# AdaptiCritters: Simulating Population Dynamics with Genetic Algorithms

Justin Morera
Department of Computer Science
University of Central Florida
Orlando, FL
ju691930@ucf.edu

Darren Bansil
Department of Computer Science
University of Central Florida
Orlando, FL
darrenbansil@gmail.com

Christian King
Department of Computer Engineering
University of Central Florida
Orlando, FL
christian.king2@ucf.edu

*CCS Concepts:* • **Bio-inspired approaches → Genetic algorithms.**

***Keywords:*** Simulation, Evolution

**ACM Reference Format:**
Justin Morera, Darren Bansil, and Christian King. 2024 AdaptiCritters: Simulating Population Dynamics with Genetic Algorithms . In *ACM SIGOPS 30th Symposium on Operating Systems Principles (SOSP '24), November 4–6, 2024, Austin, TX, USA.* ACM, New York, NY, USA, 1 pages. https://doi.org/10.1145/1234567890

## 1. INTRODUCTION

Genetic algorithms (GA) have been used on a breadth of problems primarily as a search algorithm. GAs use the principles of evolution and natural selection to maintain a population of candidate solutions to the problem. By the end of several generations, the population should have evolved towards a solution that best optimizes the fitness function.

However, in this work we seek to simulate the evolution of an ambiguous species as it lives through generations of events that may increase or decrease the fitness offered by certain schemata. The nature and frequency of these events are detailed in Section 2. To achieve this, we made substantial modifications to a standard GA to best represent the environment and evolutionary process of a real-life species.

We surmise that a GA, which is built upon the principles of evolutionary growth and population dynamics, may be able to more closely mimic the foundational concepts to which it is so closely tethered to. We propose that an individual in a GA could represent an actual individual, or subpopulation of individuals, in a biological community; this further implies that that individual's genetic information could represent the actual phenotype present in a biological organism, or the average phenotype of a biological subpopulation. Using this notion, we believe, under carefully crafted experimentation and refinement, a run of the GA could be used to model the genetic evolution of a biological community under some set of environmental conditions.

To elaborate on this, consider the case of attempting to discern the genome of a "missing link" between *Homo sapiens sapiens* and the other primate groups. In reality, the scientific community only has access to the genetic information of primate extant species, and so the genome of a common ancestor between any two primate species is not easily inferred due to the massive amount of intermediate evolutionary stages from a distantly related common ancestor and the complexity of the changes required to alter one genome into another. But now consider the case where the common ancestor is known beforehand, and the extant genomes are accessible, but unknown. A GA can initialize a population of random phenotypes from an estimated genome, in this case, a genome with similar size and structure to modern-day species' (because we do not yet have implementations for insertions, deletions, inversions, or repeats), and gauge how these phenotypes interact over generations and form speciation events through adaptive radiation. Of course, a fitness function is needed in order to evaluate the likelihood of these phenotypes surviving long enough to survive to reproduce, and there also needs to be selection pressure.

Our model serves as a very rudimentary proof-of-concept in comparison to the glorious machinery that makes up the biological system. We assume that, with enough research and data availability, the scientific community can reliably quantify the relative fitness advantage awarded by a specific trait (or combination of phenotypic traits) within a given environment. If this craft can be perfected, then researchers can quantify all the relevant traits in a set of environments and then apply these environments to a GA population to see how environmental changes affect the fitness of certain phenotypes, and how those phenotypes adapt and evolve over time. In the case of a known common ancestor, if historical environmental change is replicated accurately, then it is surmisable that the resulting dominant phenotypes of the final GA population would resemble the dominant phenotypes present in extant species in the real world. This may

*SOSP '24, November 4–6, 2024, Austin, TX, USA.*

ACM ISBN 979-8-0000-0000-0/00/00...$15.00
https://doi.org/10.1145/1234567890

give credence as to the traits present in a common ancestor of any set of extant species, including humans and other primates.

Our research questions, as they pertain to our simulation in particular, are as follows:

1. **RQ1**: How does event density, the ratio of environmental events compared to the number of generations impact a population's growth and evolutionary path?
2. **RQ2**: How do interspersed cataclysmic events shape the population dynamics of a diverse community?
3. **RQ3**: Can a GA be used to simulate a real-world scenario by replicating environmental events in an ecosystem that generates phenotypes that resemble a species' evolutionary path realistically?

Section 2 of this work will go into more detail on the modifications made to the standard GA to support our simulation. Our experimental method and configurations will be covered in Section 3. The results of those experiments and a discussion of those results are in Section 4. Finally, we'll go over limitations of our work and conclusions in Section 5 and 6 respectively.

## 2. MODIFICATIONS TO STANDARD GA

Most applications of GAs fall into two categories: search and optimization. Because this work aims to leverage the structure and properties of GAs to simulate population evolution, some modifications were made in order to be able to support that.

### 2.1. Parameters

In addition to the standard parameters present in the GA outline provided to us, we added ; Number of Runs was not removed, but it was left at 1 for all experiments due to our implementation of events. Generations per Run was generally kept low due to the explosive nature of the population if not controlled carefully (this will be explained in greater detail later). Population size was generally set to 100, but the parameter was changed to only affect the initial population in generation 0, as there was no population cap set for this GA. Selection method was unused but was left in the parameter file. Crossover rate was kept at 1 and every surviving individual was allowed to reproduce each generation. Additionally, a fecundity parameter was implemented which controlled how many offspring each pair of individuals had during each generation. A fitness threshold parameter was utilized to scale the minimum fitness allowed before death; for our experiments, this was kept at 0 for simplicity's sake. Finally, a Data Input File Name parameter was used to give the name of a .txt file containing the number of genes, number of events, as well as the base genome structure and fitness values and the events with their trigger generations and fitness modifiers. The next two sections explain this feature in greater detail.

```
Experiment ID                :adapticritters
Problem Type                 :AC
Data Input File Name         :template.txt
Number of Runs               :1
Generations per Run          :20
Population Size              :100
Selection Method (1)         :1
Fitness Scaling Type (2)     :1
Crossover Type (3)           :3
Crossover Rate (4)           :1.0
Mutation Type (5)            :1
Mutation Rate (6)            :.01
Random Number Seed           :75982
Fecundity (9)                :2
Fitness Threshold (10)       :0
```

Listing 1: Example parameter file

### 2.2. Simulation Generator

We have developed and implemented a simulation generator that creates a template defining the genome of the species as well as the nature of the events and when they occur. Alternate implementations of the basic generator were created to support the experiments that will be discussed in the following section.

```
4
6 +49 +42 -30 +16 x -25
5 -47 x -38 +12 +24
8 +19 +36 -1 +12 x -40 x -13
2 x -5
1 3
2
+9 +1 +9 +8 x -4
+0 x -8 +1 -6
+8 +4 +6 +10 x +5 x +1
x +9
```

Listing 2: Example simulation template with four genes, one event, three generations at a 0.1 lethal chance

The size of the genome is decided by the `numGenes` parameter. Each gene generally has a random number of alleles ranging [2, 10]. Each allele is assigned an integer value that defines its contribution ranging $[-50,50]$ to the individual inheriting that allele. Another parameter, `lethalChance`, defines the probability of an allele being lethal. Lethal alleles are denoted with $x$ and are assigned a fitness value of $-1000000$, rendering the individual that inherits this allele deceased upon creation. Upon initialization, individuals will be assigned a phenotype that consists of a random subset of the alleles available in the genome. It is important to note that an individual in this context represents **all** individuals with that phenotype.

### 2.3. Events

Events are the driving force of evolution in our simulation and are meant to simulate environmental occurrences like hurricanes or long-term shifts in ecological landscapes like urbanization. Each event modifies the genome which subsequently modifies the phenotype of all living individuals and all individuals to be created. The effect of this modifier ranges [−10,10] for each allele.

Cataclysms are major, catastrophic events that are meant to replicate era-defining environmental occurrences such as ice ages. Just like cataclysms in real life, individuals are most likely to suffer but some individuals may thrive after the event is over. In practice, every allele has a 50% chance of having its fitness contribution lowered from [−50,0] and a 50% chance of having it modified from [−10,40]. Having events this severe allows us to see how the population adapts and how quickly it does so.

### 2.4. Representation

The scenario's genome was represented by a two-dimensional array of integers, with a column for every gene in the scenario's genome, and a row for each allele present in the specified gene. Each cell in the array was initialized to an integer based on the base fitness stat generated by the input file for that allele. There is additionally the option for an allele to be given the character 'x' in the template, which is processed by the GA as a lethal allele and given a uniquely low fitness value to demonstrate complete inviability in a genome. Individuals were initialized with a phenotype consisting of a one-dimensional integer array with an element for each gene and were initialized to an integer along the range of alleles for that gene. Individuals also had a number ID to track specific individuals, an ArrayList tracking the two parent individuals that created them, a second ArrayList tracking the children they created, and two integers tracking the generation they were "birthed" and the generation they "died". Additionally, the cumulative population of the entire species was tracked in a separate static integer.

### 2.5. Fitness

Fitness was calculated by iterating through an individual's phenotype array and summing the integer fitness values of the corresponding alleles present in that individual's phenotype. As an example, if a phenotype consisted of the sequence 2,0,3, the function would navigate to index 2 of the first gene in the genome, index zero in the second gene, and index 3 in the third gene, and return the sum of those values. After fitness was calculated, if the total fitness was below the threshold set as a parameter, the individual would be removed from the population before mating. Additionally, "lethal alleles", when implemented, would be stored with a fitness value of −1000000000, and when encountered in an individual's phenotype, the total fitness would be set to −1000000000 without the need to calculate any other alleles. These lethal alleles are added as a separate challenge to declare certain gene expressions as utterly unfit, to simulate genetic diseases or other debilitating traits or disorders.

### 2.6. Selection

Traditional selection was removed in lieu of a fitness threshold that "kills" individuals from the population if their total raw fitness drops below the set parameter. Surviving individuals were only allowed to reproduce once per generation (see crossover for details), and individuals with the highest fitness were given priority to select a mate, which was chosen randomly, giving more fit individuals a higher chance of selecting other highly fit individuals for crossover.

### 2.7. Crossover

Crossover was implemented as uniform crossover, where for every gene in the genome, one of the two parents' alleles would be chosen to be expressed in the child. There is also a fecundity parameter that controls how many offspring are generated from each mating event; each child individual undergoes a separate crossover event, followed by a chance for mutation. It should be noted that there is no replacement, individuals do not leave the population unless their fitness falls below the threshold, and their phenotype does not change after crossover and mutation; therefore, individuals that continue to survive, continually reproduce every generation. This makes environmental events paramount to population control and also adds a hefty challenge to maintaining a growing, but controllable population level throughout many generations.

### 2.8. Mutation

Mutation was implemented similarly to most GAs, where for each trait in a phenotype, there is a chance (set by the mutation rate parameter) to change that allele to a random allele available for that gene in the scenario's genome. This is often a good way to keep lethal alleles relevant past initialization because only individuals in the starting generation may be given lethal alleles by random chance and not survive to future generations, and as stated before, there is no further mutation after birth. We were unable to implement diploid representations and gene dominance due to time constraints, so there is currently no inheritability of lethal alleles; that being noted, most of our experiments minimized, or outright excluded the use of lethal alleles as they proved to be a hindrance to the explorative nature of the GA.

## 3. METHODS

In addition to simulating population evolution, this work also seeks to analyze how events can affect the population as it evolves over many generations.

### 3.1. Tracing The Evolutionary Path

A Python script was created to visualize the evolutionary path between phenotypes. Every individual created was given an ID, allowing us to track the individual as an object along with its phenotype and average fitness while the individual was alive. After removing duplicate phenotypes and individuals with negative fitness values, the remaining individuals underwent weighted random sampling using average fitness as the weights, meaning phenotypes with higher average fitness were more likely to be sampled. We kept the sample size at 100 phenotypes for all experiments. We decided to use pairwise-distance as the similarity metric to relate phenotypes to each other. A similarity matrix was then used to build a phylogenetic tree visualizing the relationship between phenotypes. Shorter branches of the tree represent closer relationships while longer branches represent more distant relationships. Do note that a direct branch between two phenotypes does not necessarily mean that they have a direct parent-child relationship. Due to the Python libraries used, we were unable to export these trees in a .PNG format to be readily used in this report [1]. Trees were displayed in ASCII art format in the console.

### 3.2. Event Density

The goal of this experiment is to see how the population reacts and adapts to varying frequencies of events. Standard events have an equal chance of improving or worsening the fitness contribution of each allele, so it would be interesting to see how the population converges to a phenotype depending on how many events occur over the life of the experiment.

| Parameter | Value |
|---|---|
| Generations | 30 |
| Uniform Crossover Rate | 1.0 |
| Bit Flip Mutation Rate | 0.1 |
| Fecundity | 2 |
| Population Size | 10 |
| Gene Size | 10 |
| Lethal Chance | 0.1 |

Table 1: Parameters for Event Density Experiment

This experiment was run for 30 generations with frequencies of 5, 10, 15, 20, and 25 events in the run. As discussed in Section 2.1, the modification potential of each event is random, so events could be relatively balanced with enhancements and deductions or events could be skewed in one direction.

### 3.3. Scattered Cataclysms

The goal of this experiment is to look at how much sudden environmental change a population can endure and of the populations that do, which individuals initiate that adaptation process. Larger populations allow for greater genetic diversity assuming a Gaussian distribution; greater genetic diversity leads to greater robustness to large changes in the fitness environment as the likelihood of adequate solutions existing is increased. Thus, our hypothesis for this experiment is as follows: *A large population that undergoes a major event is likely to have a less fit chromosome multiply and take over the population rather than an optimal solution to the old fitness environment adapting to the new environment.*

This experiment tests this hypothesis by inserting cataclysmic events and regular events randomly throughout the GA's timeline. An event entails modifications to the fitness of each gene in the genome. In the case of a regular event these fitness modifications are randomly chosen from [+10, −10]; catastrophic event fitness modifications are randomly chosen from [−50, 0]. About one in three generations will have an event and of those events about one half are cataclysmic. These parameters were chosen for simplicity of presentation; however, many other configurations have been tested in which the high level trends discussed in the respective results section of this paper still hold.

| Parameter | Value |
|---|---|
| Generations | 20-50 |
| Uniform Crossover Rate | 1.0 |
| Bit Flip Mutation Rate | 0.01 |
| Fecundity | 1 |
| Population Size | 100 |
| Gene Size | 100 |
| Lethal Chance | 0.05 |

Table 2: Parameters for Scattered Cataclysms Experiment

Many other parameters were tested and these were chosen as a balance between real world applicability and computational complexity.

### 3.4. Real-World Simulation

This experiment was designed to demonstrate, at a high level, how a GA can be used to mimic aspects of population dynamics by attempting to repeat theorized occurrences in evolutionary history. In this case, we chose to initialize a simplistic population of 100 ten-gene individuals where every gene had 10 alleles. Each even-indexed gene had positive base fitness values increasing from 0 to 18, with a step of 2, and each odd-indexed gene had negative base fitness values decreasing from 0 to −18, with a step of 2. There were 20 generations with 8 events, but each event's modifiers were controlled to purposefully direct allele fitness ratings toward a complete inversion of the most fit set of schemata. Initially, the optimal phenotype would be 9,0,9,0,9,0,9,0,9,0 where every even gene has the highest fitness allele expressed, totaling 18 x 5 + 0 x 5 = 90. However, the first two events lower all non-zero allele values by 2, reducing the disparity between the highest and lowest alleles for each gene; this takes place at generations 5 and 6. Shortly after, the rewards are reversed, in that, for the next two generations, the even genes' allele values are boosted by 18 for index 0, lowering by 2 for each subsequent allele, and the odd genes' allele values are boosted by 0 for index 0, raising by 2 for each subsequent allele. The result of this is that the lowest index for even genes becomes the most fit on the genome, and the highest index for odd genes becomes the second most fit. This process repeats for two cycles of one event lowering values by 2, and one event raising values between 2 and 18. Given the gradual degree this change takes place, the population as a whole experiences a shift where only the outlying phenotypes are excised from the population, and previously nondominant individuals suddenly see a rise in fitness.

## 4. RESULTS

Because of the computational complexity of the simulation, only one run was performed for each experiment.

### 4.1. Event Density

Unfortunately, we were unable to obtain concrete results to answer RQ1. We observed no long-term trend when increasing the number of events per number of generations. Increasing the initial population and number of generations to 1000 and 100 respectively (while keeping the relative density the same) showed more promising results; however, just one run with those parameters took four hours. We discuss time and computational limitations in Section 5. Our main interpretation for this conclusion is that the random nature of the event generation process may have created too much chaos in the population to allow convergence of any kind.

The first thing we learned from this experiment, however, was that the robustness of the population improved drastically when run with larger populations. Large populations are able to retain diversity that hedges against mass deaths after events occur. This is especially true when events occur consecutively. This is also why lower population sizes show negative average fitness values after detrimental events since they lack diversity. The second key takeaway is that population dynamics are extremely complex even at such a simple scale as this. It is more likely that we do not have the computational power nor the time to properly analyze the fine-grained effects of events at different frequencies and with varying fitness modifiers, than it is that patterns do not actually exist between different frequencies and types of events.

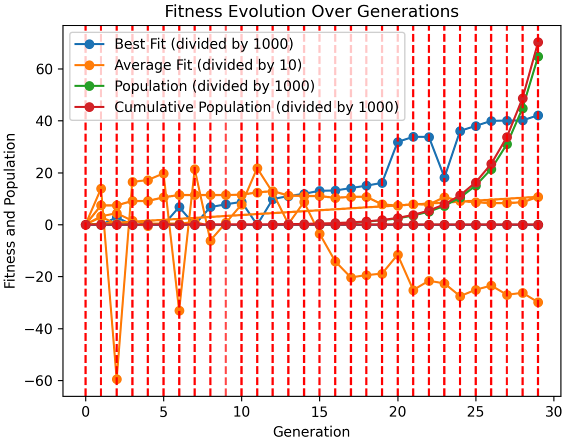

Figure 1: Fitness showed sporadic behavior when run with practical parameters (popSize at 10 over 30 generations)

### 4.2. Scattered Cataclysms

The results of this experiment partially confirmed our hypothesis but seemed to suggest a more nuanced reality. The results showed that when the fitness environment undergoes drastic changes, a previously suboptimal phenotype rapidly multiplies to take its place. In Figure 2, there are major cataclysms at generations 7, 14, 20, 28, 35, and 42 (marked by dotted red lines). All of these events change the environment such that the best fit individuals become less optimal. This can be seen at a high level by way of the average fitness metric dropping significantly.

The nuance is presented in the 2nd and 3rd most fit phenotypes as seen in Figure 3. If the data aligned with our hypothesis, the best fit phenotype would fall far out of favor and be drowned out and replaced by a new, more fit phenotype. However, this is not the case; in the early generations, the best fit phenotype stays at the top, even after the cataclysm of generation 7. Moreover, it stays the 3rd best fit after the cataclysm of generation 14. It is only replaced after the cataclysm of generation 20.

When accounting for these trends, it is clear that our hypothesis oversimplified the problem. Determining whether or not a best fit solution will stay the best fit even after a significant change to the fitness environment has proven to be a complex and nuanced problem that is not just dependent on severity of the cataclysm, but also the exact modifications of the event, the population and genetic diversity of the population bottleneck pre- and post-event, and the number of possible phenotypes available to the GA.

It is also worth noting that while no one factor determined the post cataclysmic outcome, it seems that population size likely had the most substantial impact on determining whether or not a population would survive a cataclysmic event. This is likely because larger populations inherently bring about more genetic diversity due to the

Gaussian nature of genetic algorithms. As such the likelihood of a decent solution to the new fitness environment already existing in the population is greater.

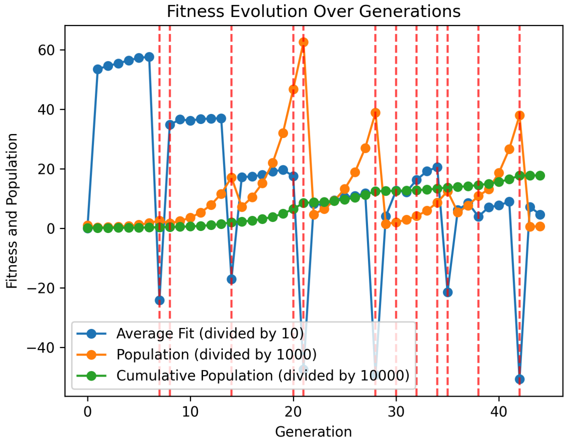

Figure 2: Scatter Cataclysms across a 45-generation run (red dotted lines mark cataclysms)

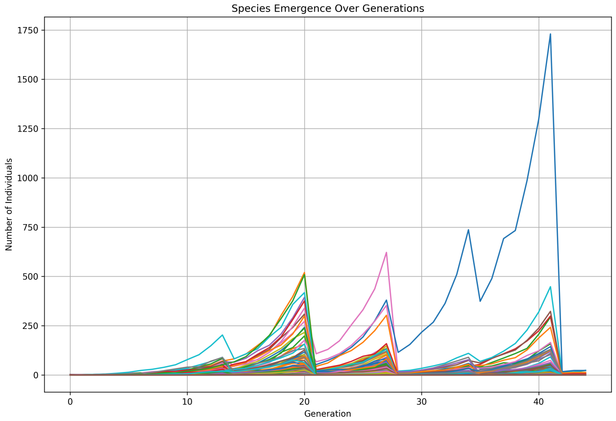

Figure 3: Number of individuals of a given genome where each line represents an individual genome (same run as Figure 2)

### 4.3. Real World Simulation

The results of this experiment show that we were able to demonstrate a shift in dominance between "subspecies" within a population. Once we were able to identify the proper graduation required to allow the population to drop without causing a mass extinction event, we were able to simulate a scenario where an environment changes in a way that prevents previously dominating individuals from thriving and moves to allow the most unsuccessful individuals to take over. This can be viewed in a number of ways, depending on the application: one, this can demonstrate a classic scenario of ecosystem decline, where the environment itself becomes less productive in the eyes of the prevalent species and begins to cater to less productive or opportunistic lifeforms. It can also be viewed as a species' exodus from a more suitable environment for a variety of reasons and the consequent adaptive radiation that results from adapting to new environments. One drastic, but evidenced example is the evolutionary path of whales, which came from a sea-dwelling common ancestor to other mammals, whose path eventually evolved legs and lungs and adapted to life on land, but then ultimately evolved back to ocean life [2].

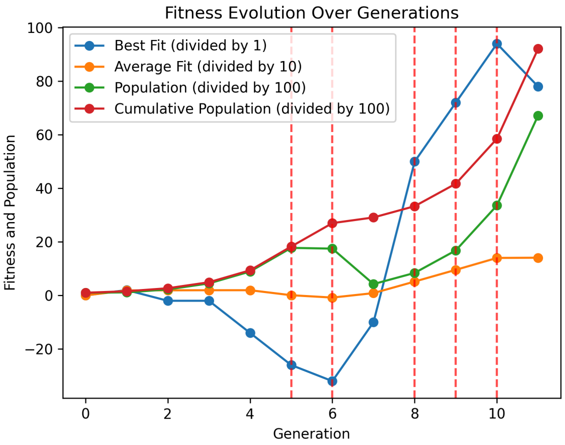

Figure 4: Evolution of fitness values over time; note the drastic decline by generation 6 before the subsequent resurfacing.

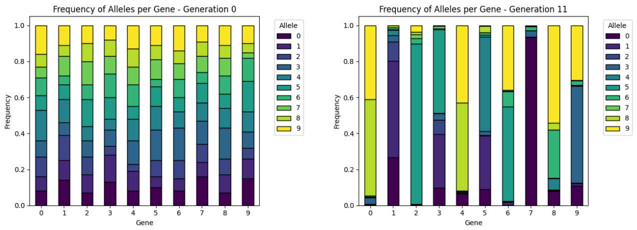

Figure 5: Frequency of each allele in generation 0 compared to generation 11; note the difference in allele distribution throughout the run, and the high frequency of the outermost alleles in alternating genes during generation 11.

## 5. Limitations

It would be unreasonable to think that our simulations accurately portrayed evolution at the scale that it occurs in reality. With the computational power and timeframe given, we were not able to fully capture every nuance of an environment. There are many environmental and species-specific factors that our simulation fails to consider such as environmental carrying capacity, locality of potential mates, cross species mating barriers, and more. Additionally, there are factors in play that affect population dynamics beyond environmental change, crossover, and mutation. This can be seen in Section 4.2 where the prevailing genome was not determined by population, genetic diversity, fitness, or shape and severity of the event modifications, but rather a combination of all of these factors. To derive meaningful comprehensive analysis of questions such as the one presented in 4.2 would require development of new metrics and experiments that simply are out of the scope of the report given the timeline.

## 6. Conclusions

As mentioned in Section 4.1, good evolution thrives off diversity in large population sizes. This combination reduces the negative impact of catastrophic events (and consecutive standard events), leading to a rapidly growing population. As with any algorithm, this GA can be viewed as addressing a biological problem, and a computational one. Biologically, we are attempting to simulate the effects of environmental events on a population of organisms in the hopes of identifying the most likely phenotype(s) to survive, or to gauge the survival of the population overall. Computationally, every experiment boils down to a numbers game; we are setting specific schemata with predefined values in the genome, and applying a shift in the landscape which changes the fitness values of those schemata through events. The events provide a way to target specific schemata, either directly or indirectly, via the frequency and structure of the events. This, of course, is a highly challenging phenomenon to explore and requires painstaking effort in many situations to generate scenarios where populations are both manageable and directed in a way that achieves a desired purpose. This GA offers a novel method of simulating biological evolution utilizing an algorithm that is based on these very principles. The sheer amount of data output from each experiment offers many avenues for analysis and exploration, and the potential impacts in modeling populations reach into a variety of different fields.

## 7. Code Availability

The repository holding all of the code for the simulation as well as scripts to build the tree and create the graphs can be found at the following GitHub link: https://github.com/JustinMorera/CAP-5512-Final-Project

## 8. ACKNOWLEDGEMENTS

A special thanks to one of our teammates, Christian, who went to great lengths to develop an **extremely** useful GUI that allowed us to easily build template files, run tests, and visualize data. **This work is in his honor.** Another shoutout to Justin's dreams for providing this extremely interesting and novel idea.

## REFERENCES

[1] T. scikit-bio development team, “scikit-bio: A Bioinformatics Library for Data Scientists, Students, and Developers,” 2020, [Online]. Available: http://scikit-bio.org/

[2] L. Bejder and B. K. Hall, “Limbs in whales and limblessness in other vertebrates: mechanisms of evolutionary and developmental transformation and loss,” *Evolution & Development*, vol. 4, 2002, [Online]. Available: https://api.semanticscholar.org/CorpusID:8448387

[3] C. M. Donihue *et al.*, “Hurricane effects on Neotropical lizards span geographic and phylogenetic scales,” *Proceedings of the National Academy of Sciences*, vol. 117, pp. 10429–10434, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:216594212

[4] G. D. Ruxton and G. Beauchamp, “The application of genetic algorithms in behavioural ecology, illustrated with a model of anti-predator vigilance.,” *Journal of theoretical biology*, pp. 435–448, 2008, [Online]. Available: https://api.semanticscholar.org/CorpusID:44776496

[5] L. J. Vitt, J. P. Caldwell, P. A. Zani, and T. A. Titus, “The role of habitat shift in the evolution of lizard morphology: evidence from tropical Tropidurus.,” *Proceedings of the National Academy of Sciences of the United States of America*, pp. 3828–3832, 1997, [Online]. Available: https://api.semanticscholar.org/CorpusID:24257525

[6] P. J. A. Cock *et al.*, “Biopython: freely available Python tools for computational molecular biology and bioinformatics,” *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, 2009, doi: 10.1093/bioinformatics/btp163.