

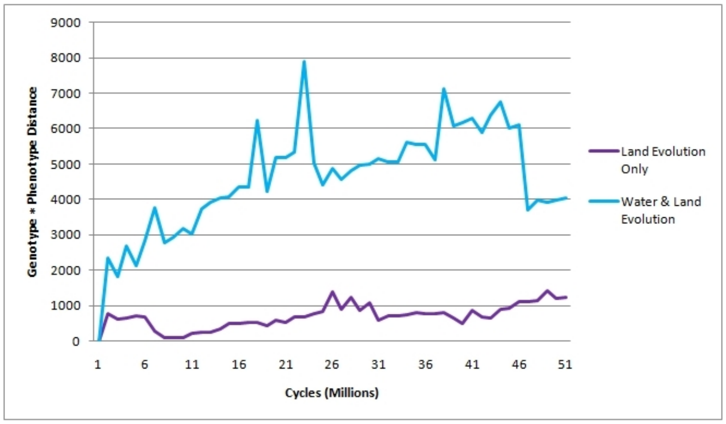


Figure 6.17: Difference in Rate of Divergence

well as the rate. This parallels the correlation between the level of ecosystem divergence and a promotion of reproductive isolation that is observed in nature [40].

6.4 Mutation Order

Though environmental differences do appear to affect the patterns of speciation, they are not a mandatory element for progress. Duplicating the previous experiments, a homogeneous population is split in two and separated geographically, which eliminates gene flow. However, instead of presenting the populations with two dissimilar environments, they will exist in identical worlds. Exactly the same selection and evolutionary pressure will be exerted on the two groups. In this situation it remains possible that the populations will, through the stochastic nature of mutations, take divergent evolutionary paths. This form of non-ecological allopatry is called Mutational Order Speciation[20].

Designing this experiment requires a relatively small adjustment from the previous setup. Rather than using a combination of ground, LHF and water environments, the mutation order test will only use ground targets, thereby creating identical worlds for both populations. Figure 6.18 shows the slow morphological evolution of the M5 robots in both populations. Without any difference in selective pressure, this process takes considerably longer. The experiment is run for two hundred million cycles, four times longer than the first M5 data in Figure 6.6.

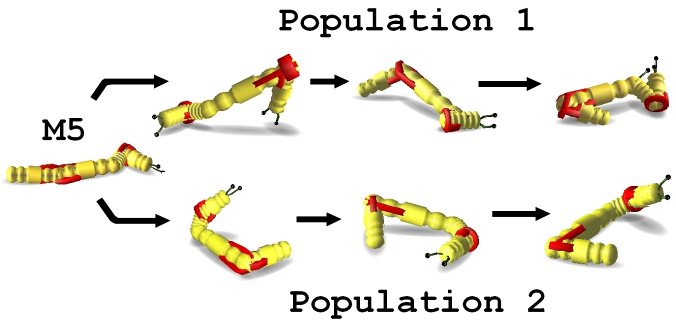


Figure 6.18: Evolutionary Progress in Mutation Order

The results in Figure 6.19 show a similar pattern but across the longer time scale. For the first hundred million cycles, which equates to approximately ten thousand generations, the hybrid robots perform comparably to both evolving populations. The eventual decrease in fitness takes nearly fifty million cycles to complete. Afterwards, the reproductive isolation remains stable with hybrids displaying consistently low fitness.

Morphological comparisons provide further evidence of the speciation beyond hybrid viability. The results show both populations growing apart but remaining relatively similar internally. The final phenotypic distance in Figure 6.20 is less than that of the other tests (see Figure 6.16). Again, this agrees with the prediction that environmental

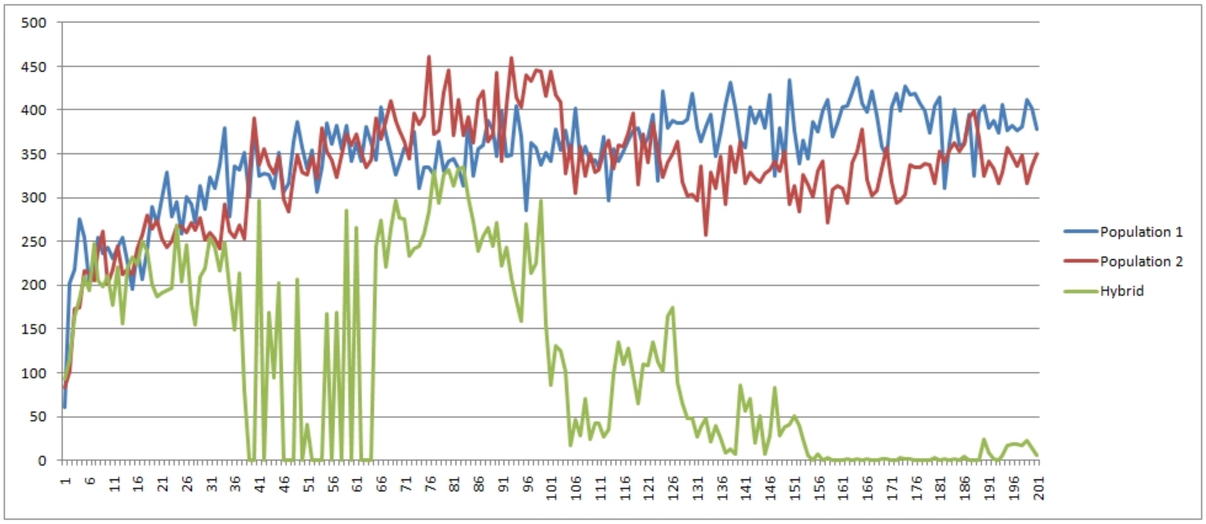


Figure 6.19: Fitness Results in Mutation Order Experiment

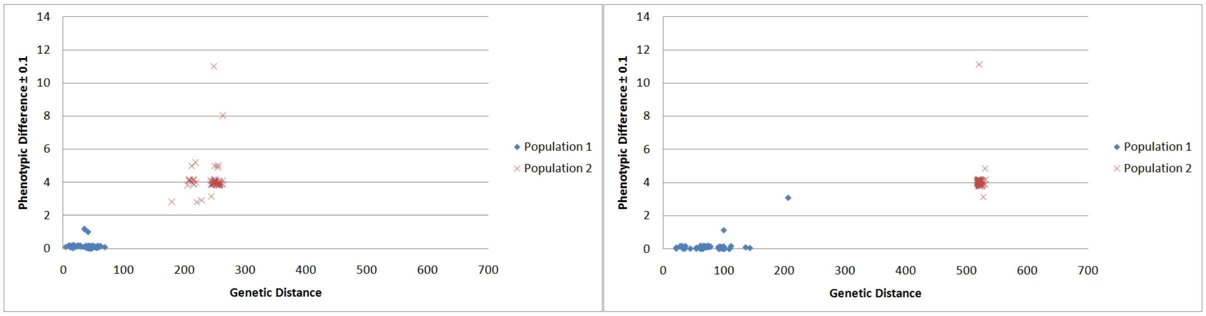
differences positively relate to the speciation process.

The results of this section may be additionally valuable to both biology and computing. Mutation order speciation is less substantiated than the more traditional scenarios precisely because it is difficult to show in nature [79]. Demonstrating that 3D simulation tools can describe process details at an individual level reveals another path of study. Also, most ER experiments do not include multiple differentiated environments or selection pressures. Evidence that robots may form species groups even in uniform surroundings is more applicable for comparison to other computing research.

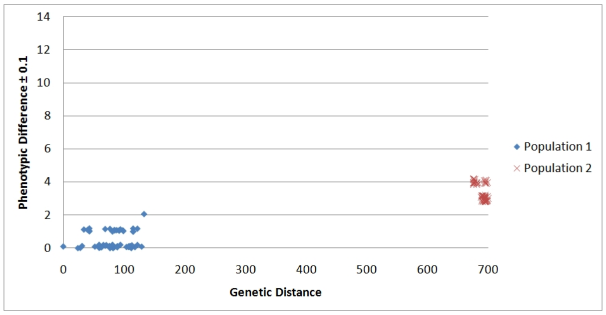
6.5 Genetic Drift

All of the results presented so far have related to two populations divergently adapting to their environments (even if they were the same). These forms of speciation are inherently selection driven, they arose from the robots gradually altering themselves for better performance. However, adaptation is not the only means of creating genomic change-chance plays a role as well.

Genetic Drift is defined as “random fluctuations in gene frequencies in a small breeding



(a) 10 Million Cycles (b) 100 Million Cycles



(c) 200 Million Cycles

Figure 6.20: Morphological Testing of Mutation Order Populations

population" [82]. While some of the mutations which occur during the transmission of genetic material impact performance, others do not. These fitness-neutral differences in the genotype do not necessarily constitute change in phenotype space. However their cumulative effect over many generations can amount to a very important change in genotype space.

The M5 population makes a poor genetic drift test subject for precisely the same reason it make a good ecological test subject, it evolves rapidly towards a more fit individual. To examine the effect of genetic drift requires that the robots maintain relatively stable performance values throughout the test. For this, another homogeneous population of robots, herein called M9, was evolved. As Figure 6.21 shows, the M9 robot uses a similar spring motion for locomotion but is smaller than M5. More importantly, it was evolved until its progress tapered off, thereby lowering the chance of the population making a jump in performance.

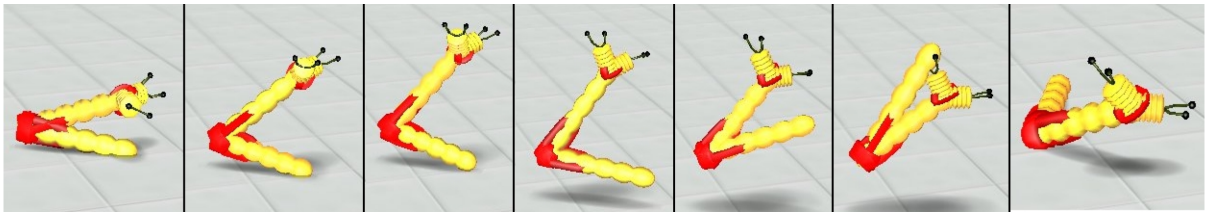


Figure 6.21: Locomotion in the M9 Robot Population

Genetic drift is believed to operate more heavily in smaller populations [41], so rather than 60 robots, these two populations will be comprised of only 25 robots. Similar to the mutation order test, only one environment is used for the experimental setup. In this case the suspended targets (LHF) are used, with no water or ground targets. The results shown in Figure 6.22 show the two M9 populations remaining stable throughout the test. This means that any genetic changes accrued by the robots were largely fitness neutral.

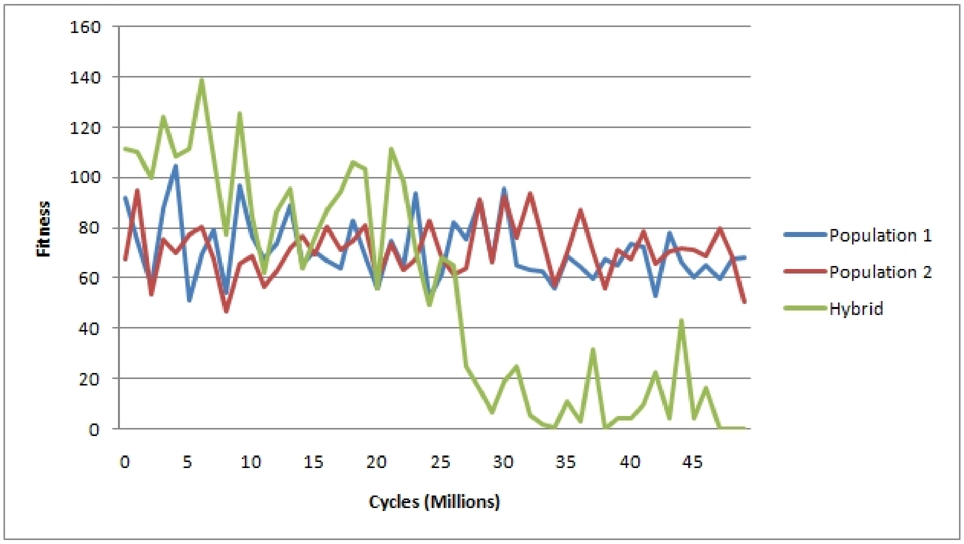


Figure 6.22: Speciation through Genetic Drift

Despite this lack of difference in performance and the complete lack of environmental differences, the two populations do begin to diverge. Hybrid fitness remains at comparable levels for over 25 million cycles but finally drops off. The two populations eventually lose the ability to produce viable offspring and become reproductively isolated demonstrating that robotic speciation is possible even in the absence of selection pressure.

6.6 Anagenesis

The previous results have all dealt with situations involving cladogenesis - speciation resulting from a branching event. This compares two populations as they evolve alongside one another. The alternative is anagenesis which describes speciation within a single lineage. It does not involve a branching event, as it is the evolution of one species into another. The comparative measure then, is not a co-evolving population but the ancestor species itself. This has application to understanding the evolutionary progress in many existing ER studies, specifically those that do not involve population segregation. The species concept could provide a framework for marking important change in an population as well as differentiating between evolutionary improvements that only impact fitness compared to those which change the genetics more fundamentally.

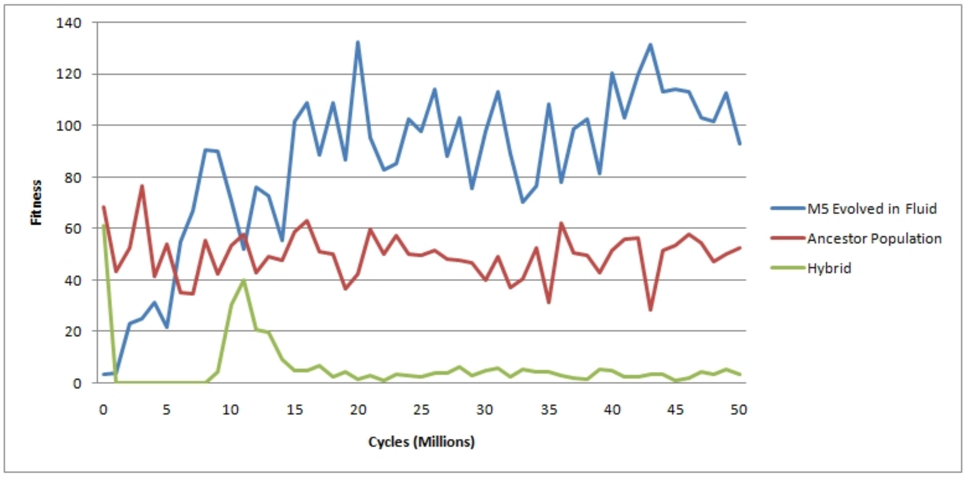


Figure 6.23: Anagenesis in Water Evolved Robots

To demonstrate anagenesis, consider the robots that evolved in water from the last test. Looking at their final form (Figure 6.13d) compared to the ancestral population of

M5 robots (Figure 6.13a), it is clear from a morphological standpoint that they lack much similarity. In biology, anagenesis typically involves species that have been separated by millions of years, so there is little information available about reproductive isolation [41]. A convenience that is available in simulated environments is that a copy of the ancestor population can be saved for testing, allowing the use of the BSC. In Figure 6.23 the same hybridization tests are performed with the evolving population and the original unchanging M5 population. After approximately fifteen million cycles the hybrid fitness levels permanently drop to almost zero. Enough change has accumulated in the new population that it can no longer generate viable hybrids with the M5 robots. In Section 6.3.2 it was demonstrated that these water robots are a different species than the evolved land robots - whereas this shows that they are a different species than the M5 robots as well.

6.7 Evolving the Controller

Most ER experiments involve evolution of the robotic controller, and occasionally the utilization of the sensors and actuators. Less common is complete morphological evolution, likely because this is difficult to implement outside of simulations. Demonstrating meaningful results to a larger research community requires a non-morphological variant of robotic speciation. To accomplish this, the physical structure of the robot is held constant through all tests and in both populations. All modifiers (see Figure 5.1) remain completely unchanged as well, so the body of the first robot is identical to the last robot. The neural network, sensors and actuators are evolved using the mutation rate shown in Figure 5.9. A population of structurally homogenous M5 robots was used to initialize the test. All other elements were kept the same as the Classical Scenario, using one environment of ground targets and the other with suspended targets (LHF).

Without morphological divergence, the only visual difference between the populations is behaviour. Figure 6.24 shows the evolved method of movement in the ground target environment. It is a basic crawling pattern where the robot rarely lifts entirely off the test surface. In contrast, the robots that evolved in the LHF test are able to effectively launch themselves both forward and up. This direction of locomotion, described in Figure 6.25 is ideal for collecting the suspended target items.

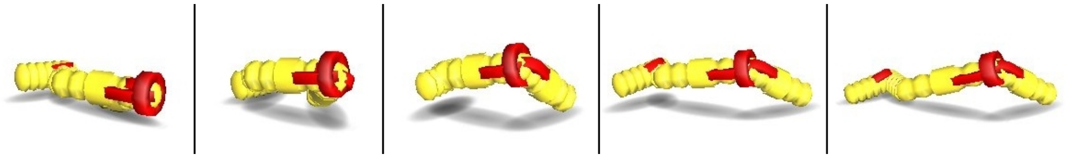


Figure 6.24: Locomotion for Population 1

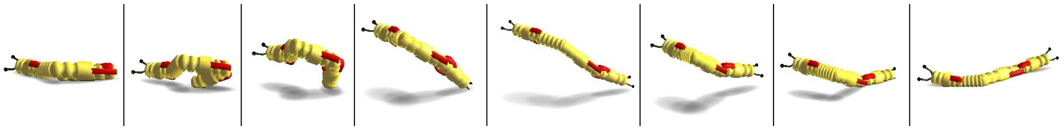
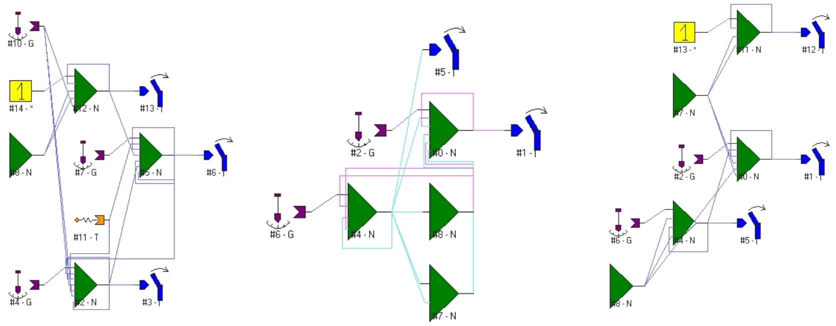


Figure 6.25: Locomotion for Population 2

Both populations evolve relatively simple neural networks to control their behaviours (see Figure 6.26). Despite their overall lack of complexity, the two are still substantially different both in layout and in function, as demonstrated by their movement. The dissimilar selection pressures between the populations lead them to generate two different controllers.

The fitness results shown in Figure 6.27 indicate that both groups achieved rapid fitness gains early into the simulation. This was followed by a sustained stabalization period marked by consistent performance values. The hybrid test scores show that speciation did not occur until near the forty million cycle mark. After this, the populations became reproductively isolated and were unable to produce viable robots from hybrid genotypes. The hybrids, of course, have the same identical structure as the original M5 population as well as all the evolved populations. However, the neural networks (an example is shown in Figure 6.26c) are not identical. Though it is not obvious from this diagram that any defects are present, the low fitness scores indicate that the controller is not able to generate the complex and synchronized patterns required to move the robot.



(a) Population 1 (b) Population 2 (c) Hybrid

Figure 6.26: Final Neural Networks in Non-Morphological Evolution

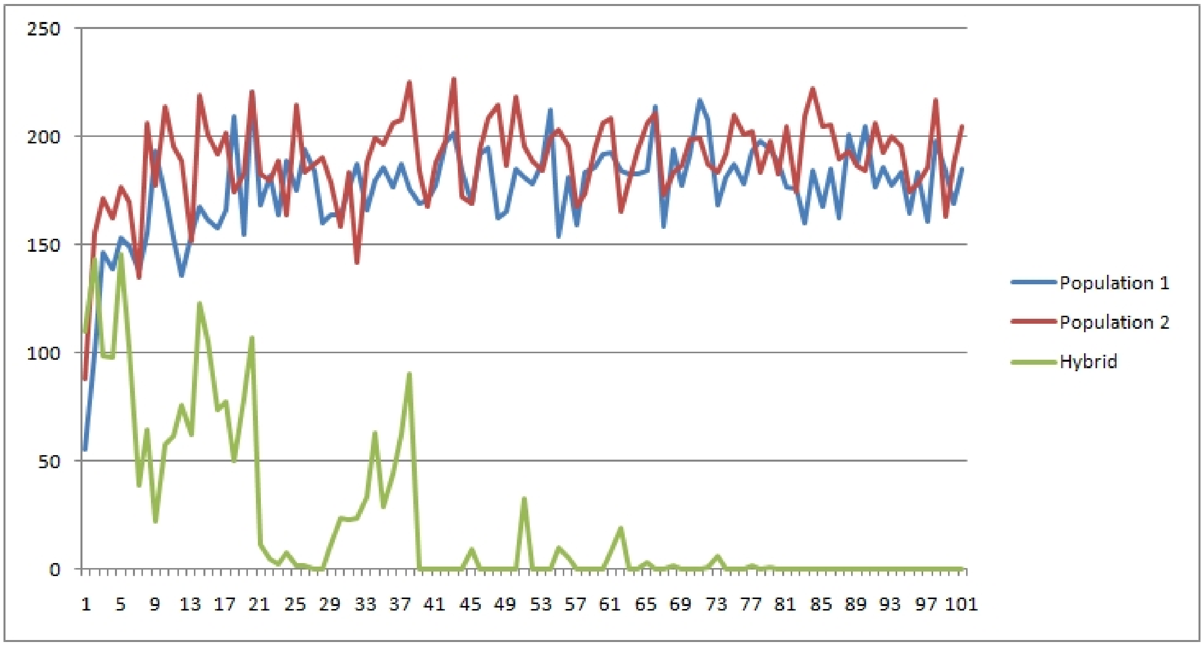
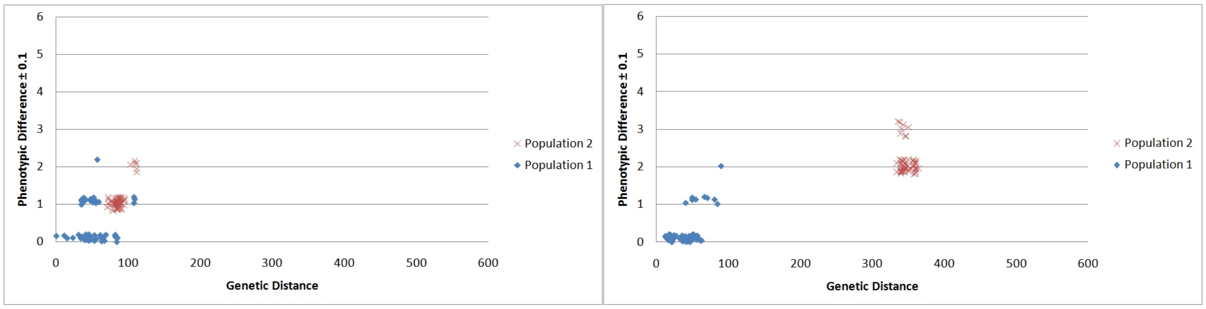


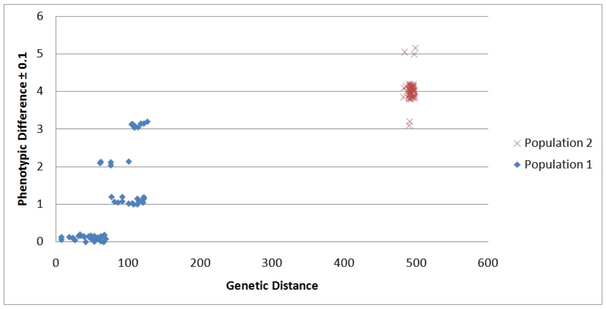
Figure 6.27: Performance Results from Non-Morphological Evolution

Phenotype and genotype testing also indicate that the two populations have diverged. Figure 6.28 shows the progression of a widening genetic gap between the groups. As with the other experiments, there is clear movement of the populations away from each other but internally the robot populations remain very similar. Distance builds across the phenotype axis as well because it incorporates a measure of neural similarity (as described in Section 5.7). This can be removed from the comparison equation, leaving only morphological elements. Not surprisingly the results of that test, pictured in Figure 6.29, show no difference between populations indicating that the robots remain structurally identical.



(a) 1 Million Cycles

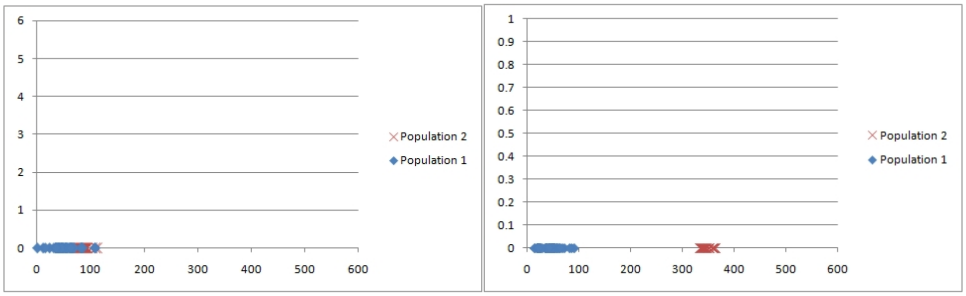
(b) 50 Million Cycles



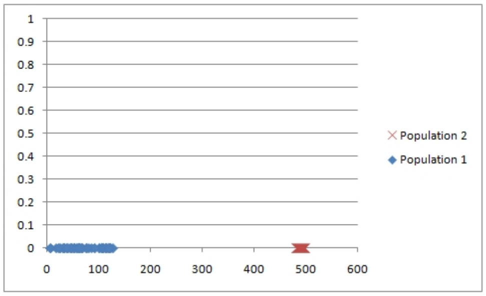
(c) 100 Million Cycles

Figure 6.28: Genotype/Phenotype Difference in Non-Morphological Evolution

An interesting feature of this speciation event is its timing with regard to the aforementioned stabalization period. When hybrid fitness levels decreased, the two populations had already ceased to increase in performance. This indicates that the speciation event may not have been the result of a difference in selective pressure but possibly due in part to genetic drift instead. In either case this demonstrates that speciation in robots can occur with non-morphological evolution as well.



(a) 1 Million Cycles (b) 50 Million Cycles



(c) 100 Million Cycles

Figure 6.29: Genotype/Phenotype Difference Ingorning Neural Similarity

6.8 Crossover function Proof

A certain amount of scepticism is always valuable, and it may have occurred to some readers that many of these results could have been generated not by speciation but by a faulty crossover mechanism. Although it seems rather unlikely, it is still theoretically possible that the evolutionary gains seen by both species were entirely the result of mutation, while concurrently every robot generated through crossover was lower fitness. Such a setup could yield results exactly like those in Figure 6.6, with two functional populations producing hybrids that fail. So it seemed worthwhile to dispel these concerns with a relatively simple test. If the genotype and phenotype divergence between two populations were removed then speciation is eliminated from the equation. This can be done rather easily by using identical populations for the test, rather than two divergent ones. The results are shown in Figure 6.30.

The data shows that the crossover function does not generate lower quality genotypes

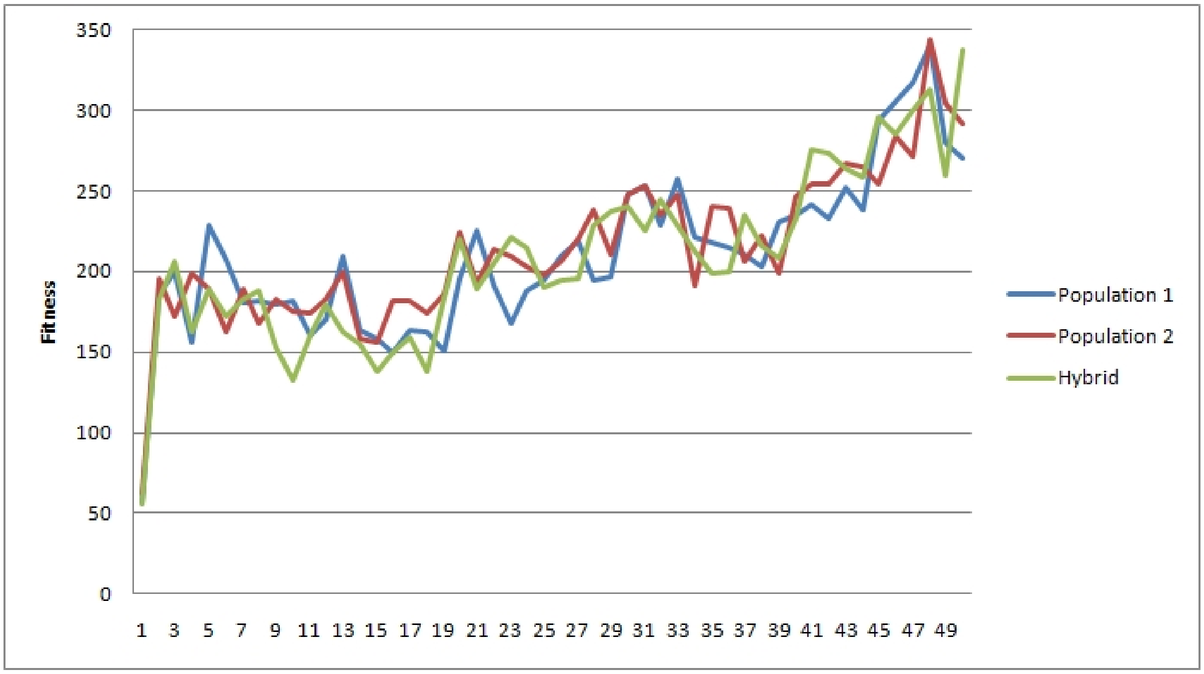


Figure 6.30: Crossover Operation Performed on Non-Divergent Populations

when divergence is eliminated. The hybrids produced are viable, high-functioning robots of equal fitness to the parent species. This proves that the low-fitness performance scores of hybrids seen throughout this thesis are not the effect of an inherently faulty crossover mechanism.

6.9 Speciation Simulation

While existing speciation simulations clearly serve a purpose, it is still argued that their accuracy could be improved [123]. The treatment of speciation as a discrete process, that is that it occurs instantly rather than over a period of time, is a gross oversimplification when reviewing a single event. Figure 6.31 shows the marked difference in the pattern of fitness between data generated from the non-morphological test compared to a species as defined only by genetic distance. Depending on the desired granularity of data, physical testing of hybrid viability may prove a useful tool for speciation simulations.

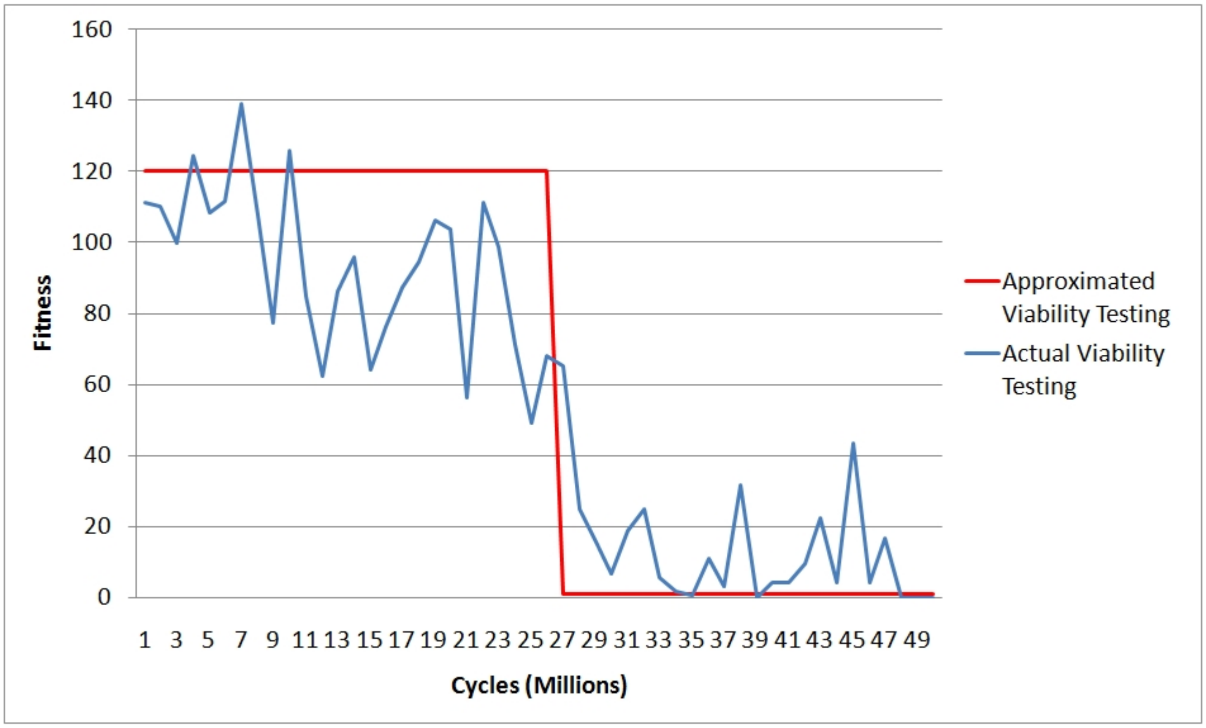
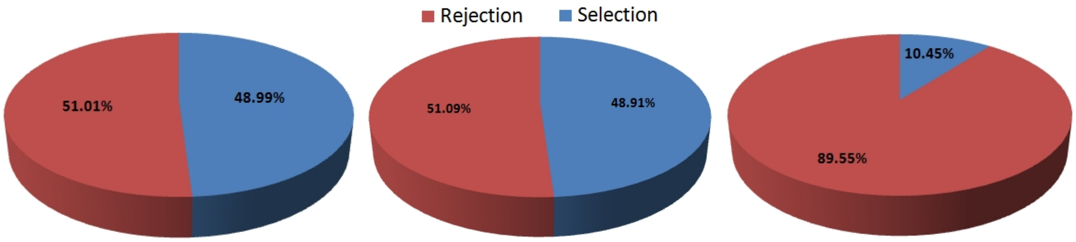


Figure 6.31: Comparison of Fitness between Discrete and Continuous Speciation

6.10 Secondary Isolation

Thus far, the demonstrated format for robotic speciation has been as follows: external isolation through separation leads to divergence resulting in low-fitness hybrids and therefore post-zygotic isolation. This exact same pattern is well understood to create biological speciation as well [123]. However most species in nature eventually develop multiple isolating barriers. For example, this inital post-zygotic isolation is often followed by a process called ‘reinforcement’. This is the active selection against hybridization as a result of lowered fitness, i.e. Organisms will not breed with hybrids because they do not appear to be good mates.

To test for this phenomenon in Framsticks a population of robots equally composed of the two species and the hybrids generated in the mutation order test will be used. After each robot is operated for 7500 cycles it randomly chooses another robot to mate with. The gene selection process used is identical to the one used in all the experiments as described by 1. In this method all mating events result in one of two possiblities: either the second robot is selected to pass on some genetic information, or it is rejected because it is not performing adequately. In general one would expect the average robot to have a selection rate of approximately 50%, half the time it has higher fitness half the time it has lower. This is precisely what is found in both species (see Figure 6.32). The hybrid robots, on the other hand, are rejected almost 90% of the time, demonstrating a strong selection against hybridized genotypes. In this way, the creation of a secondary pre-zygotic isolation barrier has naturally formed in the robot populations.



(a) Species 1 (b) Species 2 (c) Hybrids

Figure 6.32: Selection Against Hybrids

Chapter 7

Conclusion

This chapter will first present a quick discussion of the contributions and conclusions of this work and then summarize them. Finally, open questions and potential avenues for further research will be described.

7.1 Results and Conclusions

This thesis investigated the ability of a population of homogeneous robots to independently evolve into species groups as defined by biology. To accomplish this, experiments were performed in a physics-based 3D simulation platform called Framsticks. Genetically encoded robots were subjected to evolutionary pressures both neurologically and morphologically, forcing them to adapt to various scenarios and environments. The populations utilized decentralized sexual reproduction though a combination of crossover and mutation operators.

Two identical robot populations were separated so as to prevent contact, and then allowed to evolve in dissimilar environments - a scenario closely related to Darwins work. This resulted in the two populations progressing along divergent evolutionary paths. These two groups became increasingly different in their genetic makeup, a phenomenon that began to interfere with their ability to crossbreed. Hybrid robots, the genetic offspring of a pair of robots, one from each population, exhibited substantially lowered fitness - a demonstration that reproductive isolation barriers had formed. Having lost the capacity to generate viable offspring, the two robot populations emerged from their evolutionary journeys as different species.

In addition to the use of the Biological Species Concept for delineation, these robot species were further considered for both phenotype and genotype separation. Using a combination of morphological and neurological comparisons all individuals were analysed. That showed that the two populations had become quantifiably different from one another physically, yet the members of each population remained cohesive among themselves. An inspection of the genotypes yielded similar patterns, members within a species were genetically close to each other but distant from others . The two robot species were measurably distinct yet cohesive groupings.

It was further demonstrated that reproductive isolation in the robots gave rise to secondary pre-zygotic isolation barriers. In nature, the active selection against hybridized genotypes, due to their low fitness, further prevents gene flow between populations if they are no longer physically separated. This process of reinforcement is an important next step in the development of a species in nature.

Additional parallels between robotic evolutionary development and biology are demonstrated. Under testing robots exhibit a positive correlation between the degree of environmental differences and the process of genetic divergence similar to that which is seen in nature. This was accomplished by presenting the populations with fundamentally different environments - one on land and one in water. The result was an increase in both the rate of speciation and the overall genetic difference between the two populations.

A speciation event can also occur under different circumstances and through different mechanisms. When evolutionary changes that emerge as a result of environmental pressure lead to reproductive isolation, it is referred to as ecological speciation. This is in contrast to mutation order speciation wherein the two populations are separated but remain in similar environments. This arrangement was also performed using the same homogeneous population of robots. The experiment once again concluded with two populations that no longer effectively hybridize, as a result of evolutionary divergence.

Robots can form species groups even without the assistance of selection pressure.

Divergence in the robot genetics can also be the effect of random mutations rather than progress-driven selection. A test was performed in which two populations were simulated for a long period of time and yet neither showed any signs of evolutionary changes impacting performance. They did, however, accumulate enough different fitness-neutral changes to begin producing lower fitness hybrids. This shows that artificial speciation can occur through genetic drift.

It was further demonstrated that evolving robotics may undergo speciation without a separation of the populations. Anagenesis is a term used to describe the gradual change of one species into another within a single lineage. By comparing the initial population of robots against the final evolved population, it was shown that speciation had occured. The genotype and phenotype characteristics had changed significantly, but each robot remained similar to its co-existing relatives. A reproductive isolation barrier had also arisen in the form of low-fitness hybrids. This may be especially relevant to current ER experiments where allopatric separation is uncommon, providing a new method of understanding the evolutionary progression.

Further experiments were conducted by restricting evolutionary change to non-structural elements such as neurons and neural weights. Similar to the other experiments, hybrids generated from crossbreeding the populations had substantially lowered performance ratings. Even with entire sections of the genome held constant, the robot populations were able to generate enough genetic change to create reproductive isolation. This is additionally applicable to many existing ER platforms where complete morphological evolution is difficult or impossible.

Finally, the summation of these results indicates that 3D simulators may be a useful method of studying some aspects of speciation. They allow for actual hybrid viability testing rather than delineating species based on genetic distance. Treatment of a speciation event as a continuous process rather than a discrete one would more closely approximate the biological phenomenon.

7.2 Future work

The original goal of this work, to prove that robots can form into species was accomplished, but to understand the broader implications of this will require substantially more research. First, the mechanics of speciation are determined entirely by the definition that is chosen. This thesis primarily used the BSC and found supporting evidence through the MSC. The BSC was used for two basic reasons; first it is the most commonly used in biology. Second, groups divided by reproductive isolation have the feature of self-sustaining separation. This provides a platform-independent definition, whereas for instance, measuring species through genetic distance does not. However this still may not prove to be the best or most applicable choice for defining species in an artificial system. Unfortunately, the BSC does not apply well to the asexual systems that are relatively common in ER. Therefore future work may include an analysis of how speciation can occur within a variety of different setups including random or multi point crossover, or no crossover at all.

Having demonstrated speciation, the next important step may be to understand the conditions under which two species may remain stable. If the two populations are reunited will they both persist or will one overtake the other due to drift or perhaps a marginal difference in fitness? Speciation is tantamount to multiple incongruent solutions arising for a single problem. Determining how both solutions may maintain themselves could make this a useful tool for all EAs.

Appendix A

Appendix - Parameters

A.1 General Test Parameters

The following is the complete parameter list used for all tests. Exceptions and additional details will be noted in the following Section A.2.

```
created Sat Mar 19 13:18:50 2011
sim_params:
expdef:standard
usercode:
autosaveperiod:0
overwrite:1
filecomm:1
createrr:0
creatwarnfail:0
importchk:0
loadchk:0
groupchk:0
resetonexpdef:1
initialgen:X
capacity:200
delrule:0
MaxCreated:1
placement:0
rotation:0
creath:0.1
p_nop:20
evalcount:0
p_mut:64
p_xov:16
xov_mins:0
selrule:2
cr_c:0
cr_life:0
cr_v:1
cr_gl:0
cr_joints:0
cr_nnsiz:0
cr_nncon:0
cr_di:0
cr_vpos:0
```

```
cr_vvel:0

cr_norm:0

cr_simi:0

Energy0:10000

e_meta:1

feed:200

feede0:200

foodgen:

feedtrans:1

aging:0

stagnation:0

minfitness:0

boostphase:1

makesound:0

savebest:0

log:0

notes:

totaltestedcr:0

wrldtyp:0

wrldsiz:200

wrldmap:

wrldwat:-1

wrldbnd:2

wrldg:1

simtype:0

nnspeed:1

odeshape:0

odestep:0.05

odemusclemin:0

odemusclemax:10

odemusclespeed:1

odeairdrag:0.01

odewaterdrag:0.5

odewaterbuoy:1

odeseed:0

odesepsticks:0

odeworlderp:0.2

odeworldcfm:1e-05

odecolmumin:0.1

odecolmumax:5

odecolbounce:0.1

odecolbouncevel:0.01

odecolsoftcfm:0

odecolsofterp:0

odecol2mumin:0.1

odecol2mumax:1

odecol2bounce:0.1

odecol2bouncevel:0.01

odecol2softcfm:0

odecol2softerp:0

gen_hilite:1

gen_extmutinfo:0
```

```
genoper_f0:0
genoper_f1:0
genoper_f2:0
genoper_f3:0
genoper_f4:0
genoper_f7:0
neuadd_N:1
neuadd_Nu:0
neuadd_G:1
neuadd_T:1
neuadd_S:0
neuadd_*:1
neuadd_|:1
neuadd_@:1
neuadd_D:0
neuadd_Fuzzy:0
neuadd_VEye:0
neuadd_VMotor:0
neuadd_Sti:0
neuadd_LMu:0
neuadd_Water:0
neuadd_Energy:0
neuadd_Ch:0
neuadd_ChMux:0
neuadd_ChSel:0
neuadd_Rnd:0
neuadd_Sin:0
neuadd_Delay:0
neuadd_Light:0
neuadd_Nn:0
neuadd_PIDP:0
neuadd_PIDV:0
neuadd_SeeLight:0
neuadd_SeeLight2:0
neuadd_Sf:0
neuadd_Thr:0
f0_nodel_tag:1
f0_nomod_tag:1
f0_p_new:5
f0_p_del:5
f0_p_swp:10
f0_p_pos:10
f0_p_mas:10
f0_p_frc:10
f0_p_ing:10
f0_p_asm:10
f0_j_new:5
f0_j_del:5
f0_j_stm:10
f0_j_stf:10
f0_j_rsf:10
f0_n_new:5
```

```
f0_n_del:5
f0_n_prp:10
f0_c_new:5
f0_c_del:5
f0_c_wei:10
f1_xo_propor:0
f1_smX:0.05
f1_smJunct:0.02
f1_smComma:0.02
f1_smModif:0.1
f1_mut_exmod:EeAaMmSsIi
f1_nmNeu:0.05
f1_nmConn:0.1
f1_nmProp:0.1
f1_nmWei:1
f1_nmVal:0.05
f2_mutAddOper:0.4
f2_mutJointElem:0.33
f2_mutNeuroElem:0.33
f2_mutConnElem:0.33
f2_mutDelOper:0.1
f2_mutHandleOper:0.3
f2_mutPropOper:0.2
f3_mutSubstitution:0.6
f3_mutSubstPerChar:0.1
f3_mutDelIns:0.1
f3_mutDelInsPerChar:0.05
f3_mutDelInsLength:5
f3_mutDuplication:0.05
f3_mutTranslocation:0.15
f3_xovGeneTransfer:0.8
f3_xovCrossingOver:0.2
f4_mut_add:50
f4_mut_add_div:20
f4_mut_add_conn:15
f4_mut_add_neupar:5
f4_mut_add_rep:10
f4_mut_add_simp:50
f4_mut_del:20
f4_mut_mod:30
f7_mutAddChar:0.01
f7_mutAddGene:0.35
f7_mutReplaceChar:0.01
f7_mutDeleteChar:0.01
f7_mutReplaceGene:0.05
f7_mutDeleteGene:0.34
genkonw0:1
genkonw1:1
genkonw2:1
genkonw3:1
genkonw4:1
randinit:0.01
```

```
nnoise:0.01
touchrange:1
bnoise_struct:0
bnoise_vel:0
ncl_N:1
ncl_Nu:1
ncl_G:1
ncl_T:1
ncl_S:1
ncl_*:1
ncl_|:1
ncl_@:1
ncl_D:1
ncl_Fuzzy:1
ncl_VEye:1
ncl_VMotor:1
ncl_Sti:1
ncl_LMu:1
ncl_Water:1
ncl_Energy:1
ncl_Ch:1
ncl_ChMux:1
ncl_ChSel:1
ncl_Rnd:1
ncl_Sin:1
ncl_Delay:1
ncl_Light:1
ncl_Nn:1
ncl_PIDP:1
ncl_PIDV:1
ncl_SeeLight:1
ncl_SeeLight2:1
ncl_Sf:1
ncl_Thr:1
simil_method:0
simil_parts:0
simil_partdeg:1
simil_neuro:0.5
symPosSteps:10
symAlphaSteps:20
symBetaSteps:20
minjoint:0
maxjoint:2

GenePool:
name:Genotypes
fitness:return 0.0+this.velocity*1.0;
fitfun:0
fitm:2
fitma:2

Population:
```

```
name:Creatures
energy:1
death:1
nnsim:1
perfperiod:100
stabilperiod:100
killnostable:1000000
stabledist:0.01
enableperf:1
colmask:13
selfmask:65537
othermask:131073
bodysim:1
selfcol:0
em_stat:0
em_dyn:0
en_assim:0
Population:
name:Food
energy:1
death:1
nnsim:0
perfperiod:100
stabilperiod:100
killnostable:1000000
stabledist:0.01
enableperf:0
colmask:148
selfmask:131074
othermask:65538
bodysim:1
selfcol:0
em_stat:0
em_dyn:0
en_assim:0
```

A.2 Specific Test Parameters

In general, the lengths of the tests were not held rigidly, but if speciation did not occur within these timeframes the populations were usually abandoned. The following is a list of any exceptions to the parameter list in the previous section, as well as a summary of the experiment setups.

A.2.1 Ground Environment

ExpParams.feed/=2;

Ground targets are given half as many points to equalize fitness values.

A.2.2 LHF Environment

$$\forall \in Populations.group(1), Population.bodysim = 0;$$

By use of this parameter differentiation, the ‘floating’ targets of the LHF are created

A.2.3 Water Environment

World.wrldwat=3;

A.2.4 The Classical Scenario

The first test results, shown in Section 6.1, comprised an allopatric setup with 60 M5 robots evolving in both a ground environment and an LHF environment. Tests were run for 50 million cycles.

A.2.5 Anagenesis and the Water Test

The water test from Section 6.3.1 used LHF and Water environments with M5 populations. The tests were run for 50 million cycles. The anagenesis test from Section 6.6 used the data from the water evolved robots and compared it against their own ancestral M5 robot population. Comparisons were calculated for all 50 saved files.

A.2.6 Mutation Order

The mutation order test in Section 6.4 used two M5 populations both evolving in ground target environments. The test was run for 200 million cycles.

A.2.7 Genetic Drift

The genetic drift test in Section 6.5 was performed with the M9 population composed of 25 robots (rather than 60) (see Section B.2 for details), both populations were evolved in the LHF environments. The test concluded at 50 million cycles.

A.2.8 Non-Morphological

```
f1_smX:0
    f1_smJunct:0
    f1_smComma:0
    f1_smModif:0
    f1_mut_exmod:EeRrLlAaCcFfMmSsIiQq
```

These commands prevent any mutations in structural elements and modifiers. These parameters were used both for the test (see Section 6.7) as well as the generation of a special initial population. This population was derived from the same M5 robot as the other populations, except that all morphological features were kept constant in the 60 robots.

A.2.9 Additional notes

The parameter file in the previous section includes some evolutionary algorithm data such as

fitness:return 0.0+this.velocity*1.0;

However it should be noted that the built-in genetic algorithm for the Framsticks systems is entirely bypassed in these tests so as to use the algorithm defined in Section 1.

Appendix B

Appendix - Population information

B.1 M5

To generate the M5 robot, a copy of the s_7 genotype (evolved from scratch by Komosinski of Framsticks) was additionally evolved for 2 billion cycles.

The full Genotype to create the M5 Population is

```
LLLLLLLLLLLLccccFMMsiMsia(, llcfiwX[9:13.271, 3:-4.322, in:0.886299,
9:2.635][|G:4.175, 1:-3.078, 0:-0.305, fo:0.04, 3:1.986, fo:0.04,
7:0.175], rlcMX[|G:-432.668, 6:2.508, 0:-3.463, in:0.8, fo:0.04]llllllc
cffmmmIqqX[6:1.811]RRccccccccFFFFqqX[T, r:1][N][G][@, -5:25.396,
p:0.867][@1:-2.185, -7:13.467, 1:-0.653, s:0.521][|0:-1.341, *:2.605])
```

The original s_7 genotype is

```
LLLLLLLLLccccFMM(, rllcccfmmqqX[|G:4.480, 1:-1.673, 0:2.229], rlcMX
[|G:-429.431, 1:3.365]lllcccffmmqXcqX[@1:2.968][|0:-0.785, *:2.886])
```

B.2 M9

To generate the M9 robot, a copy of the s_2 genotype (evolved from scratch by Komosinski of Framsticks) was additionally evolved for 1.5 billion cycles.

The full Genotype to create the M9 Population is

```
lcffMMMIIIww(,,, ,, LLLLLLLLaFFFFFFFFMMMMMMMMX[*][*][@1:0.546,
7:-0.434, s:0.036, in:0.892368, si:-0.792136, si:2, in:0, in:0.8][|=:
3.615, 0:1.09, 7:0.142, s:0.099, si:2][N, 7:-13.323, -2:9.725, s:-0.342,
si:2, -1:-8.606]LMFLFMFFLLFMFFFFMFFLLLLLFFLFMMFFMLFFFFMMMFFMFMMFFLMFMMLFFLM
```

```
MFMMMFFFLLMMFFFFMMMMMLFFLFFLFFLFMMLFFLMFLFFFMMFMLLFLLFFFLLLLFFFLLMMMMMM
FMFLMMMMMFFFFMMMFFMFMMFFMLMFFLLFMFFFFMFFLLFMFFMMMFFFLLFFMFFLMMFFLLLFFL
LLFFFFMMMFFFLLMMMMMMMMMLLLLLFFFLLLLLFFFMMMMMLMFLFMMMFLMMMMMFFFLLMMMFM
MMMMMMMFFMMMFFFLFMMMFFFFLLLLLFMMMFFMMMFMLFFFLLFMFFFFMFFMMMFFFLLFFMFFL
LLLFFFFMMMFFFLLMMMMMMMFMFFLLFMFFFFMFFMFFLLFMFFFFMMFFFFMMMMMMMFFMMMFF
LLFFLFFFFLLMMMMFMFFLMFLMFFLMFLFFFMFFMLFffFFLFFLFFLFMMLFFLMFLMMMMMFFFFMMM
FFLFMFFFFMLFFFFMFMMMMMLFLFFFFFLffmmIqWW(,llllllllffffffffmmmmmmmm
ssiQQX[T],rlllllllffffffffmmmmmmmmssiiQX[T][@1:-1.724, -5:-20.771, s:-0.073,
-4:21.059, si:2, fo:1, 1:2.631, -5:2.656, -5:7.015,0:3.131][|=:0.724145,
0:-7.056, -4:3.556, si:1.75916,-7:3.35][@1:-5.497, in:0.869808, in:0]),
rLacF SiiwFqL(LLLLLLLLFFFFFFFFMMMMMMMMIX[@1:-16.504, -7:7.982,
si:2,2:1.695][|=:3.615, 0:1.115, s:0,si:2][|=:0.832162, 0:-22.453, s:-0.618,
0:14.862, -5:-0.16, -8:-11.705,-10:  -9.111,0:-18.096,-9:-7.763]))
```

The original s_2 genotype is

```
LLLLLLFFMMX[0:289.731][|0:0.350][@2:2.543, !:1.000, 2:-0.711]X[1:0.628]
[0:-2.267, 3:-1.828]X[@1:0.955][@1:1.964][@1:1.193][@1:1.193][1:1.788]
LLLLLLLLLFFFFMMMMMMX[@1:0.643][|=:3.615, 0:2.818]
```

Appendix C

Appendix - Additional Data Sets

Some additional results. Many, if not most tests, did not result in effective/timely speciation. No effort has been made to detail the statistical likelihood of speciation - it appears to depend heavily on a number of factors including environment and genotype.

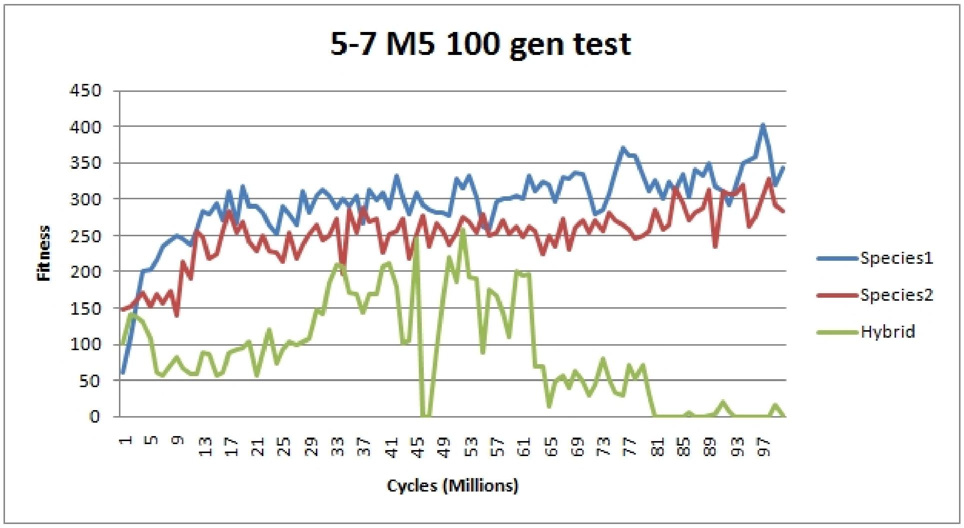


Figure C.1: Another basic allopatry test using the M5 population

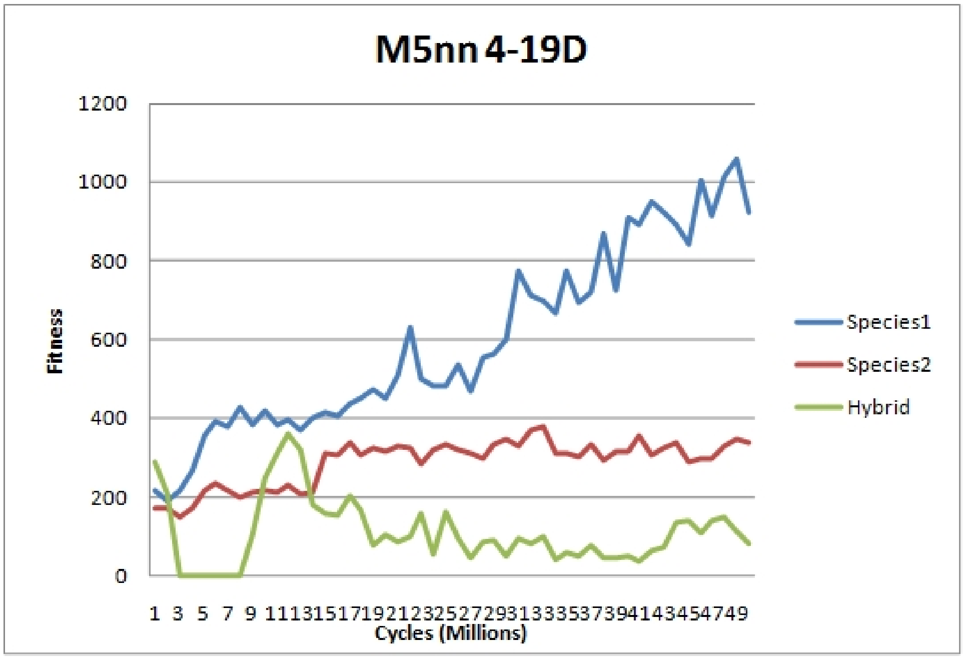


Figure C.2: Another non-morphological test using the M5 population

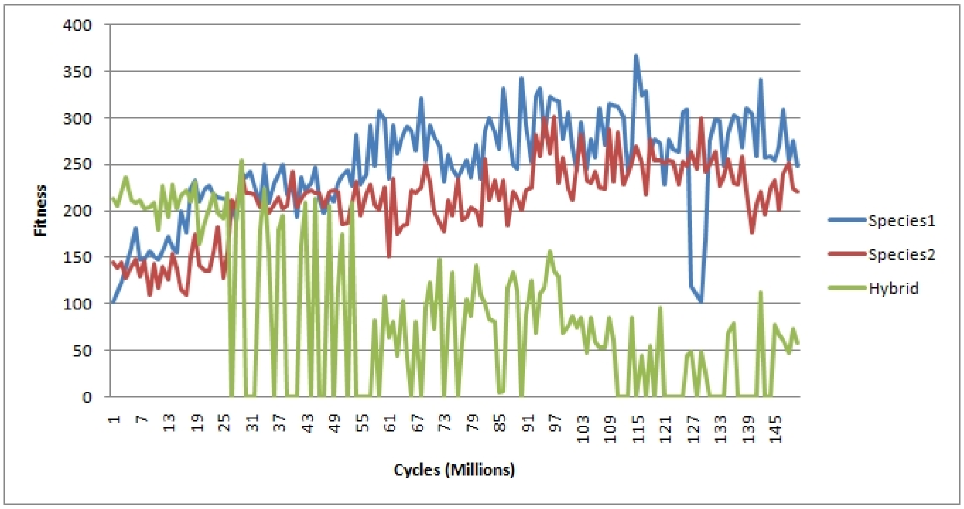


Figure C.3: Another mutation order test using the M5 population

Bibliography

[1] P.M Agapow, O.R.P Bininda-Emonds, K.A Crandall, J.L Gittleman, G.M Mace, J.C Marshall, and A Purvis. The impact of species concept on biodiversity studies. *Quarterly Review of Biology*, pages 161–179, 2004.

[2] D Ashlock, E.L Clare, T.E von Königslöw, and W Ashlock. Evolution and instability in ring species complexes: An in silico approach to the study of speciation. *Journal of theoretical biology*, 264(4):1202–1213, 2010. non physics sim of formation of rings species.

[3] J Auerbach and JC Bongard. Evolution of functional specialization in a morphologically homogeneous robot. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 89–96, 2009.

[4] G Baele, N Bredeche, E Haasdijk, S Maere, N Michiels, Y Vandepeer, T Schmickl, C Schwarzer, and R Thenius. Open-ended on-board evolutionary robotics for robot swarms. *E-Commerce Technology, IEEE International Conference on*, pages 1123–1130, 2009.

[5] R Barate and A Manzanera. Automatic design of vision-based obstacle avoidance controllers using genetic programming. *Proceedings of the Evolution artificielle, 8th international conference on Artificial evolution*, pages 25–36, 2007.

[6] N.H. Barton. *Evolution*. Cold Spring Harbor Laboratory Press, 2007.

[7] RK Belew. Artificial life: A constructive lower bound for artificial intelligence. *IEEE Expert*, 6(1):8–15, 1991.

[8] H.G Beyer and H.P Schwefel. Evolution strategies–a comprehensive introduction. *Natural computing*, 1(1):3–52, 2002.

[9] C William Birky and T.G Barraclough. Asexual speciation. *Lost Sex*, pages 201–216, 2009.

[10] J Bongard. Behavior chaining: Incremental behavior integration for evolutionary robotics. *Artificial Life*, 11:64, 2008.

[11] J Bongard. The utility of evolving simulated robot morphology increases with task complexity for object manipulation. *Artificial Life*, 16(3):201–223, 2010.

[12] JC Bongard and R Pfeifer. Evolving complete agents using artificial ontogeny. *Morpho-functional machines: The new species (designing embodied intelligence)*, pages 237–258, 2003.

[13] N Bredeche, E Haasdijk, and A Eiben. On-line, on-board evolution of robot controllers. *Artifical Evolution*, pages 110–121, 2010.

[14] R Brooks. Intelligence without representation. *Artificial intelligence*, Jan 1991.

[15] L Bull. On evolving social systems: Communication, speciation and symbiogenesis. *Computational & Mathematical Organization Theory*, 5(3):281–302, 1999.

[16] L Bull and T Fogarty. Evolutionary computing in multi-agent environments: speciation and symbiogenesis. *Parallel Problem Solving from Nature—PPSN IV*, pages 12–21, 1996.

[17] D Chowdhury and D Stauffer. Computer simulations of history of life: speciation, emergence of complex species from simpler organisms, and extinctions. *Physica A: Statistical Mechanics and its Applications*, 340(4):685–696, 2004.

[18] A Christensen and M Dorigo. Incremental evolution of robot controllers for a highly integrated task. *From Animals to Animats 9*, pages 473–484, 2006.

[19] A Della Cioppa, C De Stefano, and A Marcelli. Where are the niches? dynamic fitness sharing. *Evolutionary Computation, IEEE Transactions on*, 11(4):453 – 465, 2007. other kind of speciation.

[20] B Clarke. Mutational order: a major stochastic process in evolution. *Proceedings of the Royal Society of London. Series B . . .* , Jan 1990.

[21] R Clement. Visualizing speciation in artificial cichlid fish. *Artificial Life*, 12(2):257, 2006.

[22] J.A. Coyne and H.A. Orr. *Speciation*. Sinauer Associates, 2004.

[23] P Darwen and X Yao. Automatic modularization by speciation. *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation (ICEC'96)*, pages 88–93, 1996. other kind (fitness sharing).

[24] C Darwin. On the origin of species by means of natural selection, or the preservation of favoured races in the struggle for life. *London*, 1878.

[25] T.G. Dobzhansky and T. Dobzhansky. *Genetics and the origin of species*. Columbia Classics in Evolution Series. Columbia University Press, 1937.

[26] M Dorigo. . . . titre de docteur en sciences appliquées on the evolution of self-organising behaviours in a swarm of autonomous robots. *Citeseer*, Jan 2008.

[27] E. J. P. Earon. *The origin of digital species: The evolution of autonomous agents and lineages in a simulated ecosystem*. PhD thesis, University of Toronto (Canada, 2005.

[28] A Eiben, P Raué, and Z Ruttkay. Genetic algorithms with multi-parent recombination. *Parallel Problem Solving from Nature—PPSN III*, pages 78–87, 1994.

[29] E Elbeltagi, T Hegazy, and D Grierson. Comparison among five evolutionary-based optimization algorithms. *Advanced Engineering Informatics*, 19(1):43–53, 2005.

[30] Electronic Arts Inc. `http://www.spore.com/`.

[31] Trujillo et al. Speciation in behavioral space for evolutionary robotics. *Journal of intelligent and robotic systems*, pages 1–29, 2011.

[32] European Commission. `http://www.symbrion.eu/`, 2011.

[33] SG Ficici, RA Watson, and JB Pollack. Embodied evolution: A response to challenges in evolutionary robotics. *Proceedings of the Eighth European Workshop on Learning Robots*, pages 14–22, 1999.

[34] BM Fitzpatrick, JA Fordyce, and S Gavrilets. Pattern, process and geographic modes of speciation. *Journal of Evolutionary Biology*, 22(11):2342–2347, 2009. mostly about sympatric.

[35] D Floreano, S Nolfi, and F Mondada. Co-evolution and ontogenetic change in competing robots. *Advances in the evolutionary synthesis of intelligent agents*, pages 273–306, 2001.

[36] D Floreano and J Urzelai. Evolutionary robots with on-line self-organization and behavioral fitness. *Neural Networks*, 13(4-5):431–443, 2000.

[37] Dario Floreano, Stefano Nolfi, and Francesco Mondada. Competitive co-evolutionary robotics: from theory to practice. In *Proceedings of the fifth international conference on simulation of adaptive behavior on From animals to animats 5*, pages 515–524, Cambridge, MA, USA, 1998. MIT Press.

[38] JA Foster. Evolutionary computation. *Nature Reviews Genetics*, 2(6):428–436, 2001.

[39] S. Freeman and J.C. Herron. *Evolutionary analysis*. Pearson Prentice Hall, 2007.

[40] D.J Funk, P Nosil, and W.J Etges. Ecological divergence exhibits consistently positive associations with reproductive isolation across disparate taxa. *Proceedings of the National Academy of Sciences of the United States of America*, 103(9):3209, 2006.

[41] D.J. Futuyma. *Evolution*. Sinauer Associates, 2009.

[42] S Gavrilets. Perspective: models of speciation: what have we learned in 40 years? *Evolution*, 57(10):2197–2215, 2003.

[43] S Gavrilets, R Acton, and J Gravner. Dynamics of speciation and diversification in a metapopulation. *Evolution*, 54(5):1493–1501, 2000.

[44] S Geritz, É Kisdi, and G Meszéna.... Adaptive dynamics of speciation: ecological underpinnings. *Adaptive Speciation*, Jan 2004.

[45] D.E Goldberg and J.H Holland. Genetic algorithms and machine learning. *Machine Learning*, 3(2):95–99, 1988.

[46] S Goschin, E Franti, M Dascalu, and S Osiceanu. Combine and compare evolutionary robotics and reinforcement learning as methods of designing autonomous robots. *IEEE Congress on Evolutionary Computation, 2007. CEC 2007*, pages 1511–1516, 2007.

[47] B.K. Hall, B. Hallgrímsson, and M.W. Strickberger. *Strickberger's evolution: the integration of genes, organisms and populations*. Jones and Bartlett, 2008.

[48] T.I Hayashi and M Kawata. Impact of ancestral populations on postzygotic isolation in allopatric speciation. *Population Ecology*, 48(2):121–130, 2006.

[49] N Jakobi, P Husbands, and I Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. *Advances in Artificial Life*, pages 704–720, 1995.

[50] T Jones. Crossover, macromutation, and population-based search. *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 73–80, 1995.

[51] K-Team Corporation. `http://www.k-team.com/mobile-robotics-products/`, 2011.

[52] G Kampis, L Gulyás, and S Sos. The species problem in artificial life. *First IEEE Conference on Artificial Life. Honolulu, HI*, 2007.

[53] D Katagami and S Yamada. Interactive evolutionary robotics from different viewpoints of observation. *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, 2:1108–1113 vol. 2, 2002.

[54] S Kernbach, E Meister, F Schlachter, K Jebens, M Szymanski, J Liedke, D Laneri, L Winkler, T Schmickl, and R Thenius. Symbiotic robot organisms: Replicator and symbrion projects. *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, pages 62–69, 2008.

[55] S Kernbach, E Meister, O Scholz, R Humza, J Liedke, L Ricotti, J Jemai, J Havlik, and W Liu. Evolutionary robotics: the next-generation-platform for on-line and on-board artificial evolution. *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pages 1079–1086, 2009. symbrion.

[56] M. Kimura. *The neutral theory of molecular evolution.* Cambridge University Press, 1985.

[57] Komosinski and Ulatowski. `http://www.framsticks.com/`, 2011.

[58] M Komosiński and M Kubiak. Taxonomy in alife. measures of similarity for complex artificial organisms. *Advances in artificial life*, pages 685–694, 2001.

[59] M Komosinski and A Rotaru-Varga. Comparison of different genotype encodings for simulated three-dimensional agents. *Artificial Life*, 7(4):395–418, 2001.

[60] L König and H Schmeck. A completely evolvable genotype-phenotype mapping for evolutionary robotics. *2009 Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, pages 175–185, 2009.

[61] J.R. Koza. *On the programming of computers by means of natural selection.* A Bradford book. MIT Press, 1996.

[62] JR Koza and JP Rice. Automatic programming of robots using genetic programming. *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, pages 194–194, 1992.

[63] J Labonne and A.P Hendry. Natural and sexual selection giveth and taketh away reproductive barriers: models of population divergence in guppies. *American Naturalist*, 176:26–39, 2010.

[64] A Lafusa. Studying long-term evolution with artificial life. *IEEE Symposium on Artificial Life, 2007. ALIFE'07*, pages 15–22, 2007.

[65] W.P Lee, J Hallam, and H.H Lund. Applying genetic programming to evolve behavior primitives and arbitrators for mobile robots. *Evolutionary Computation, 1997., IEEE International Conference on*, pages 501–506, 1996.

[66] H Lipson and JB Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978, 2000.

[67] Anske Van Luijtelaar, Sjoerd Kranedonk, David Smits, and Joachim De Greeff. Sympatric speciation in framsticks virtual life research project, 2004.

[68] H Majeed and C Ryan. A less destructive, context-aware crossover operator for gp. *Genetic Programming*, pages 36–48, 2006.

[69] E. Mayr. *Systematics and the origin of species, from the viewpoint of a zoologist.* Harvard University Press, 1942.

[70] M Mazzapioda, A Cangelosi, and S Nolfi. Evolving morphology and control: A distributed approach. *Evolutionary Computation, 2009. CEC'09. IEEE Congress on*, pages 2217–2224, 2009.

[71] T Menezes and E Costa. Designing for surprise. *Advances in Artificial Life*, pages 1079–1088, 2007.

[72] O Miglino, HH Lund, and S Nolfi. Evolving mobile robots in simulated and real environments. *Artificial Life*, 2(4):417–434, 1995.

[73] Neil Fazakerley. `http://www.anf.nildram.co.uk/beebcontrol/arms/`, 2011.

[74] A Nelson, E Grant, and T Henderson. Competitive relative performance evaluation of neural controllers for competitive game playing with teams of real mobile robots. *NIST SPECIAL PUBLICATION SP*, pages 43–50, 2002.

[75] AL Nelson and E Grant. Developmental analysis in evolutionary robotics. *2006 IEEE Mountain Workshop on Adaptive and Learning Systems*, pages 201–206, 2006.

[76] Nick Jakobi. Minimal simulations for evolutionary robotics. Master's thesis, University of Sussex, 1998.

[77] S Nolfi. Evolutionary robotics: Exploiting the full power of self-organization. *Connection Science*, 10(3):167–184, 1998.

[78] Stefano Nolfi and Domenico Parisi. Evolution of artificial neural networks. In *In Handbook of brain theory and neural networks*, pages 418–421. MIT Press, 2002.

[79] P Nosil and S.M Flaxman. Conditions for mutation-order speciation. *Proceedings of the Royal Society B: Biological Sciences*, 278(1704):399, 2011.

[80] P Nosil, L.J Harmon, and O Seehausen. Ecological explanations for (incomplete) speciation. *Trends in Ecology & Evolution*, 24(3):145–156, 2009.

[81] H.A Orr and D.C Presgraves. Speciation by postzygotic isolation: forces, genes and molecules. *Bioessays*, 22(12):1085–1094, 2000.

[82] Oxford English Dictionary. `http://dictionary.oed.com/`.

[83] S Parham. Robotic self-appraisal and reproduction. *Kybernetes*, 24(1):34–43, 1995.

[84] GB Parker, GE Fedynyshyn, C Coll, and N London. Enhancing embodied evolution with punctuated anytime learning. *IEEE International Conference on Systems, Man and Cybernetics, 2007. ISIC*, pages 190–195, 2007.

[85] GB Parker, PJ Nathan, C Coll, and N London. Evolving sensor morphology on a legged robot in niche environments. *Automation Congress, 2006. WAC'06. World*, pages 1–8, 2006.

[86] F Pasemann, M Hülse, and K Zahedi. Evolved neurodynamics for robot control. *Proc. European Symposium on Artificial Neural Networks*, 2:439–444, 2003.

[87] JL Payne, MJ Eppstein, and CJ Goodnight. Sensitivity of self-organized speciation to long-distance dispersal. *Proceedings of the IEEE Symposium on Artificial Life*, pages 1–7, 2007.

[88] ALF Perez, G Bittencourt, and M Roisenberg. Embodied evolution with a new genetic programming variation algorithm. *Fourth International Conference on Autonomic and Autonomous Systems*, pages 118–123, 2008.

[89] A Petrowski and MG Genet. A classification tree for speciation. *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99*, 1:204–211, 1999.

[90] ML Pilat and C Jacob. Creature academy: A system for virtual creature evolution. *IEEE Congress on Evolutionary Computation, 2008. CEC 2008.(IEEE World Congress on Computational Intelligence)*, pages 3289–3297, 2008.

[91] M.L Pilat and C Jacob. Evolution of vision capabilities in embodied virtual creatures. *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 95–102, 2010.

[92] M.A Potter, L.A Meeden, and A.C Schultz. Heterogeneity in the coevolved behaviors of mobile robots: The emergence of specialists. *INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE*, 17(1):1337–1343, 2001.

[93] J Pugh and A Martinoli. Parallel learning in heterogeneous multi-robot swarms. *IEEE Congress on Evolutionary Computation CEC07*, pages 3839–3846, 2007.

[94] M. Ridley. *Evolution*. Blackwell Pub., 2004.

[95] M.R. Rose and L.D. Mueller. *Evolution and ecology of the organism*. Pearson Prentice Hall, 2006.

[96] H Rundle.... Ecological speciation. *Ecology Letters*, Jan 2005.

[97] Howard D Rundle and Patrik Nosil. Ecological speciation. *Ecology Letters*, 8(3):336–352, Jan 2005.

[98] E Şahin. . . . Special issue on swarm robotics. *Swarm Intelligence*, Jan 2008.

[99] M Saum and S Gavrilets. Ca simulation of biological evolution in genetic hyperspace. *Cellular Automata*, pages 3–12, 2006.

[100] D Schluter. Evidence for ecological speciation and its alternative. *Science*, 323(5915):737, 2009.

[101] D Schluter and G.L Conte. Genetics and ecological speciation. *Proceedings of the National Academy of Sciences*, 106(Supplement 1):9955, 2009.

[102] Y.S Shim and C.H Kim. Evolving physically simulated flying creatures for efficient cruising. *Artificial Life*, 12(4):561–591, 2006.

[103] EDV Simoes and KR Dimond. Embedding a distributed evolutionary system into a population of autonomous mobile robots. *IEEE international conference on systemsman and cybernetics*, 2:1069–1074, 2001.

[104] K Sims. Evolving virtual creatures. *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 15–22, 1994.

[105] J.M Sobel, G.F Chen, L.R Watt, and D.W Schemske. The biology of speciation. *Evolution*, 64(2):295–315, 2010.

[106] Donald Sofge, Mitchell A. Potter, Magdalena D. Bugajska, and Alan C. Schultz. Challenges and opportunities of evolutionary robotics. *CoRR*, abs/0706.0457, 2007.

[107] W Spears. Adapting crossover in evolutionary algorithms. *Evolutionary programming 4*, Jan 1995. old school notes on crossover.

[108] Stamm, K. Individual learning and the dynamics in predator-prey populations. Master's thesis, University of Gttingen, 2007.

[109] T Taylor and C Massey. Recent developments in the evolution of morphologies and controllers for physically simulated creatures. *Artificial Life*, 7(1):77–87, 2001.

[110] The Lego Group. `http://www.mindstorms.lego.com/`, 2011.

[111] Inc. The MathWorks. `http://www.mathworks.com/`, 2011.

[112] X THIBERTPLANTE and A.P Hendry. When can ecological speciation be detected with neutral loci? *Molecular Ecology*, 19(11):2301–2314, 2010.

[113] V Trianni, R Groß, TH Labella, E Sahin, and M Dorigo. Evolving aggregation behaviors in a swarm of robots. *Advances in Artificial Life*, pages 865–874, 2003.

[114] V Trianni, S Nolfi, and M Dorigo. Evolution, self-organization and swarm robotics. *Swarm Intelligence*, pages 163–191, 2008.

[115] E Tuci, C Ampatzis, F Vicentini, and M Dorigo. Operational aspects of the evolved signalling behaviour in a group of cooperating and communicating robots. *Symbol Grounding and Beyond*, pages 113–127, 2006.

[116] E Uchibe and K Doya. Finding intrinsic rewards by embodied evolution and constrained reinforcement learning. *Neural Networks*, 21(10):1447–1455, 2008.

[117] E Vassev and M Hinchey.... A formal approach to self-configurable swarm-based space-exploration systems. *... and Systems (AHS)*, Jan 2010.

[118] S Via. Natural selection in action during speciation. *Proceedings of the National Academy of Sciences*, 106(Supplement 1):9939, 2009. early speciation hard to tell.

[119] M. Wahde. Evolutionary Robotics: The use of Artificial Evolution in Robotics. Technical report, Department of Applied Mechanics Chalmers University of Technology Go teborg, Sweden, 2006. Technical Report TR-BBR-2005-002.

[120] J Walker, S Garrett, and M Wilson. Evolving controllers for real robots: A survey of the literature. *Adaptive Behavior*, 11(3):179, 2003.

[121] JH Walker, SM Garrett, and MS Wilson. The balance between initial training and lifelong adaptation in evolving robot controllers. *IEEE transactions on systems, man and cybernetics - part b: cybernetics*, 36(2):423, 2006.

[122] RA Watson, SG Ficici, and JB Pollack. Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18, 2002.

[123] F.J Weissing, P Edelaar, and G.S van Doorn. Adaptive speciation theory: a conceptual review. *Behavioral ecology and sociobiology*, pages 1–20, 2011.

[124] S Wischmann, K Stamm, and F Wörgötter. Embodied evolution and learning: The neglected timing of maturation. *Advances in Artificial Life*, pages 284–293, 2007.

[125] S Wright. Evolution in mendelian populations. *Genetics*, 16(2):97, 1931.

[126] C.I Wu and C.T Ting. Genes and speciation. *Nature Reviews Genetics*, 5(2):114–122, 2004.

[127] X. Yu and M. Gen. *Introduction to Evolutionary Algorithms*. Decision Engineering. Springer, 2010.