universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„A simulation tool to study evolutionary theories of aging“

verfasst von / submitted by

Martin Bagic, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2021 / Vienna 2021

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 875
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Bioinformatik
Betreut von / Supervisor:	Univ.-Prof. Dr. Arndt von Haeseler

Contents

Summary

Senescence is defined as the age-dependent functional decline that results in an age-increasing probability of death. Evolutionary theories of aging are theories that attempt to explain how senescence can evolve, most prominent theories being mutation accumulation theory and antagonistic pleiotropy theory.

This work started with the hypothesis that the demography, as a modulator of random genetic drift, plays a notable role in shaping the life history by intensifying or lessening the effects predicted by mutation accumulation theory and antagonistic pleiotropy theory. More precisely, we hypothesized that mutation accumulation would be stronger in smaller populations than in larger ones, while antagonistic pleiotropy would be stronger in larger populations than in smaller ones.

We opted for a numerical approach by using AEGIS+, an extended version of AEGIS, to simulate various evolutionary scenarios.

We first started by exploring how mutation accumulation interacts with population size to shape life history. The results revealed that the smaller populations consistently evolved shorter life histories and that this is robust with regards to the way the model is run and parameterized and not an artifact of methodology. Furthermore, we discussed the various scenarios demonstrating that genetic drift and mutation accumulation theory are indispensible if one is to systematically explain the evolutionary outcomes of finite populations.

Afterwards, we explored how demography affects antagonistic pleiotropy via modulation of the strength of positive selection. The results indicated that larger populations are more successful at acquiring and maintaining antagonistic-pleiotropic genes. To increase confidence in result robustness, these observations were contextualized within a constructed expectation matrix which was demonstrated to be consistent with the theory of antagonistic pleiotropy. Furthermore, a case of the interaction of MA and AP has been demonstrated and discussed.

Apart from delivering extensive justification for the noteworthiness and theoretical consistency of the effects of demography on forces of mutation accumulation and antagonistic pleiotropy in the evolution of life history, this thesis additionally provides a case for the utility of simulational approaches in studying evolutionary theories of aging which it exemplifies with the AEGIS+ model.

Zusammenfassung

Seneszenz ist definiert als der altersabhängige funktionelle Rückgang, der zu einer mit dem Alter zunehmenden Wahrscheinlichkeit zum Tod führt. Evolutionstheorien des Alterns befassen sich mit der Frage, wie sich Seneszenz entwickeln kann. Die prominentesten Theorien sind die Mutationsakkumulationstheorie (MA) und die antagonistische Pleiotropie Theorie (AP).

Diese Arbeit ging von der Hypothese aus, dass die Demographie als Modulator des zufälligen genetischen Drifts eine bemerkenswerte Rolle bei der Gestaltung der Lebensgeschichte spielt, indem sie die durch die MA and AP vorhergesagten Effekte intensiviert oder abschwächt. Genauer gesagt, stellten wir die Hypothese auf, dass die Mutationsakkumulation in kleineren Populationen stärker ist als in größeren, während die antagonistische Pleiotropie in größeren Populationen stärker ist als in kleineren.

Wir entschieden uns für einen numerischen Ansatz, indem wir AEGIS+, eine erweiterte Version von AEGIS, verwendeten, um verschiedene evolutionäre Szenarien zu simulieren.

Zunächst untersuchten wir, wie die Mutationsakkumulation mit der Populationsgröße interagiert, um die Lebensgeschichte zu formen. Die Ergebnisse zeigten, dass die kleineren Populationen durchweg kürzere Lebensgeschichten entwickelten. Zudem stellte sich heraus, dass dies robust ist in Bezug auf die Art und Weise, wie das Modell ausgeführt und parametrisiert wurde und es sich daher nicht um ein Artefakt der Methodik handelt. Darüber hinaus diskutierten wir die verschiedenen Szenarien, die zeigen, dass der genetische Drift und die MA unverzichtbar sind, wenn man die evolutionären Ergebnisse endlicher Populationen systematisch erklären will.

Anschließend untersuchten wir, wie die Demographie die antagonistische Pleiotropie über die Modulation der Stärke der positiven Selektion beeinflusst. Die Ergebnisse zeigten, dass größere Populationen erfolgreicher sind, antagonistisch-pleiotrope Gene zu erwerben und zu erhalten. Um das Vertrauen in die Robustheit der Ergebnisse zu erhöhen, wurden diese Beobachtungen innerhalb einer konstruierten Erwartungsmatrix kontextualisiert, welche sich als konsistent mit der Theorie der antagonistischen Pleiotropie erwies. Außerdem wurde ein Fall der Interaktion von MA und AP demonstriert und diskutiert.

Abgesehen davon, dass diese Arbeit eine ausführliche Begründung für die beachtenswerte Ergebnisse und theoretische Konsistenz des Einflusses der Demographie auf Mutationsakkumulation und der antagonistischen Pleiotropie in der Evolution der Lebensgeschichte liefert, zeigt sie auch die Nützlichkeit von Simulationsansätzen bei der Untersuchung von evolutionären Theorien des Alterns, die sie mit dem AEGIS+ Modell exemplifiziert.

Acknowledgements

First of all, I would like to thank Dr. Dario Riccardo Valenzano for giving me the opportunity to join his lab to write this thesis but also for his guidance, inspiring discussions and positive energy.

I am also thankful to all the lab colleagues, as well as my university supervisor, Prof. Dr. Arndt von Haeseler, for their assistance in completing this thesis. Finally, I am grateful for the continuous support of my family and friends who helped me retain high spirits in the rough times of the COVID-19 pandemic.

Chapter 1

Introduction

Contents

1.1 Aging

1.1.1 The aging paradox

Aging is the ubiquitous natural and progressive general functional decline leading to an age-dependent increase in the probability of death [1, 2, 3]. It is natural in the sense that it is internally caused and not a consequence of the environmental impacts; it is progressive in the sense that its effects are lasting and steadily worsening; and it is general in the sense that it impacts the whole body thus driving numerous pathologies—those in humans being cancer [4], macular degeneration [5], cognitive impairment [6], Alzheimer's disease [7], Parkinson's disease [8], atherosclerosis [9], benign prostatic hyperplasia [10], stroke [11], diabetes [12], COPD [13], hearing loss [14], and others.

Since aging[1] so profoundly impacts individual fitness by ultimately impairing the abilities to survive and reproduce, its seemingly paradoxical existence poses a kind of an evolutionary riddle—how does this maladaptive phenotype evolve and persist [15, 16]?

1.1.2 Evolutionary theories of aging

Evolutionary theories of aging are theories that attempt to resolve the aging paradox[2]. They can be roughly divided into programmed and non-programmed theories [17]. The fundamental difference between the two types is that programmed theories claim that senescence itself is in fact adaptive (hence no paradox), while non-programmed theories do not make such a claim[3].

Programmed theories generally share the characteristic of invoking group selection theory to explain aging but they differ in the exact benefit to the group that justifies the cost of senescence. The purported utilities range from stabilization of population dynamics (preventing overshoot and population crashes) [20], to preventing spread of disease [21, 22], freeing resources for the younger generations [19], or simply purging the population of the older generations [23]. Some theories arise due to application of the kin selection theory [24, 25] and some criticize the theory of natural selection itself for being too simplistic to explain aging [26, 27]. Though potentially useful in deepening our understanding of the evolution of aging, they are not widely accepted as its ultimate drivers.

Non-programmed theories treat senescence as an evolutionary blind spot or tradeoff. There are three main flavors: mutation accumulation (MA) theory [28], antagonis-

[1] Senescence is another commonly used term for aging. However, unlike the term *aging* which can denote both the abovementioned functional decline and neutral passage of time, *senescence* refers solely to the functional decline. It is also sometimes used as a shorter version of the term *cellular senescence*—a subtype of senescence.

[2] Note that there are fundamentally two types of theories of aging—mechanistic and evolutionary. Mechanistic theories of aging aim to identify and characterize the biological and chemical processes that cause the functional decline, while the evolutionary theories of aging aim to explain how these processes evolve in the first place.

[3] The classification of evolutionary theories of aging as programmed or non-programmed is probably the most fundamental classification as it differs in the application of the natural selection theory but also because it has profound consequences for the development of medical interventions [18][19].

tic pleiotropy (AP) theory [15] and Disposable Soma (DS) theory [29]. These three theories share an important axiom: they recognize that *the force of natural selection declines with age*—i.e., genes with effects felt earlier in life are under stronger evolutionary control than genes with effects felt later in life[4]. However, they run with this axiom in different directions.

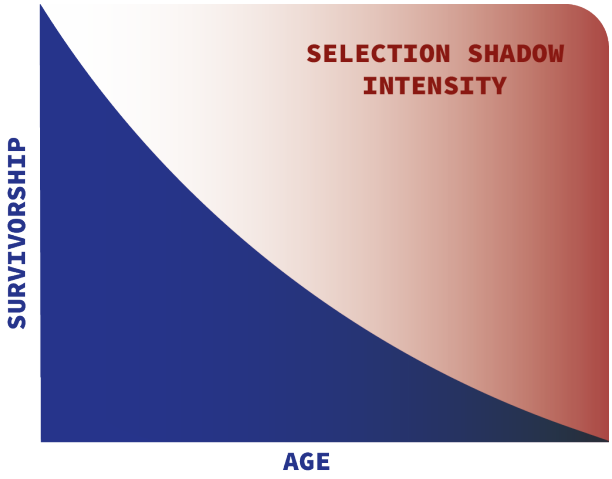


Figure 1.1: *Survivorship declines with age thus increasing the intensity of selection shadow*

The mutation accumulation theory [28, 30, 31] proposes that the declining force of natural selection renders a *selection shadow* over genes whose effects are felt much later in life (or would be theoretically felt by individuals living lives considerably longer than the average individual); that is, there are genes which are barely affected (or completely unaffected) by natural selection, thus they evolve (nearly) neutrally. If deleterious[5] mutations arise in such genes, they will not be purified, rather they will accumulate and cause pathologies.

The antagonistic pleiotropy theory [15] goes in another direction with the shared assumption. Instead of allowing for senescence to develop, it recognizes that senescence can in fact—indirectly—be positively selected[6] given that it is a by-product of otherwise beneficial genes. Thus, AP frames senescence as a postponed tradeoff.

The Disposable Soma theory [29, 32] is mostly a subcase of the AP theory as it

[4]There are multiple lines of reasoning that might bring one to conclude this. Here, I am providing a brief reasoning from the Medawar's lecture that claimed that even genetically immortal populations experience selection shadow: Natural selection becomes weaker with age, not because individuals become more frail, nor because they become less reproductively capable, but simply because a cumulative risk of death creates a demography that greatly values the effects genes produce in early life but hardly values effects they produce in late life. *"[...] the force of natural selection weakens with increasing age—even in a theoretically immortal population, provided only that it is exposed to real hazards of mortality."* [28]

[5]Neutrally evolving deleterious mutation is a bit of an oxymoron from the fitness perspective. Technically, if a mutation is neutrally evolving it is not beneficial nor deleterious. The word deleterious here is used in the sense that it generates a phenotype that would be recognized as disadvantageous had it appeared earlier in life.

[6]Note that being positively selected is not the same as being useful, as was the case in the programmed theories of aging. Whereas in programmed theories of aging, senescence is a standalone positively selected feature, in this case senescent phenotype is still a disadvantageous feature; it is merely hijacking the beneficial gene.

specifies the AP tradeoff as a tradeoff between reproduction and soma maintenance[7]. Limited investment in soma maintenance results in the senescent phenotype which is nonetheless positively selected since its limitation is freeing up resources that can be redirected toward reproduction.

These three theories are very much compatible and complementary, and they offer explanations to the aging paradox by demonstrating how senescence *can be allowed to evolve (by drift)* or *even be positively selected*. They are sometimes referred to as the *general/classical evolutionary theory of aging* as they have garnered the most acceptance in the field [33, 34, 35, 36, 37, 38, 39, 40]. However, one should bear in mind that the MA and AP can produce somewhat different theoretical predictions as senescence evolves by random drift in one, and by positive selection in the other.

1.1.3 Aging models

Up to this point, we have discussed evolutionary theories of aging which attempt to resolve the paradox of why aging even exists. Since they have been proposed, these theories have been used to generate and test hypotheses—an effort that has been ongoing.

Though the support for the non-programmed theory of aging has been consolidating over the years, the contribution of MA versus AP to the senescent phenotype is still very much an open question with inconclusive and sometimes inconsistent evidence [41, 42, 43, 44]. Nevertheless, since the advent of the evolutionary theories of aging, there has been considerable work on aging models as well. In contrast to theories which treat aging philosophically, models attempt to treat it quantitatively in the hopes of acquiring more specific and precise predictions.

Aging models mostly fall into one of the categories: analytical population genetics models or numerical simulations (though there are many other approaches including applications of reliability theory [45, 46, 47] or network theory on aging [48, 49]). Analytical approaches [50, 51, 52, 53, 54, 55, 56, 57] have been quite popular; however, their advancement has proven to be challenging since integrating more evolutionary factors quickly becomes very complicated when using closed-form mathematics. This ultimately leads to a fragmented strategy in the treatment of the subject, wherein a number of models are created such that each model treats a certain set of factors of interest, while other factors are accounted for by auxiliary assumptions. Though useful in certain cases, this approach does not yield a holistic framework and is often distorted by relying on non-biological assumptions.

To address some of these issues, simulational approaches present an opportunity [58, 59, 60, 61, 62, 63, 64]. They are able to integrate a diverse range of factors in one model and do not have to rely on as many non-biological assumptions as analytical models usually do.

[7] Soma maintenance is an umbrella term for all biological processes that bring the organism closer to its homeostatic balance by reverting harmful chemical modifications and restoring metabolic balances.

1.2 The plan

In this thesis, I will extend AEGIS [8], to study in depth a factor that is difficult to treat analytically and has regularly been ignored—random genetic drift[9]. Injecting genetic drift into analytical models requires introducing stochastics; meanwhile, it is naturally present in numerical models as they have finite population sizes due to computational limitations.

My focus will be on the effects of random genetic drift on the two leading evolutionary theories of aging: the mutation accumulation theory, and the antagonistic pleiotropy theory[10]. As random genetic drift is the core controller of the efficiency of natural selection, I will propose the hypothesis that the importance of MA in shaping the life history increases as population sizes shrink, while the importance of AP increases as population sizes grow. This, in effect, introduces population size as a relevant factor in evolution of lifespan—a connection that is non-obvious but should become non-controversial after considering the experiments and the discussion.

Additionally, I will attempt to demonstrate that the simulated evolutionary outcomes are stable and robust though stochastic; that the theory can properly account for the observed evolutionary outcomes; but also that the theoretically predicted behavior emerges from the simple uncontroversial setup of the evolutionary algorithm. This should increase our confidence in both the model and the respective applied evolutionary theory.

[8] AEGIS[65] (Ageing of Evolving Genomes In Silico) is the existing Python package that is extended into AEGIS+ for the purpose of this thesis.

[9] Random genetic drift refers to stochastic evolutionary forces that make populations unable to reach the fitness peaks in fitness landscapes due to random chance affecting life history outcomes. The larger the population, the smaller the drift it experiences. In limit, when population is infinite, no genetic drift occurs.

[10] There has been previous work on the effect of finite populations on the mutation accumulation [66] as well as the consequences thereof [67]; however the considered scenarios were limited, the findings were loosely contextualized in the mutation accumulation theory, and genomic structure was much simpler thus obscuring the age-dependent evolutionary dynamics of survival and reproduction.

Chapter 2

Methods

Contents

2.1 Project structure

In this section, I will present the structure of the code, the outputs it produces when executed, the tool that visualizes its results as well as the instructions on how to run the simulator and the visualization tool.

2.1.1 Code structure

In general, the program modules can be categorized as either *data classes* or *function classes*. Data classes store data about the population (e.g. genomes of individuals) or data about the simulation (simulation parameters and simulation variables), while function classes contain functions that are executed during the simulation (e.g. a function that mutates a genome).

Data classes

This simulator handles many different kinds of data. Using modularization, data are split into four classes:

1. `Pop` class contains population-specific data: genomes, phenotypes, current ages, parental identifiers, personal identifiers, number of births, stage of birth.
2. `Recorder` class contains the same kinds of data as the `Pop` class but of individuals that have been recorded after dying.
3. `Config` class contains some input parameters and derived parameters. Input parameters in this class can either specify population-wide properties of individuals (such as mode of reproduction), or ecological parameters (such as the maximum population size). Derived parameters are values that are not given by the user, but can be computed using the input parameters (e.g. total number of loci in a genome).
4. `Aux` class contains some input parameters and simulation variables[1]. Input parameters in this class specify some simulation-wide properties (such as the number of cycles the simulation will run). Simulation variables are values that are properties of the simulation as a whole, but they change over time (such as the number of stages that have passed).

Function classes

Function classes store simulation functions which can be categorized as containing *core logic* or *diversified logic*.

Core logic encodes the immutable aspects of basic events (aspects of the simulation that cannot be modified by the input parameters, such as the incremental aging of

[1]Splitting input parameters between two classes (`Config` and `Aux`) may seem impractical but there is a clear semantic distinction between the simulation-specific and population-specific parameters. Furthermore, making this distinction lays groundwork for implementing multiple biosystems with unique parameterizations and gene flow since multiple biosystems will share the same simulation-specific parameters while having unique population-specific parameters.

individuals). The functions that execute these events are stored by the `Biosystem` class.

Diversified logic refers to functions that add variability to the core logic. These functions are stored in multiple classes in the `xhelp` module: `Overshoot`, `Interpreter`, `Envmap`, `Phenomap`, and `Season`. Only a subset of these functions from each module is called in a particular simulation, choice of which is determined by the input parameters.

For example, when the population overshoots a predefined ecological capacity, a number of individuals will be eliminated (core logic); however, the logic by which these individuals are chosen is diverse and can be set by the user via input parameters (diversified logic). In this case, the executed function resides in the `Overshoot` class.

2.1.2 Output

Output contains data on dead and recorded individuals as well as some other auxiliary data. Output files are created by the `Recorder` class. There are five kinds of files:

1. `GEN` **files**. These files contain genomes saved in a table format in which each column represents one genomic position (one bit), and each row represents one individual.
2. `PHE` **files**. These files contain phenotypes saved in table format in which each column represents one trait at a certain age, and each row represents one individual.
3. `DEM` **files**. These files contain individual demographic data (such as age at death, cause of death, number of offspring, etc.) in a table format in which each column represents one statistic, and each row represents one individual.
4. `VISOR` **file**. This file records data that are plotted by Visor, which are genomic and phenotypic data of the survival and reproduction loci, as well as data about the number of deaths and causes of death grouped by age. Genomic data comprises of average genome bits (fraction of 1's at each genomic position), while the phenotypic data comprises of population-median phenotype bits (the median value of a trait at each age).
5. `PICKLE` **files**. These files contain the *pickled* class `Biosystem` [2].

File formats

`GEN`, `DEM`, `PHE`, and `PICKLE` files are saved in the *feather* format (which can be read by Python using `pandas` package, or using R), and are saved in the experiment folder. `VISOR` files are saved in the *json* format, and are saved in the Visor folder, as well as the experiment folder.

[2] *Pickling* is a process wherein complete instantiated Python objects are saved in a format that can be reloaded after.

File naming

The stems of the filenames of format `GEN`, `DEM`, and `PHE` denote the index of the file (e.g. first generated `DEM` gets an index 0, the second one gets an index 1, and so forth). The stem of the filenames of format `PICKLE` denotes the stage at which the pickle was generated. The stem of the filename of `VISOR` file is the name of experiment.

File use

The purpose of the `GEN`, `DEM` and `PHE` files is to store high-resolution data about the population at specific times during the evolution. The purpose of the `PICKLE` files is to store a pre-evolved population that can later be used to initialize another simulation. The purpose of the `VISOR` file is to store data that is appropriate for the visualization using `Visor`.

2.1.3 Visor

Visor visualizes compiled statistics that are being updated during a simulation. The statistics contain mean genomes, median phenotypes, as well as death frequencies sorted by cause of death and age. Note that only the Visor profile is depicted in this thesis and explained in detail in this section. More information on other visualized elements can be found in the appendix.

Visor profile

In the main area of the Visor, we can see two genomic matrices and two phenotypic vectors. The order of elements from top to bottom is the following: genomic survival matrix, phenotypic survival vector, phenotypic reproduction vector, and genomic reproduction matrix. The elements are laid out in such a way that values in one column describe genomic and phenotypic values at one age. The totality of these elements is referred to as the *Visor profile* (figure 2.1).

With regards to the genome, the matrix is filled from the top-left to the bottom-right. That is, the columns of matrices represent individual loci so that loci encoding traits at earlier ages are positioned more to the left, while loci encoding traits at later ages are positioned more to the right. Furthermore, each column consists of squares representing bits that are ordered so that more upstream bits are closer to the top, while more downstream loci are closer to the bottom.

Note that the positioning of the bits may or may not be relevant depending on the kind of interpreter used. Some interpreters—such as *decimal*, *switch*, and *exp*—are not position-dependent[3], while some are—such as *binary* and *binary_exp*.

In the case of phenotypic vectors, each square gives the final expressed value of a trait at a certain age (depending on its horizontal position). Similarly to genomic matrices, the values for earlier ages are represented by columns more to the left, while values for later ages are represented by columns more to the right.

[3]Position-dependent interpreters assign different weights to different bits in a locus, while position-independent interpreters assign equal weights to all bits in a locus.

Note that the four blocks are separated horizontally into two parts—left one and right one—representing values before and one after maturity. Since evolution on the loci before and after maturity are quite different, separating them makes visual inspection easier.

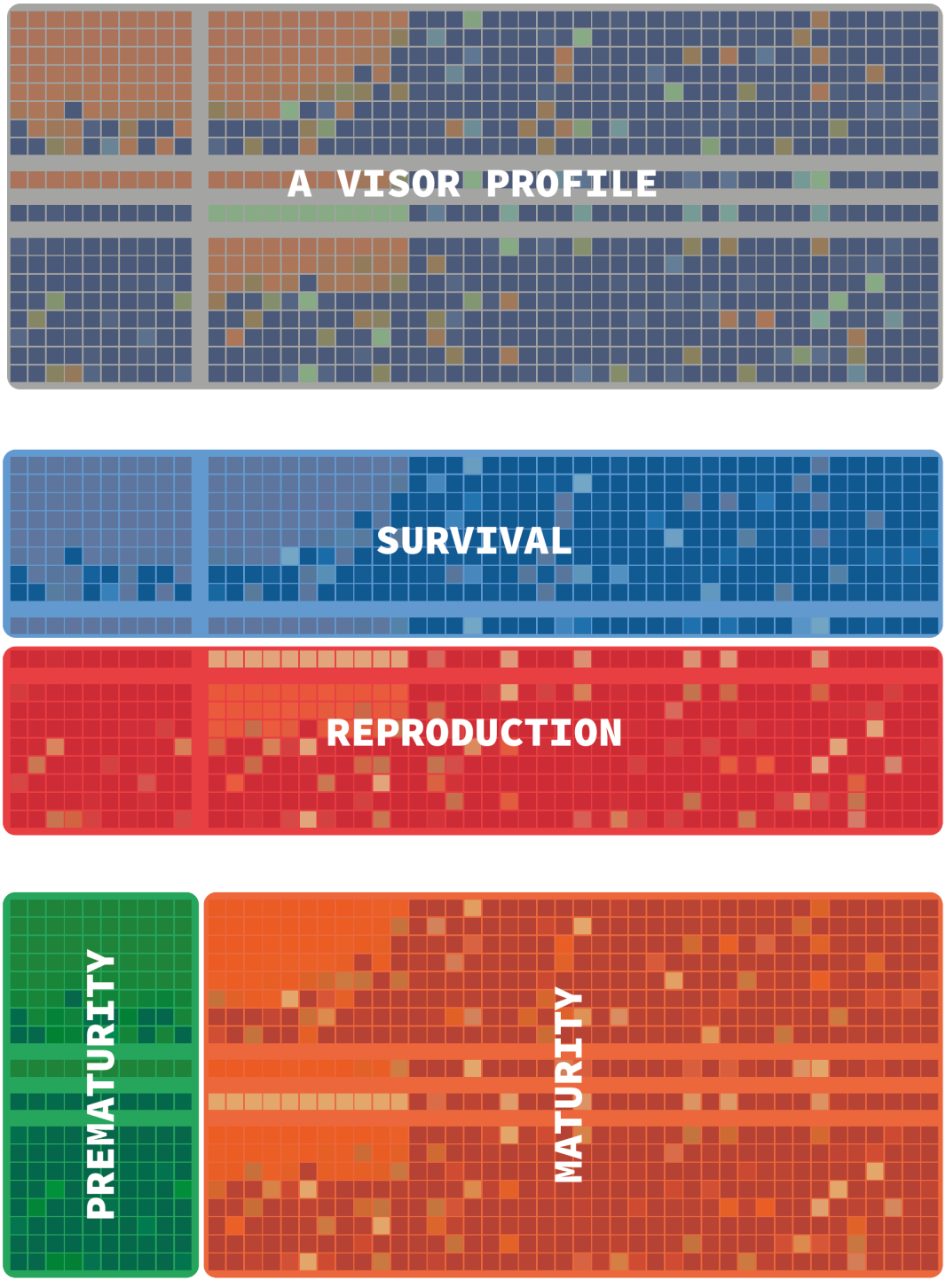


Figure 2.1: *Overview of the Visor profile and its parts*

Figure 2.2: *Color scale encoding bit values; colors are taken at increments of* `+0.05`

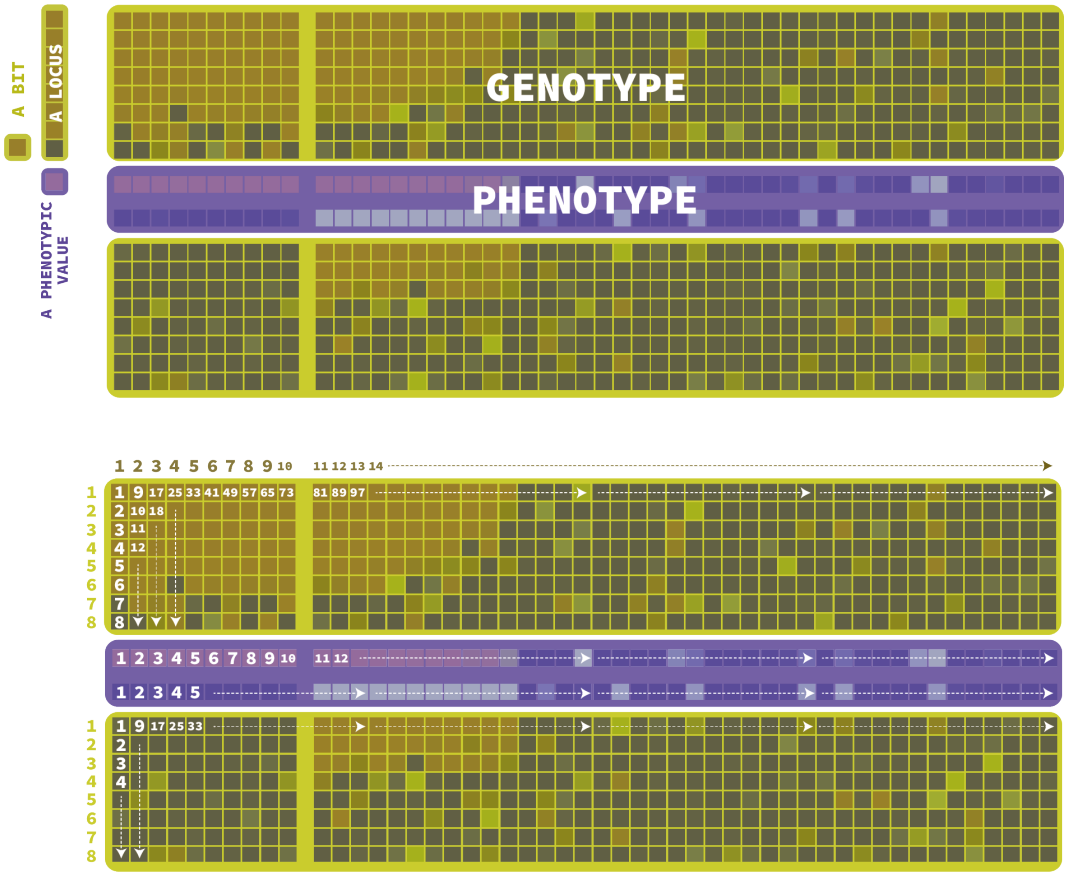


Figure 2.3: *Arrangement of the bit string into the Visor profile*

2.2 Model overview

In this section, I will give an overview of what happens during a simulation, as well as how genomes are encoded and translated into biological traits. This should serve as a high-level blueprint of the model.

2.2.1 Simulation timeline

Each complete simulation consists of three distinct phases: *initialization*, *evolution*, and *termination*.

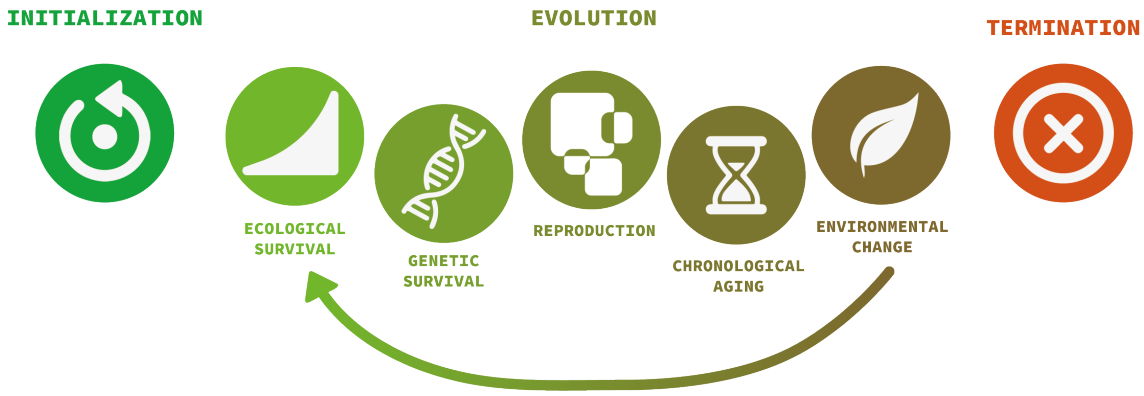


Figure 2.4: *Overview of the simulation timeline*

During *initialization*, the simulator creates wrapper objects that will contain the genomic, phenotypic and demographic data about the population as well as model parameters given by the user.

During *evolution*, the simulator runs the core logic of the model which alters the simulated population over time, i.e. the population evolves. This phase consists of a high number of discrete *stages* which are the smallest time units simulated and represent one complete biological cycle. During each stage, a number of steps are simulated:

1. **Ecological survival**. If the population size is greater than the allowed maximum population size, the overshoot function is activated which eliminates a part of the population. This contributes to *ecological/external* mortality. (Details on population control in section 2.3.4.)
2. **Genetic survival**. Every individual is subjected to potential death with the probability that is encoded by their genomes. This contributes to *genetic/internal* mortality and represents the mortality probability caused by aging.
3. **Reproduction**. Individuals produce offspring with a probability that is genetically encoded. Offspring genomes are generated by mutating and, if reproduction is sexual, assorting and recombining parental genomes. In overlapping populations, the offspring is added immediately to the population. In non-overlapping populations, the offspring is kept separate and at age 0 till the end

of the season (that lasts a predefined number of stages) when all living individuals are eliminated and the population is replenished by the saved offspring. (Details on seasons in section 2.3.5.)

4. **Chronological aging**. Age of all individuals is incremented by one every stage. Individuals with age that surpasses the maximum lifespan are eliminated. This contributes to the *simulational* mortality.

5. **Environmental change**. When enabled, AEGIS+ models the environment in which individuals virtually reside and changes it gradually. When the environment changes, the adaptive values / fitness consequences of genotypes change with it, thus altering the survival and reproduction probabilities of individuals. (Details on environmental maps in section 2.3.1.)

During *termination*, the model tackles auxiliary functions such as creating reports, and saving remaining data to output files.

2.2.2 From genotype to phenotype

Genomes are 2D boolean arrays with one dimension denoting loci, and the other denoting locus components. Phenotypes are 1D real-valued arrays of phenotypic locus values.

When new individuals are spawned, their genomes are translated into phenotypes and saved[4]. This is a complex process comprising of the following steps:

1. **Genomes are contextualized in an environment.**
   This is done by applying the environmental map. This will lead to some genomic bits flipping their values. (Details on environmental maps in section 2.3.1.)

2. **Gene activity is assessed.**
   This is done by applying the interpreter locus-wise. An interpreter will give a real number from an interval $[0, 1]$ which may be understood as the activity level of a gene. Each bit contributes in a way to the activity of a gene; depending on the interpreter, various bits may contribute with different weights. At this stage, we have a vector of values that is referred to as *interpretome* in this text. (Details on interpreters in section 2.3.2.)

3. **Pleiotropic effects are considered.**
   This is done by applying the phenomap on the interpretome obtaining a phenotype. The default phenomap is equivalent to a 1-to-1 map; that is, an interpretome *is* the phenotype. However, when phenomap is modified, some genes may affect functionality of multiple traits. (Details on phenomaps in section 2.3.3.)

4. **Limiting trait variability that is genetically encoded.**
   By default, the genome can fully dictate gene activity (from 0% to 100%). However, there is an option to curb the effect of genome, and to limit the

[4]This contrasts the previous version in which genes were translated ad hoc as they were being activated.

variability onto a subinterval of $[0, 1]$. For instance, there may be a baseline probability of a trait that is dictated by the environment, or a cap.

Note on usage of terms gene, trait, locus and bit

In a non-pleiotropic scenario, each gene is monotropic and each trait is monogenic; that is, there is a bijection between genes and traits, which means that both terms can be very much used interchangeably when discussing this model (figure 2.5).

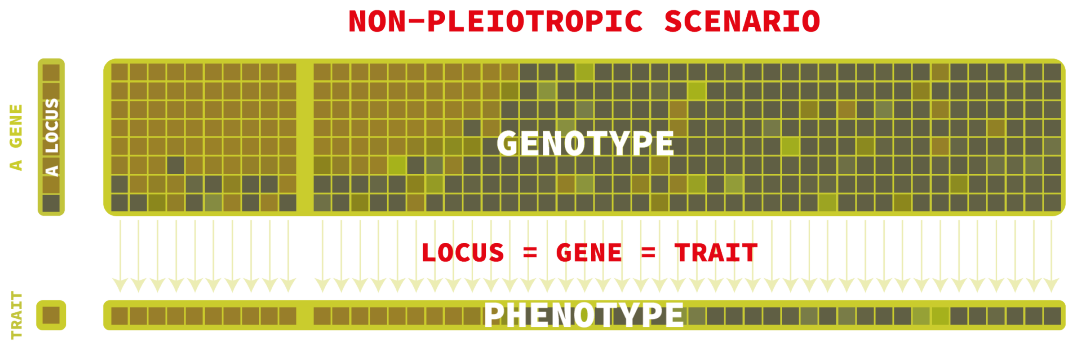


Figure 2.5: *Schematics of the relationship between genes and traits. When genes are monotropic, these two terms are interchangeable.*

However, when pleiotropy is used, this bijection disappears and terms *gene* and *trait* are not interchangeable anymore (figure 2.6). In that scenario, genes are positions in the genome which carry multiple bits with genomic values, while traits are positions in the phenotype which carry one bit respectively with phenotypic values.

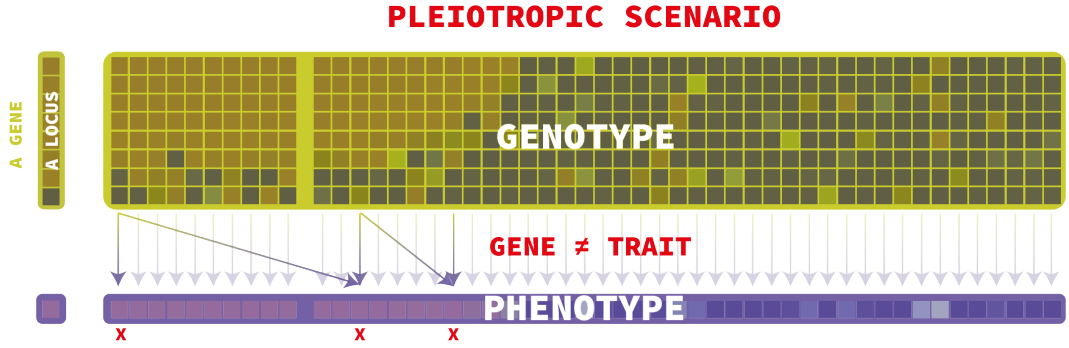


Figure 2.6: *Schematics of the relationship between genes and traits. When genes are pleiotropic, these two terms are not interchangeable since one gene is responsible for multiple trait values.*

Gene-trait equivalence still hold for all loci presented in the figure 2.6 except for the three loci marked with red crosses. The genes at those three loci are not equivalent to the traits at those same three positions as they are either only partially responsible for those respective traits, or they are also responsible for other traits.

In order to reduce the terminological complexity, I will refer to distinct chunks of the genome and the phenotype *loci*, and the components of loci shall be referred to as *bits* regardless of them describing phenotypes or genomes.

2.3 Model mechanics

In this section, I will expand on model overview by providing mechanistic descriptions of diversified logic and lists of customization options.

2.3.1 Envmap

AEGIS+ introduces a new module that can make a static environment dynamic by changing the adaptive value / fitness consequence of genes with time. Biologically speaking, as time passes and environment changes, some genes that were previously adaptive may become maladaptive. This is easily translated into the model by flipping how the model interprets a 0-bit and a 1-bit at a certain genomic position.[5]

Computationally speaking, this can be done quite elegantly using an environmental map (which should be a genome-shaped array) and the `XOR` ($\oplus$) operation. The envmap (environmental map) is initialized as an array containing all zeros, and evolves over time, during which it randomly flips bits from 0 to 1 or 1 to 0. Note that applying $0\oplus$ to a boolean value will return the same value, while applying $1\oplus$ to a boolean value will flip the value.

Currently, the environmental map is not extremely customizable as it only permits changing the evolution rate of the envmap; i.e. the rate at which a bit in the map is flipped. However, the modular design allows straightforward implementation of extended functionality[6]

2.3.2 Interpreter

Interpreters are functions that transform 0/1 bits from a genomic locus into a real number. Applying an interpreter over the whole genome outputs an *interpretome* $\vec{i}$ – a vector of real values describing all of the traits of an individual[7].

Here is a short summary of the main interpreters, their mathematical definitions[8], their visual representations, and the reason for their introduction.

Let $\vec{l} = (l_0, l_1, ..., l_{n-1})$ be a locus vector, a vector of bits of a certain locus, where n is the `BITS_PER_LOCUS` parameter; let i be the interpreted value; and let I be the interpreter function such that $I(\vec{l}) = i$.

[5] Though not pursued in this work, one of the main motivations for implementing the environmental map is enabling the study of mutation rates. Namely, mutation rates are variable across species and are a key piece in understanding evolvability, thus the evolutionary dynamics. AEGIS+ also introduces evolvable mutation rates; however, they only have sense in dynamic environments since their evolution in static environments inevitably leads to minimization and ultimately to rate 0 since mutations become increasingly costly as the population climbs closer to the fitness peak. Making the environment change from time to time gives the reason for the mutation rate to be kept at a certain non-zero level so individuals can continuously adapt to new conditions.

[6] Note that the effect of envmap on the genome is calculated at birth and is not changed throughout the life of a single individual.

[7] If there are no pleiotropic genes, the interpretome values are equal to the phenotypic values.

[8] Note that all given mathematical equations suppose haploidy. In diploidy, the homologous bits are combined using `AND` function before they are interpreted. (Details on ploidy in section 2.3.6.)

Uniform interpreter

$$I(\vec{l}) = \frac{1}{n}\sum_{j=0}^{n-1}\vec{l}_j$$

In words, the probability of an event encoded by a locus is proportional to the number of 1's in the locus.

This was the previously used interpreter as it is conceptually the simplest.

Figure 2.7: *A visual representation of the uniform interpreter. The squares represent ordered bits in one locus, while the pie chart illustrates the contribution of individual bits to the locus value. In uniform interpreters each bit contributes equally to the locus value.*

Binary interpreter

$$I(\vec{l}) = \frac{1}{2^n - 1}\sum_{j=0}^{n-1} 2^{n-1-j}\cdot\vec{l}_j$$

In words, this interprets the locus as a binary number with digits previously reversed in order, which is then weighted by the biggest binary number with n bits.

This interpreter has been introduced to assign growing weights to individual bits, in contrast to the *uniform* interpreter for which all bits contribute equally. Assigning weights has two beneficial consequences: firstly, evolution will unfold faster (that is the population will approach the final outcome faster) as more phenotypic change can be achieved with fewer mutations; secondly, when bits have different selection coefficients, we can directly observe the selection strength (not just of loci but of bits as well). Most simulations in this thesis use this interpreter.

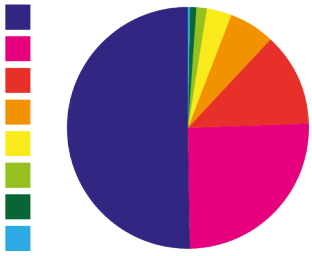


Figure 2.8: *A visual representation of the binary interpreter. The squares represent ordered bits in one locus, while the pie chart illustrates the contribution of individual bits to the locus value. In binary interpreters each bit contributes to the locus value half the amount than the previous bit.*

Switch interpreter

$$I(\vec{l}) = \begin{cases} 0, & \sum \vec{l} = 0, \\ 1, & \sum \vec{l} = n, \\ 0 \text{ or } 1 \text{ with equal probability}, & \text{otherwise} \end{cases}$$

In words, this interpreter returns 1 if all bits equal 1; it returns 0 if all bits are 0; and it returns either 0 or 1 otherwise (with equal probability).

This interpreter has been introduced as a tool for detecting evolutionary forces (positive selection, purifying selection, and neutral evolution). As there is only one state that results in high locus function (thus indicating positive selection) and only one state that results in absence of locus function (thus indicating purifying selection), when qualitative statements are to be made about presence of positive and purifying selection in loci, this interpreter is the best choice.

This interpreter is best used with a relatively small `BITS_PER_LOCUS` since evolution in the *otherwise* case is essentially a random walk—the *otherwise* case is a case that a locus can easily get into (by a single mutation) but it is difficult to get out of. Preferably this number is 2, 3, or 4. Lower numbers strengthen the sensitivity to selective forces and improve evolvability, but make neutrality harder to recognize. Higher numbers make neutrality easier to recognize, but make maintaining and attaining positive or purifying selection more difficult.

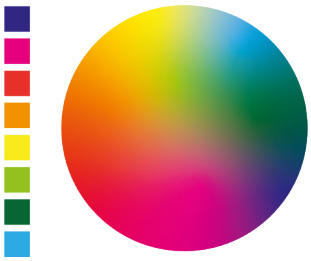


Figure 2.9: *A visual representation of the switch interpreter. The squares represent ordered bits in one locus, while the pie chart represents the locus value. In switch interpreters the locus value is not a sum of individual contributions but depends on the values of all bits.*

Exponential interpreter

$$I(\vec{l}) = 2^{n - \sum_{j=0}^{n-1} \vec{l}_j}$$

In words, exponential interpreter cuts the probability of an event in half with each zero symbol present in the locus.

This interpreter is used for the mutation loci as its range spans many orders of magnitude.

Binary switch interpreter

This interpreter is a hybrid of the *binary* and *switch* interpreters. If the last bit in the locus is 0, the returned values is always 0 (i.e. the gene is turned off). However, if the

last bit is 1, the returned value is the value that would be returned when computing the binary value of $(\vec{l}_0, \vec{l}_1, ..., \vec{l}_{n-2})$, that is the locus without the last bit.

The switch part of the interpreter gives a clue about the selection status of the locus, while the binary part provides some sense of how strong the selection might be.

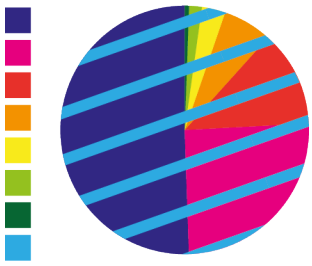


Figure 2.10: *A visual representation of the binary switch interpreter. The squares represent ordered bits in one locus, while the pie chart illustrates the contribution of individual bits to the locus value. In binary interpreters each bit contributes to the locus value half the amount than the previous bit except the last bit which determines whether the locus will have value 0 or the value determined by other bits.*

2.3.3 Phenomap

This class introduces pleiotropy (ability of a gene to affect multiple traits) which is used in the second part of the thesis in the work on antagonistic pleiotropy.

Mathematically, pleiotropy is implemented as a matrix-vector multiplication of the phenomap matrix P and an interpretome $\vec{i}$. Phenomap matrix is a matrix of the form $P_{a,p}$ where locus a affects the locus p. Thus, the locus a is pleiotropic as it affects the trait encoded by the locus a as well as the trait encoded by the locus p. (Note that the trait can be the same—e.g. survival—but the age of expression different.) Pleiotropy is effectively non-existent if and only if phenomap is an identity matrix (which is its default value).

Note that values in the phenomap can be set to any real value, even outside of the [-1,+1] interval. This is because a small functionality of the locus a may have a disproportionately big effect on the locus p. To prevent probabilities from leaking outside of the [0, 1] range, a normalization is applied after the matrix-vector multiplication.

Pleiotropic effects can be added to the identity matrix by modifying the `PHENOMAP_PLUS` parameter. This parameter is an array of triples of the form $T = (a, p, m)$, where the a is the index of the *active locus* (the locus of the pleiotropic gene), p is the index of the *passive locus* (the affected trait), and m is the *magnitude* (the phenomap multiplicative factor).

When triples are added to the `PHENOMAP_PLUS` parameter, phenomap matrix P is modified in such a way that $P_{T_a,T_p} = T_m$ for every triple. Remember that $P_{n,n} = 1$ for every n by default.

For example, given the two interpretome values $\vec{i}_1 = 0.7$ and $\vec{i}_{10} = 0.4$ and a phenomap triple $T = (1, 10, -0.2)$, the phenotypic value of locus 1 is still 0.7, but the phenotypic value of locus 10 is 0.28 (because $0.4 - 0.14 = 0.28$).

Encoding AP genes

AP genes are genes that have both beneficial and detrimental effects on fitness.

To encode AP genes we need to modulate four fitness factors of the gene effects—magnitude of the beneficial effect, magnitude of the detrimental effect, reproductive relevance of the beneficial effect[9], and the reproductive relevance of the detrimental effect. I will refer to those four factors using the following symbols respectively: m_+, m_-, p_+, p_-.

The simplest way of encoding AP genes then is to add a detrimental effect to a locus by adding a phenomap triple $[a, b, m_{ab}]$ with a negative m_{ab}. This practically creates an AP gene encoded at locus a with a detrimental effect at locus b (and the default beneficial effect at locus a due to the default implied phenomap triple $[b, b, 1]$).

However, this approach has a major limitation—we can only specify the magnitude of the detriment; we cannot specify the magnitude of the benefit (it is 1). However, AP genes usually merely improve some traits while harming some other traits, which is what we want to explore.

A way to lift this limitation is to use the help of another locus; thus encoding AP genes usually requires encoding pairs of pleiotropic loci. First we add a phenomap triple as described: $[a, b, m_{ab}]$, where $m_{ab} < 0$. This triple encodes for the detrimental effect of the AP gene situated at locus a, with the effect felt at locus b. Next, we add another phenomap triple of form $[a, a, m_{aa}]$, which overrides the default $[a, a, 1]$ thus modulating the magnitude of the beneficial effect of the AP gene. Per default, the m_{aa} is set to 1, but we can set it to any number from $[0, 1]$. Finally, we add the third phenomap triple wherein we make use of another locus c to fill up the opened gap in the locus a, namely $[c, a, 1 - m_{aa}]$. This represents the basal level of the trait a without the AP gene at locus a. This way the trait a can obtain full 100% functionality with the contributions from AP locus a and the basal locus c[10].

This technique will be used throughout the AP section of this work (section 3.2). To make talking about these AP genes easier, we will use a shorthand of the form $(a_{m_{ca}}, b_{m_{ab}})$ such that a and b are the affected loci whose phenotypic values are modified by magnitudes m_{ca} and m_{ab}.

2.3.4 Overshoot

When population overshoots, the model eliminates a number of individuals [11]. There are many conceivable ways by which the model can choose which individuals are to be eliminated. AEGIS+ introduces a module that enables different choice regimes.

[9] Reproductive relevance is the cumulative reproductive probability of the portion of lifespan affected by the AP gene.

[10] It should be noted that giving the locus c an additional beneficial effect will increase its importance and make it more likely to be selected. This is important to consider if we are interested in the selection status of that locus. Also, the basal locus should be a strongly positively selected locus if we wish to guarantee the baseline value of locus a. Thus, c should be set at 0 per default, though modifications can always be made.

[11] Population control is necessary in this model to keep the simulation tractable, but it is also useful as it provides a source of extrinsic mortality which is core in testing the hypotheses of interest.

Currently, `Overshoot` module offers five different choice regimes. Note that `starvation` and `cliff` lead to significant fluctuations of the population size, while `treadmill` regimes keep the population size stable.

`starvation`

Starvation represents the biologically realistic scenario and is set as the default option. While population size is under the limit, no external mortality is applied. Once the population size exceeds the limit, some external mortality is applied in the sense that the survival of all individuals is scaled down by a common factor. However, it is usually too low to bring the population under the limit in one stage. As simulation proceeds and starvation continues, the external mortality rate is increased exponentially, until the population is under control. (Note that the resulting population size can, and often is, considerably lower than the ecological limit since an overcorrection is commited due to lag.) Next time starvation is again triggered, the external mortality rate is again at the initially low value.

`treadmill_random`

Whenever population overshoots, treadmill regimes remove the exact number of individuals needed to bring the population size just under the ecological limit, and they do it in one stage. There are three flavors of the treadmill regimes and they differ in the logic by which they choose which individuals are to be eliminated.

Random treadmill choses individuals randomly.

In effect, this is a version of starvation that corrects instantaneously and does that precisely (without overcorrection).

`treadmill_boomer`

Boomer treadmill is a type of the treadmill regime that eliminates the oldest individuals.

`treadmill_zoomer`

Zoomer treadmill is a type of the treadmill regime that eliminates the youngest individuals.

`cliff`

Whenever population overshoots, cliff regime brings the population size down to a level that is a fraction of the maximum population size, and it does so in one stage.

In effect, this is a version of the random treadmill that commits an overcorrection.

2.3.5 Season

This is a new module that introduces *seasons*—fixed periods during which all newly added individuals are saved as *eggs* so they do not age nor contribute to reproduction until they are hatched at the beginning of a new season. This is an abstraction of

the concept of annual species which are able to reproduce only during a brief season, when resources are favourable.

Adding seasons has profound consequences on the external mortality and reproductive dynamics. Currently, seasonality is not very customizable—there are no other parameters to be set apart from the season duration. Generally, it makes sense to set the season duration to be equal to or less than the maximum lifespan. If the season duration were longer than the maximum lifespan, there would be idle stages with only eggs waiting to hatch and no alive individuals.

2.3.6 Ploidy

All sexually reproducing populations are diploid, while the asexually reproducing populations can either be diploid or haploid. Ploidy plays a role in recombination, assortment and interpretation.

In the case of diploidy, before interpretome is calculated, loci are partitioned into adjacent pairs of bits and those pairs are collapsed using the `AND` operation—that is, homologous bits need to be 1's for the resulting combined bit to be 1, otherwise the resulting bit is 0. This makes bits behave like recessive genes.

Asexually reproducing populations can be either diploid or haploid. When diploid, bits undergo the identical collapsing treatment as just described. When haploid, no such collapsing takes place, and every bit is evaluated independently. Note that in the case of asexual reproduction, there is no recombination nor assortment, so the only difference is in the interpretation.

Furthermore, in AEGIS+ there is no explicit encoding of chromosomes; rather they exist implicitly. In both cases—diploidy and haploidy—the genomes look exactly the same, but their bits treated differently. Namely, in the case of diploidy, all bits with odd positions belong to one chromosome, while the bits with even positions belong to the other chromosome.

2.3.7 Parameters

- `MAX_POPULATION_SIZE` is the maximum population size which does not trigger the overshoot event.
- `OVERSHOOT_EVENT` is the type of overshoot event that is executed when max population size is exceeded.
- `CLIFF_SURVIVORSHIP` is the fraction of the population that survives the cliff overshoot event.
- `MAX_LIFESPAN` is the age that is not attainable, i.e. at which every individual is forced to die.
- `MATURATION_AGE` is the age at which an individual can reproduce.
- `BITS_PER_LOCUS` is the number of bits that are found in each locus.
- `HEADSUP` is the parameter that can be used to set early-life survival and reproduction independently of the whole genome. If it is set to -1, there is no effect.

If it is set to 0, bits of all immature loci are set to 1. If it is set to a number X that is greater than 0, then bits of all immature loci as well as the first X mature loci are set to 1.

- `GENOME_STRUCT` is the most complex parameter. It encodes the structure of the genome. It consists of key→list items, such that the key denotes the trait, and the list contains settings for that trait. The list itself consists of exactly five parameters:

  1. Age variability. This parameter is a boolean. If True, this trait will change value with age, and it requires many loci (one for each age). If False, this trait will not change with age (so it is encoded by one locus only) but it is still individual-specific and evolvable.
  2. Intepreter kind. This parameter must be a name of one of the many possible interpreters.
  3. Lower bound. This is the minimum effective value of a trait.
  4. Upper bound. This is the maximum effective value of a trait.
  5. Initial value. This is the fraction of bits that will be initialized as 1's in all of the loci that encode for a particular trait.

  Currently, there are four traits that can be evolvable: survival, reproduction, mutation and neutral[12]. If some of these traits are not present in the `GENOME_STRUCT`, they should be present in the `GENOME_CONST`.

- `GENOME_CONST` contains a mapping of traits onto values. These values are the constant values that do not evolve, nor are individual specific.
- `REPR_MODE` is the reproduction mode of individuals. It can be asexual, sexual or *diasexual*. Diasexual is diploid asexual, and sexual is like asexual but diploid and with recombination and assortment.
- `MUTATION_RATIO` is a fraction which, when multiplied with the mutation probability, gives the probability that a beneficial mutation (0→1) happens.
- `CYCLE_NUM` is the number of stages that are simulated before the simulation is stopped.
- `LOGGING_RATE` is the number of stages which pass between each log message in the terminal being printed during simulation.
- `REC_EVERY_NTH` is the number of individuals that have to die in order for one individual to be stored in the records.
- `FLUSH_RATE` is the number of records that have to be accumulated before they are written out to files.
- `JSON_RATE` is the number of stages which have to pass before another `JSON` file is saved.

[12]Neutral loci have no biological effect on the simulated individuals, so they serve as control loci which display perfectly neutral evolution.

- `REC_RATE` is the number of `JSON` files that have to be saved before other population data files are saved (such as `GEN`, `PHE` and `DEM`).
- `PICKLE_RATE` is the number of stages that have to pass before the living population is pickled (saved in a way that enables loading the population later).
- `PHENOMAP_PLUS` is the list of all additional phenomap triples.
- `ENVMAP_RATE` is the number of stages that have to pass before a bit is flipped in the environmental map.

2.4 How to use

How to install

AEGIS+ requires no installation. Instead, the code is to be downloaded from the git repository. The code can be run using Python3, and it requires following packages: numpy, pandas, PyYAML.

How to configure

To customize parameter values, the user has two options: they can either create a new configuration file in the config folder, or they can add the values in the command line when starting the program. Usually, if a batch of experiments is to be run, a new configuration file would be created with the parameters that are common to all experiments in the batch, and then the unique parameter values would be given through the command line.

How to run

A simulation can be run easily by entering a command of the form
`python3 main.py [PARAMS] [NAME] [DIR] -c [YAMLs] -r [PICKLE]`

- `PARAMS` is a JSON-formatted string that contains key:value data about parameters that should overwrite the parameters given in the config files.
- `NAME` is the name of the particular simulation.
- `DIR` is the name of the experiment directory. One directory can contain multiple differently named simulations.
- `YAMLs` is the name of the config file that contains customized parameter values. Per default, parameter values are taken from `config_default.yml`, and if no config file is specified, these will be the final simulation values; however, if a config file is specified, the parameters in that file will overwrite the default parameters. If multiple config files are specified, every subsequent config file will overwrite the parameters from the previous files.
- `PICKLE` is the path to the existing `PICKLE` file that contains a preevolved population. If not specified, a fresh population will be initialized.

An example of a valid command is: `python3 main.py '{"MAX_POPULATION_SIZE":200}' neutral2000_to_200 final -c upvolution.yml -r final/up2000/200000.pickle`

How to visualize using Visor

Open a terminal in the `Visor` folder in the project folder. Run the command

`python3 -m http.server.`

This is important since `VISOR` files need to be served in order for Javascript to have access to them. Go to the URL that the Python3 is serving.

Chapter 3

Results

Contents

3.1 Mutation accumulation

Due to the age-dependent decline in the force of natural selection, mutation accumulation theory [28] predicts that genes with early-life effects should be under strong selection control, while genes with late-life effects should tend to evolve more neutrally.

As smaller populations experience stronger genetic drift, thus weakening the force of natural selection, we expect that more genes succumb to neutral evolution than in the larger populations, resulting in a shrinking population undergoing a shrinking of lifespan.

In this section, I will demonstrate that larger populations evolve longer lifespans (section 3.1.1), that the observed differences are not due to differential evolutionary dynamics but due to expected long-term outcomes (section 3.1.2), and that these outcomes are robust with regards to the parameterization of the model (section 3.1.3).

3.1.1 Larger populations evolve longer lifespans

The most straightforward way to investigate any differences between the evolution of life history in a small and a big population is to simply run two simulations that are identically parameterized apart from the parameter of interest—population size.

In this section, two simulations have been run with the default parameterization and with population sizes of 200 and 20000 respectively. (Here, the binary interpreter is used[1].)

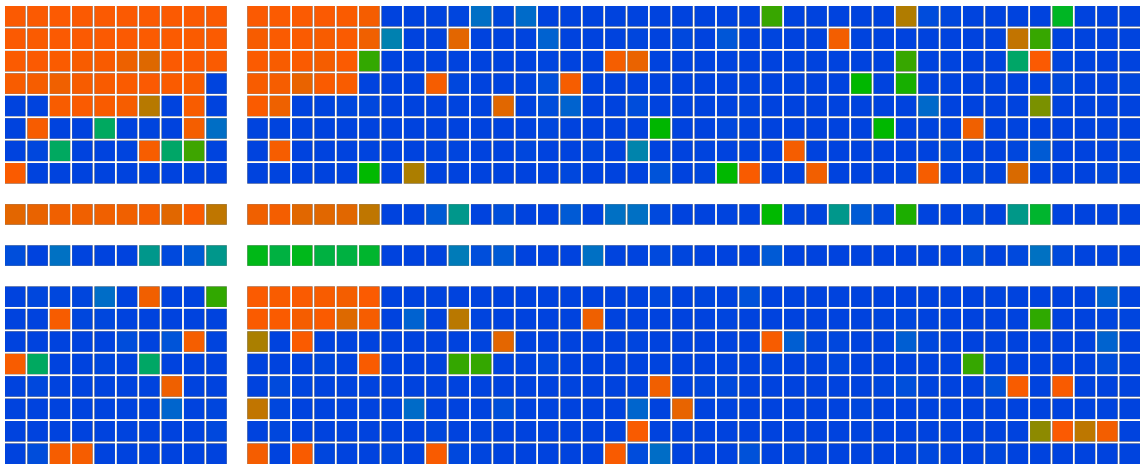


Figure 3.1: *Visor profile of the final smaller population (N=200) evolving under default parameterization*

Looking at the Visor profile of the smaller population (figure 3.1), it is apparent that all ten immature survival loci, first six subsequent mature survival loci, as well as the first six mature reproduction loci are positively selected. Most of these loci have multiple high bit values that occupy the high-weight positions within the loci.

[1] When binary interpreters are used, bit weightings are approximately halved by each subsequent bit. More details on binary interpreters can be found in section 2.3.2.

Furthermore, notice that the selection strength across the immature survival loci is similar (since the number and position of high-value bits are similar), while the selection strength of mature survival loci decreases with age (since more and more low-weight bits are declining in value). The rest of the loci seem to be evolving neutrally as they mostly have low-value bits with some randomly positioned high-value bits.

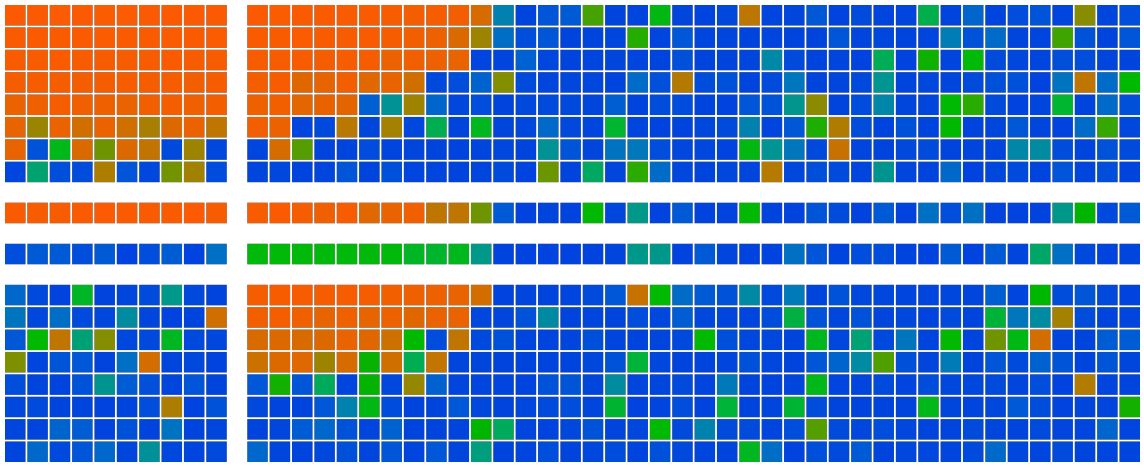


Figure 3.2: *Visor profile of the final larger population (N=20000) evolving under default parameterization*

When we apply the same analytical eye to the simulation of the larger population (figure 3.2), it is obvious that all ten immature loci, first eleven mature survival loci, as well as the first eleven mature reproduction loci are selected. Similarly to the previous population, the selection strength of immature survival loci is similar, but it decreases with age for the mature survival loci. The same seems to be true for the reproduction loci too. Other loci seem to be evolving neutrally.

Differences between the two populations

The difference in evolutionary outcomes for the two populations are apparent on two levels: locus and bit level.

On the locus level, the larger population holds more mature survival and reproduction loci under positive selection (eleven) than the smaller population (six).

On the bit level, the larger population positively selects more bits of those positively selected loci than the smaller population. This is true both for mature survival and mature reproduction loci, as well as for immature survival loci. For the immature survival loci the number of selected bits is 6–8 for the larger population, while it is 4–6 for the smaller population. For the mature survival and reproduction loci, we see that both populations experience a decline in selection; however, this decline starts later for the larger population and with higher initial number of positively selected bits.

In conclusion, results suggest that larger populations evolve longer life histories with higher survival rates at each age. This might be because larger population are less sensitive to stochastic events that modify its genetics.

3.1.2 Addressing the runtime bias

One of the fundamental problems with evolutionary algorithms is knowing when to terminate them [68, 69]. This is especially pressing if we are interested in comparing evolutionary outcomes of multiple simulations or if we simply want to observe long-term stable evolutionary outcomes (such as in the classical case of the *traveling salesman problem* [70]).

If we observe differences between two simulations, how can we know that they are not merely due to transient kinetic effects and that they will not, in fact, eventually converge? Also, more generally, how do we know if a single simulation has converged or is close to converging?

Here, I present a technique for quantifying the expected range of the long-term evolutionary outcomes. This technique is not a termination condition, per se. Instead it informs about the degree of completion of a simulation. However, there has been work put into finding a termination condition which is presented and discussed in the appendix.

Assumption of evolutionary order and monotonicity

Before explaining the techniques, it is important to make an observation that is valid in this model and serves an important logical support—and that is that evolution progresses monotonically.

In essence, that means that there is an inherent order to selection and that evolution proceeds respecting that order and not changing its direction—i.e. monotonically.

The existence of the inherent order is a consequence of the monotonic decline in the selection coefficients of mature survival and reproduction loci. Immature survival loci are first to be positively selected, then the mature survival and reproduction loci starting with the youngest mature ages and progressing towards the oldest mature ages.

The direction of evolution refers to the step-wise increase or decrease in the number of selected loci. Finally, monotonicity refers to the evolution retaining its direction; that is, it should not initially increase the number of positively selected loci, and then later start decreasing it.

Obviously, there is some randomness to the evolutionary algorithm (propped up by genetic drift for that matter) which will result in some minor variation of that direction on the smaller time scale. However, the general trend should be stable and monotonic. (Once the long-term outcome is achieved, no significant increase or decrease should be observed anymore.)

Bounding the evolutionary outcome

The genetic makeup of the initial population is important as it is echoed throughout the simulation. In fact, the initialization signal should be interpreted as a bias which is completely eliminated once the simulation reaches its long-term outcome.

However, since reaching the long-term outcome can take a long time and we can never be sure it has been reached, the initialization bias can never quite be eliminated. This

technique offers a workaround.

Assuming evolutionary order (section 3.1.2), one can construct two populations with *opposite* initialization signals: an *optimistic population* in which only loss of positively selected loci should occur under any imaginable circumstances, and a *pessimistic population* in which only gain of positively selected loci should occur under any circumstances other than extinction. Assuming evolutionary order and monotonicity, the evolved *optimistic population* should inform us about the maximum fitness long-term evolutionary outcome that we can expect, while the evolved *pessimistic population* should provide information on the expected minimum-fitness long-term outcome. Taken together, we have constructed an interval, a subset of the evolutionary order, which contains the long-term evolutionary outcome. The longer the simulations run, the narrower the interval is[2].

Let us apply this technique to our experiment.

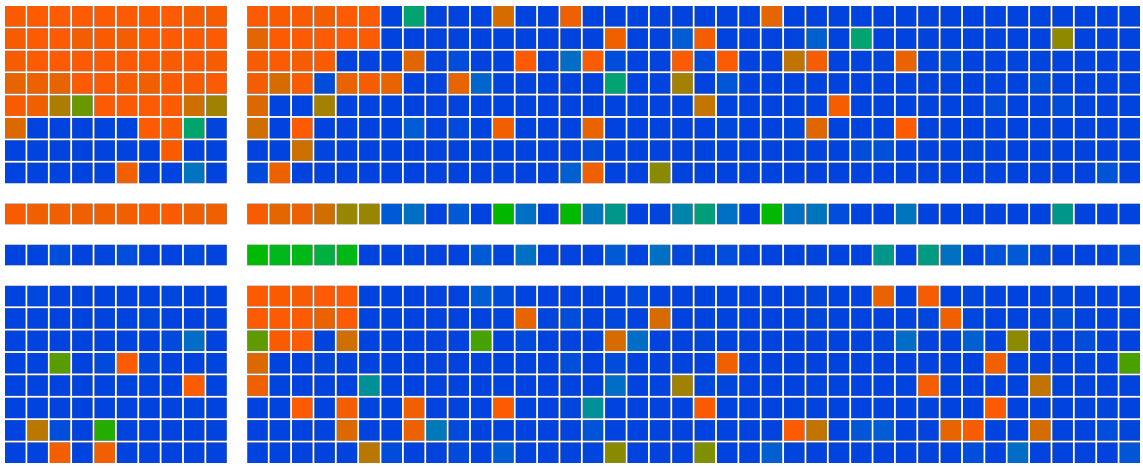


Figure 3.3: *Most pessimistic scenario of the Visor profile of the final smaller population (N=200) evolved under default parameterization*

Looking at the Visor profile of the pessimistic scenario for the smaller population (figure 3.3), it is obvious that the first six mature survival loci and first five mature reproduction loci are selected. In the most optimistic scenario for the smaller population (figure 3.1) these numbers were six and six. Thus, we conclude that the final converging genome should positively select six mature survival loci, and five to six mature reproduction loci.

[2]Note that, even when not reported, this technique is used throughout the thesis to verify the evolutionary outcomes.

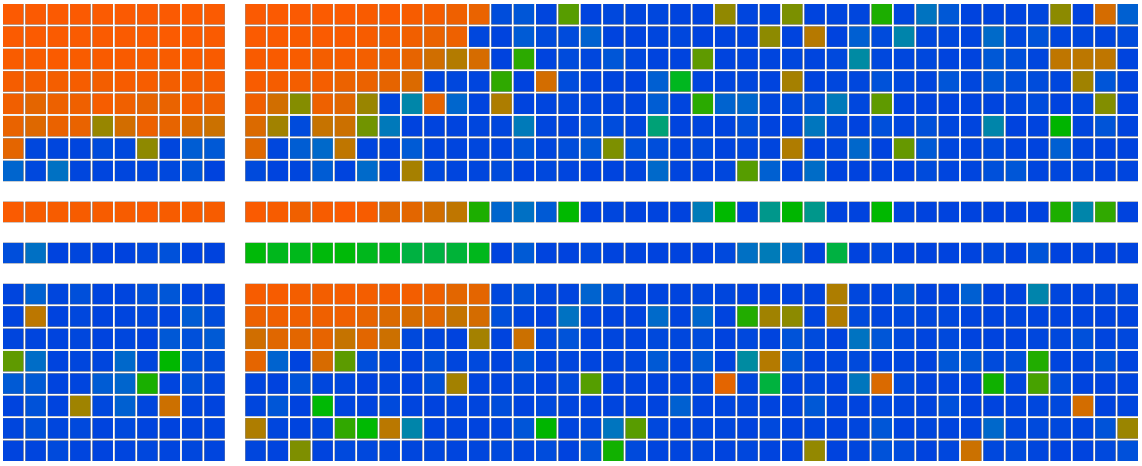


Figure 3.4: *Most pessimistic scenario of the Visor profile of the final larger population (N=20000) evolved under default parameterization*

Looking at the Visor profile of the most pessimistic scenario for the larger population (figure 3.4), it is obvious that the first eleven mature survival loci and first eleven mature reproduction loci are selected. In the most optimistic scenario for the larger population (figure 3.2) these numbers were eleven and eleven. Thus, we conclude that the final converging genome should positively select eleven mature survival loci, and eleven mature reproduction loci.

Using this method, we have demonstrated that the primary experiments have not been overly optimistic but very representative as they have yielded similar results to those of the secondary experiments; thus, any transient effects have not affected much the observed outcomes. Moreover, this further confirms the hypothesis that larger populations evolve longer lifespans than shorter populations.

3.1.3 Addressing parameterization bias

Presented results indicate that larger populations evolve longer lifespans than smaller populations; but could this be a mere consequence of the simulation parameters we have chosen? In other words, are these results robust with respect to the parameterization?

In order to test the result robustness, we can vary the parameters and verify if the observation still holds. Here, we present one-dimensional experiment batteries where all parameters are held at default values while one parameter is varied over many values in a sizeable interval[3]. This may help increase our confidence that the observation made is valid generally, regardless of the chosen parameters. Futhermore, it will provide us with an opportunity to test the theoretical expectations.

To perform this analysis, we are testing 16 different parameters with 5+ population

[3]It should be noted that there is a theoretical possibility that the observed behavior is broken with a particular combination of parameters—mathematically speaking, we are only looking at lines in a multidimensional parameter space and confirming our observation, but there is a vast unexplored rest of the parameter space in some part of which the observation might not hold. Using biological intuition, we conclude that this is very unlikely and, if possible, it probably encodes a rare or a non-biological scenario.

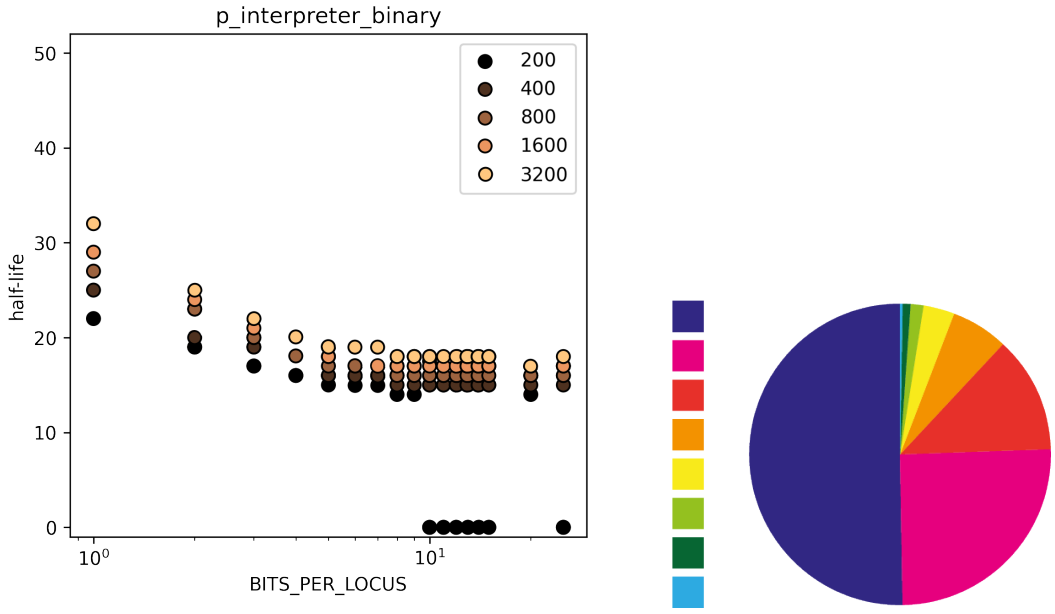


Figure 3.5: *Half-lives of the populations evolved under a binary interpreter and a varied number of bits per locus (a) and the schematic of the binary interpreter (b)*

sizes (between 200 and 3200) and 10–15 different parameter values. All simulations run for two million stages. Once simulations are finished, we use the median phenotypes from `VISOR` files to calculate *half-life* which we define as the earliest age at which the cumulative survival probability dips below 50% (i.e. an individual is more likely to be dead than alive)[4]. We plot half-lives against the varied parameter values using color to distinguish between population sizes (lighter shades represent larger populations, while darker shades represent smaller ones). If it is true that larger populations always develop longer or equally long lifespans as smaller populations, we should observe lighter points always sit more north from darker points.

Binary interpreter

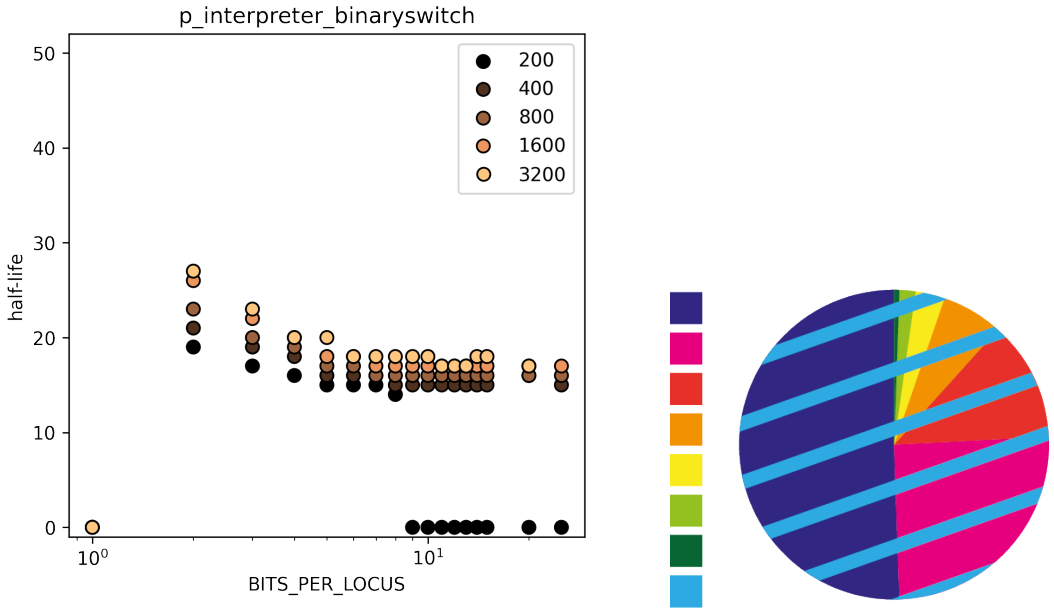
This experiment explores how half-life is affected by the locus granularity of *binary*-interpreted loci.

Half-life is highest when loci consist of a single bit; i.e. when `BITS_PER_LOCUS` (BPL) is set to 1. It decreases as BPL increases from 1 to 8, after which it stabilizes. Some extinction occurs to the smallest populations for BPL greater than 9.

Binary switch interpreter

This experiment explores how half-life is affected by the locus granularity of *binary switch*-interpreted loci. Half-life is highest when loci consist of two bits; i.e. when BPL is set to 2. Similarly to the previous case with the binary interpreter, half-life initially increases (up to BPL value of 7) after which it remains invariant. Also, some extinction occurs in this case as well, after BPL surpasses values greater than 9. In

[4]It is difficult to summarize a life history trajectory using one number. Half-life has been chosen because it represents a sort of an inflection point in life history.



(a) (b)

Figure 3.6: *Half-lives of the populations evolved under a binary switch interpreter and a varied number of bits per locus (a) and the schematic of the binary switch interpreter (b)*

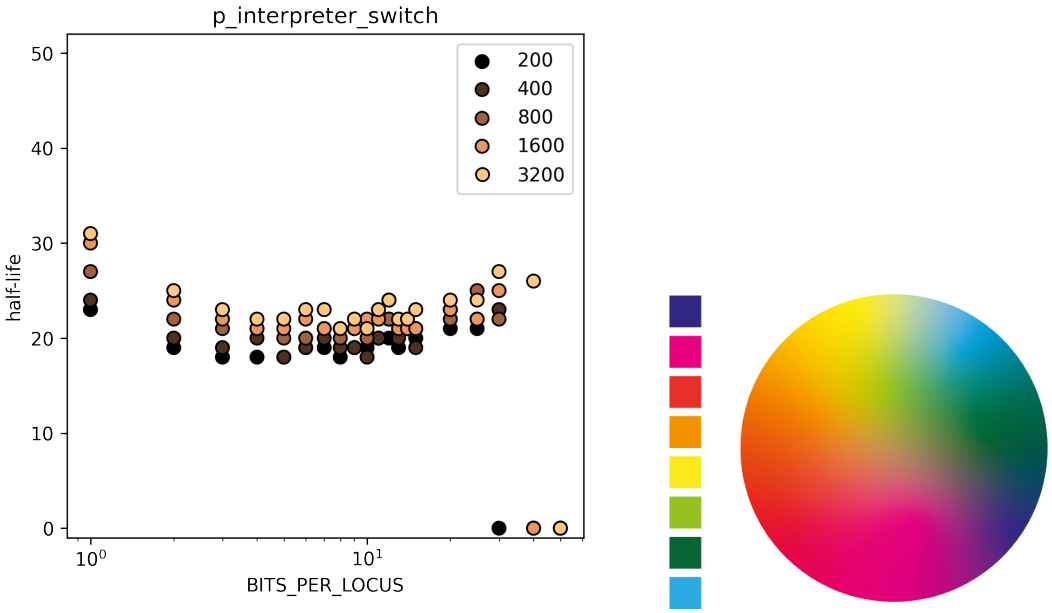
this case, extinction also happens with BPL set to 1, as one bit is not sufficient in encoding trait probabilities with the binary switch interpreter.

Switch interpreter

This experiment explores how half-life is affected by the locus granularity of *switch*-interpreted loci. Half-life is highest when *switch*-interpreted loci consist of a single bit, that is, when BPL is 1; and it is the lowest when BPL takes on values close to 6. It decreases as BPL goes from 1 to 6, but then starts slightly increasing. Some extinction occurs for BPL higher than 25, with higher BPL values leading to more severe extinction.

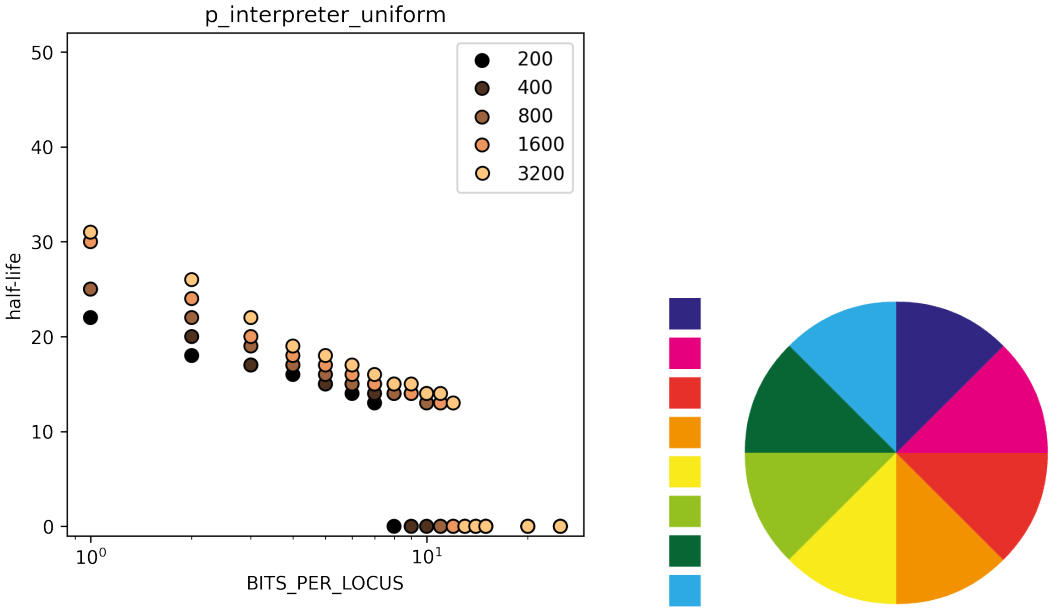
Uniform interpreter

This experiment explores how half-life is affected by the locus granularity of *uniform*-interpreted loci. When uniform interpreter is used, half-life is highest when BPL is set to 1. It consistently decreases in a log-fashion with growing BPL. High BPL values lead to extinction, which sets in at value 8, progresses severely, and affects all tested population sizes by value 13.



(a) (b)

Figure 3.7: *Half-lives of the populations evolved under a switch interpreter and a varied number of bits per locus (a) and the schematic of the switch interpreter (b)*



(a) (b)

Figure 3.8: *Half-lives of the populations evolved under a uniform interpreter and a varied number of bits per locus (a) and the schematic of the uniform interpreter (b)*

Maturation age

Here, we explore the effect of the minimum reproductive age on half-life.

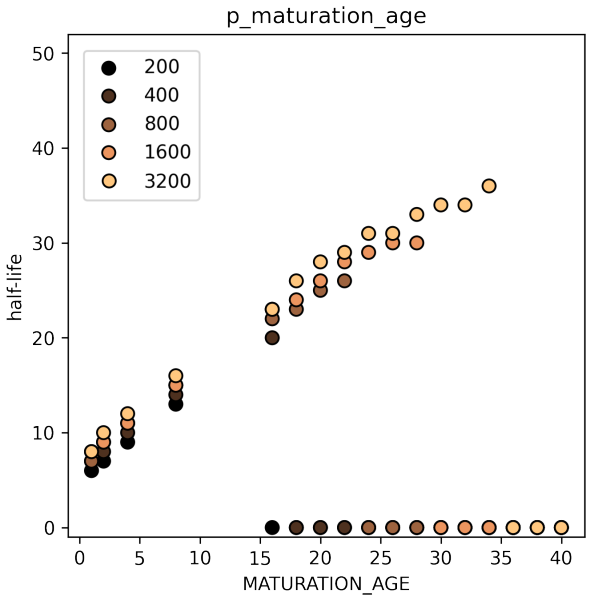


Figure 3.9: *Half-lives of the populations evolved under varied maturation age*

As maturation age increases, half-life increases virtually proportionally up to the age of 30, when it appears to flatten somewhat. Higher maturation ages also lead to extinction, which first sets in at value 15, and it affects all tested populations by value 36.

Maximum lifespan

Here, we explore the effect of limitation of maximum achievable lifespan on half-life.

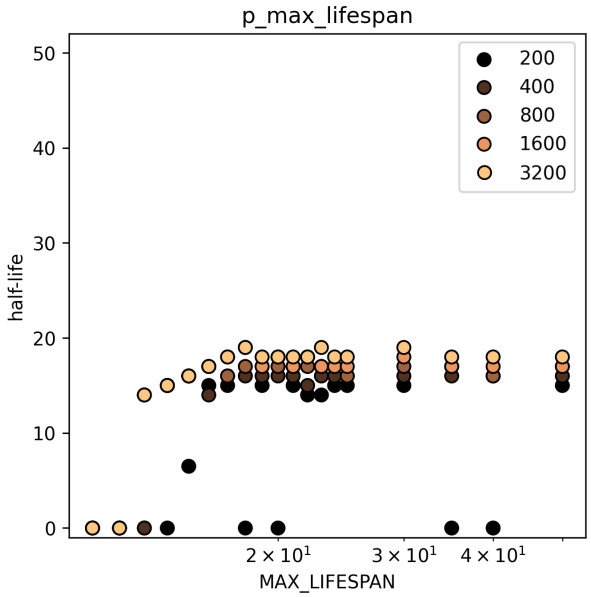


Figure 3.10: *Half-lives of the populations evolved under varied maximum lifespan*

For very low maximum lifespan values which are below default maturation age (10), all populations go extinct. For a few subsequent ages (10–14), all populations go extinct as well. For values greater than that half-life increases initially (15–19) and extinction propensity decreases greatly. For values greater than or equal to 20, half-life remains stable, and there is virtually no extinction occurring.

Mutation probability

Here, we explore the effect of probability of bits mutating on half-life.

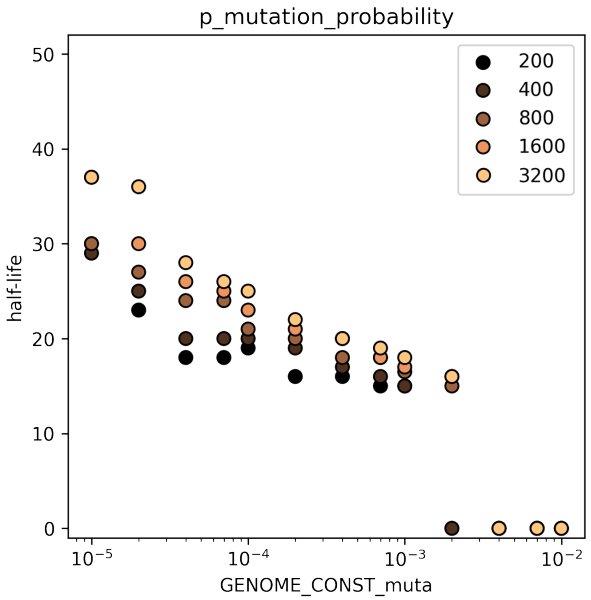


Figure 3.11: *Half-lives of the populations evolved under varied mutation probability*

Half-life is highest for the lowest tested mutation rate. As the mutation rate increases, half-life decreases in a log fashion. Once the mutation probability hits values higher than 0.1%, extinction sets in.

Mutation ratio

Here, we explore the effect of the ratio of beneficial to detrimental mutations on half-life.

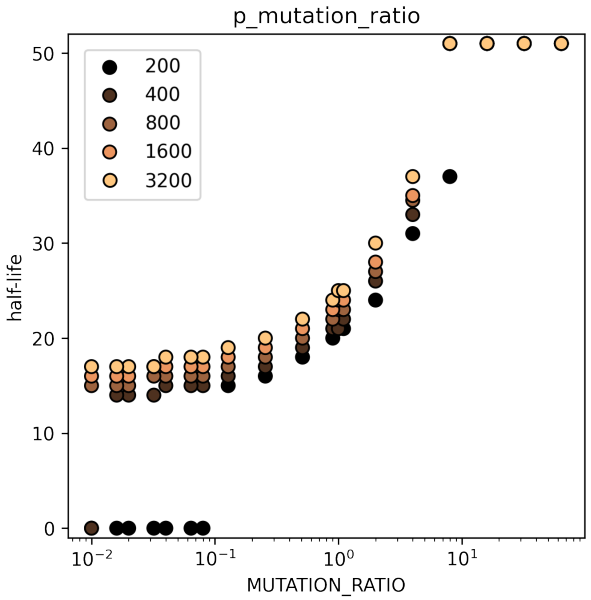


Figure 3.12: *Half-lives of the populations evolved under varied mutation ratio*

As *mutation ratio* increases, half-life increases; first by little, later by much more. At very low *mutation ratios*, there is some extinction of the very small population. At *mutation ratios* of 8 or higher, populations seem to not accumulate any detrimental mutations.

Cliff survivorship

This experiment explores the effects of cliff survivorship on half-life. Cliff survivorship refers to the fraction of the individuals that survive a population crash which occurs whenever a population exceeds its ecological capacity.

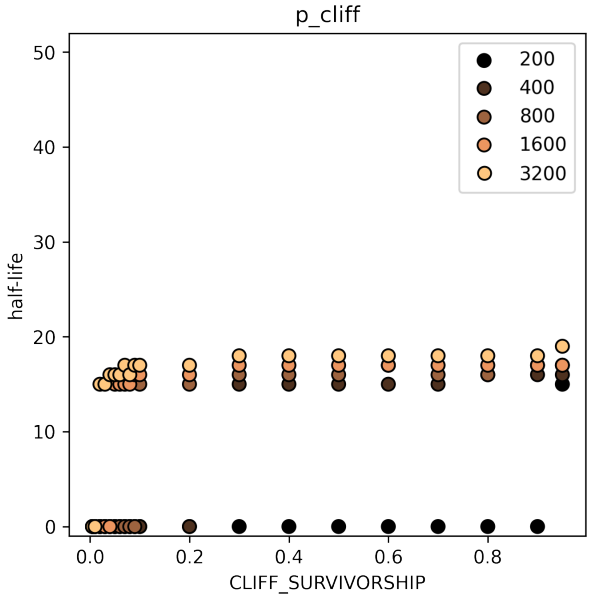


Figure 3.13: *Half-lives of the populations evolved under varied cliff survivorship*

Half-life increases as the cliff survivorship increases from very low values up to 25%; after that, it remains very much flat. The increases in cliff survivorship have the strongest effect when the cliff survivorship is still quite low. Furthermore, extinction is quite rampant for low cliff survivorship but it is virtually absent for cliff survivorships greater than 20% (apart from the very smallest population).

Overshoot scenarios

Here, we explore effects of different population control strategies on half-life[5].

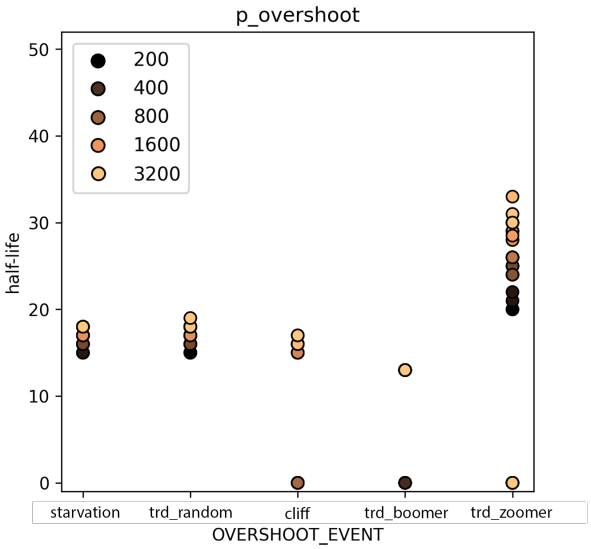


Figure 3.14: *Half-lives of the populations evolved under varied overshoot scenarios. Overshoot events on the x-axis are from left to right: starvation, treadmill random, cliff, treadmill boomer, treadmill zoomer.*

Four overshoot scenarios are considered. Boomer treadmill yields the lowest half-life, and it leads to extinction for some populations. Random treadmill and starvation lead to greater half-life than boomer treadmill, and lead to no extinction. Furthermore, their results are indistinguishable. Zoomer treadmill leads to even better results with a bigger spread between the smaller and the larger populations.

[5]Note that not all population sizes are given in the legend. The points with shades that come in between those in the legend represent population sizes that are also in between those given in the legend.

Upper bound on reproduction

Here, we explore effects of limitation of maximum reproduction probabilities on half-life.

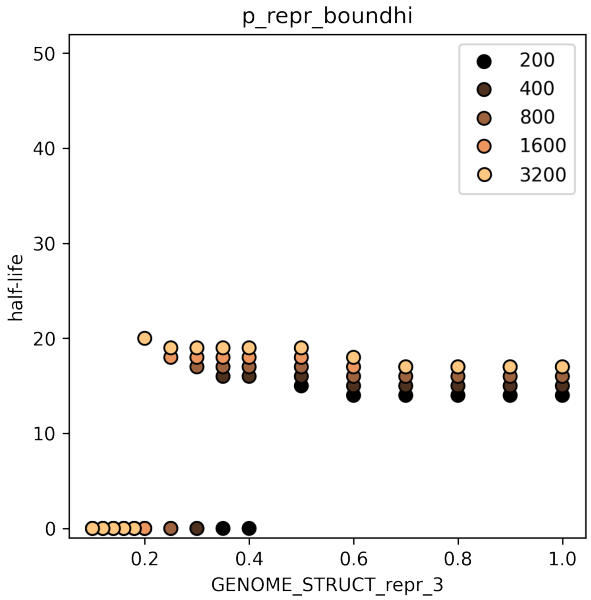


Figure 3.15: *Half-lives of the populations evolved under varied upper reproduction bound*

Half-life decreases when upper reproduction bound increases. On the other, hand the propensity to extinction decreases with it as well; it affects all populations for bounds smaller than 10%, and by value of 30%, it does not occur anymore.

Lower bound on reproduction

Here, we explore effects of lower reproduction bound on half-life. Lower reproduction bound is the minimum probability of an individual to reproduce at any mature age.

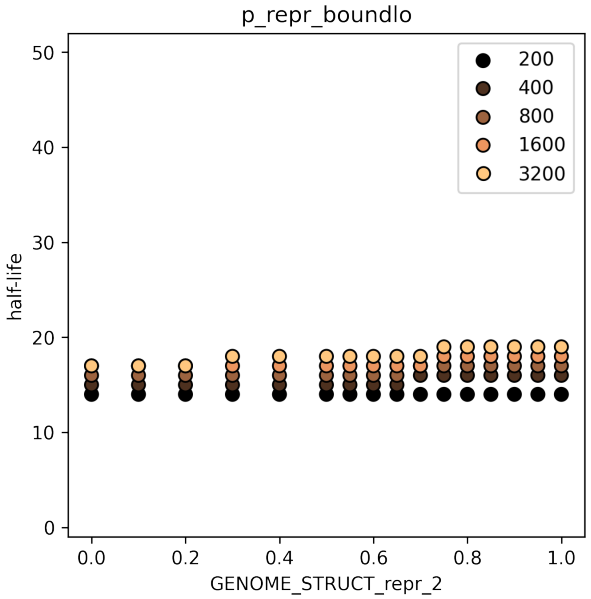


Figure 3.16: *Half-lives of the populations evolved under varied lower reproduction bound*

As lower reproduction bound increases, it leads to a very mild increase in half-life in very big populations, and to virtually no increase in smaller populations.

Reproduction mode and generation overlap

This experiment explores the effects of different combinations of asexual/sexual reproduction mode and overlapping/non-overlapping generations on half-life.

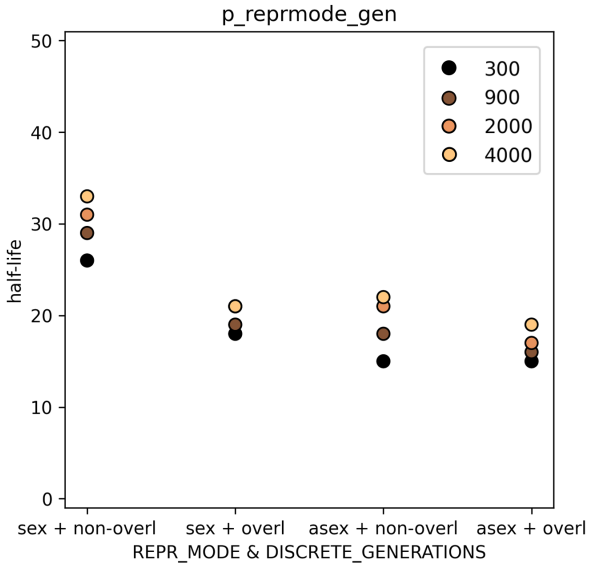


Figure 3.17: *Half-lives of populations evolved under sexual or asexual reproduction mode as well as under non-overlapping or overlapping generations. First and second x-ticks represent populations evolved under sexual reproduction, while the third and fourth represent those evolved under asexual reproduction. First and third populations evolved under non-overlapping generations, while the second and fourth populations evolved under overlapping generations.*

Sexual reproduction leads to higher half-lives for both non-overlapping and overlapping generations, though much less so for overlapping generations. Furthermore, populations with non-overlapping generations tend to achieve higher half-lives than those with overlapping generations, especially for higher population sizes.

Conclusion

Regardless of the chosen parameterization, larger populations consistently evolve half-lives that are longer than or just as long as half-lives evolved by smaller populations.

3.2 Antagonistic pleiotropy

Williams recognized that genes that may contribute to aging (e.g. by reducing survival) alongside beneficial effects on fitness can, in fact, be positively selected. He proposed that the selective forces on such genes (that is, with mixed effects on fitness) would depend on the direction and the magnitude of the respective effects as well as the *"proportion of the total reproductive probability influenced by the effect"*[15].

Here, we will be considering the simplest case of antagonistic pleiotropism (AP) genes which have one beneficial and one detrimental effect.

In this section, I will demonstrate that larger populations can hold AP genes under stronger selection (section 3.2.1), thus acquiring AP genes more easily (section 3.2.2). Finally, I will present an example of an inversion of the expected evolutionary outcomes that arises as a consequence of an interaction with MA (section 3.2.3)

To encode AP genes, we will use the technique described in detail in section 2.3.3.

3.2.1 Larger populations can select AP genes more strongly

In this section, we will simulate the evolution of AP genes in two differently sized populations to verify if differences in population size translate into differences in selection outcomes.

We will investigate AP genes with early-life benefit or mid-life benefit, and late-life detriment[6].

Approach

To encode an AP gene with early-life benefit and late-life detriment, the following AP gene is used: $(1_{+0.2}, 49_{-0.5})$[7]; and for the AP gene with mid-life benefit and late-life detriment, the following AP gene is used: $(13_{+0.2}, 48_{-0.5})$[8]. Thus, we are investigating two AP genes—at loci 1 and 13.

Here, a binary switch interpreter is used since it facilitates detecting selection while allowing for evaluation of the strength of selection. Remember that the last bit constitutes the switch part of the *binary switch interpreter*, and it must have the value 1 in order for the locus to be active at all. If the last bit is 0, the phenotypic value of the locus is 0. All the other bits are interpreted as they would be by the *binary* interpreter, with upstream bits having doubling weights.

[6] When it comes to the timing of the effects of AP genes, lifespans can be split into three periods: early-life, mid-life and late-life. Early-life encompasses exactly all the years up until the point of maturation; mid-life encompasses the early years after maturation which are reached by most individuals in a population; and late-life encompasses the years after maturation that are only exceptionally or never reached by individuals. Note that early-life can be delineated quite precisely while the limit between mid-life and late-life is fuzzy.

[7] This AP gene increases survival probability at age 1 by 20% and decreases survival probability at age 49 by 50%.

[8] This AP gene increases survival probability at age 13 by 20% and decreases survival probability at age 38 by 50%.

Two simulations are run, one with a population size of 300, and another with a population size of 3000.

Results

In the box plots 3.18 and 3.19 we can see the bit values of AP genes 1 and 13 from the smaller evolved population (N=300).

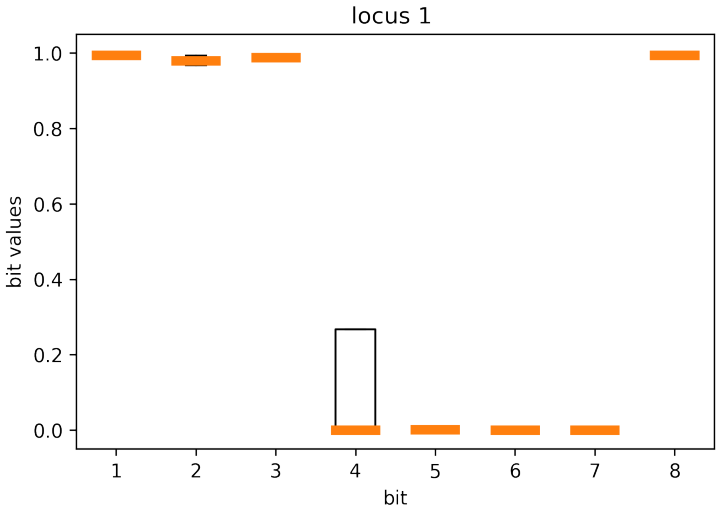


Figure 3.18: *Box plot of bit values of AP locus 1 in the smaller population (N=300)*

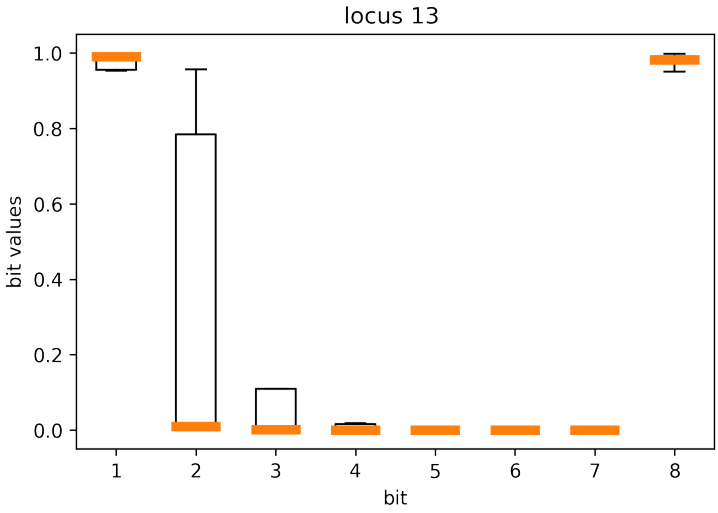


Figure 3.19: *Box plot of bit values of AP locus 13 in the smaller population (N=300)*

Since last bits (bit 8) indicates selection status in binary switch interpreters, it is apparent that both AP genes are positively selected as their eigth bits have high values with low variance.

Furthermore, locus 1 (conferring benefit in early-life) manages to hold first three bits under positive selection while locus 13 (conferring benefit in mid-life) manages to hold only the first bit under positive selection.

In the figures 3.20 and 3.21 we can see the analogous results from the larger population (N=3000).

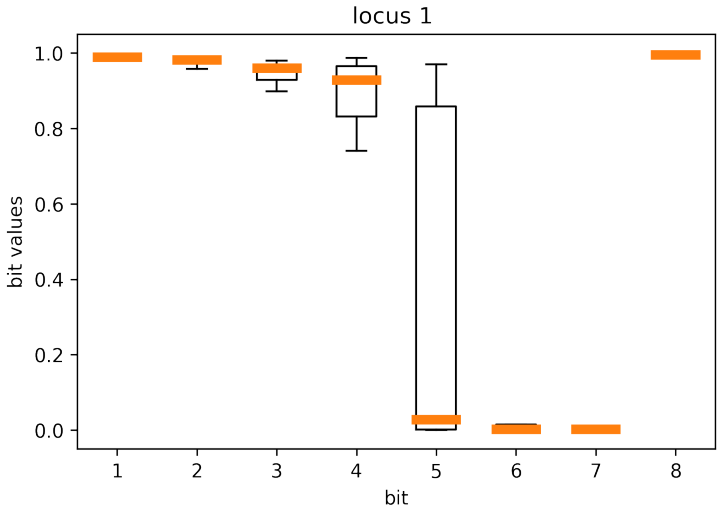


Figure 3.20: *Box plot of bit values of locus 1 in the larger population (N=3000)*

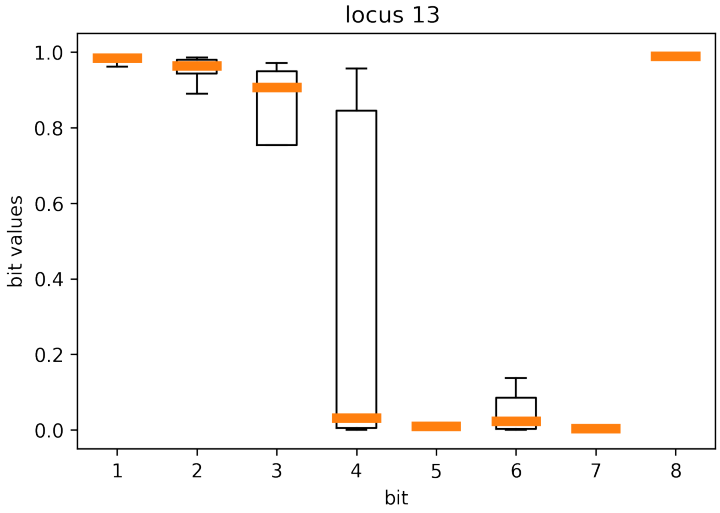


Figure 3.21: *Box plot of bit values of locus 13 in the larger population (N=3000)*

Guided by the same logic, both AP genes are positively selected in this case as well. However, a difference emerges. In this case, locus 1 holds first four bits and partially the fifth bit under positive selection; and locus 13 holds first three bits and partially the fourth bit under positive selection.

In contrast to the smaller population, larger population manages to positively select 2–3 bits more of these AP loci.

Conclusion

Both populations have positively selected the two AP loci; however, the selection in the larger population appears to be considerably more robust than in the smaller

population.

3.2.2 Larger populations can acquire AP genes more readily

The previous experiment demonstrated that larger populations can select AP genes more strongly. However, it may be useful to demonstrate that this selection strength can ultimately translate into larger population accruing and maintaining more AP genes in general.

Approach

In this experiment, ten AP genes have been made available for selection: five AP genes have varying early-life benefits and late-life detriment, and five other AP genes have equal mid-life benefits and late-life detriment.

The early-late AP genes are: $(5_{+0.2}, 30_{-0.5})$, $(6_{+0.1}, 31_{-0.5})$, $(7_{+0.05}, 32_{-0.5})$, $(8_{+0.02}, 33_{-0.5})$, and $(9_{+0.01}, 34_{-0.5})$.

The mid-late AP genes are: $(10_{+0.2}, 40_{-0.5})$, $(11_{+0.2}, 41_{-0.5})$, $(12_{+0.2}, 42_{-0.5})$, $(13_{+0.2}, 43_{-0.5})$, and $(14_{+0.2}, 44_{-0.5})$.

Here, a binary switch interpreter is used, as in the previous example, although only the switch bits will be displayed since here we are interested in the status, not the strength of selection.

Results

The figure 3.22 displays switch bit values of the AP loci in the smaller evolved population (N=300).

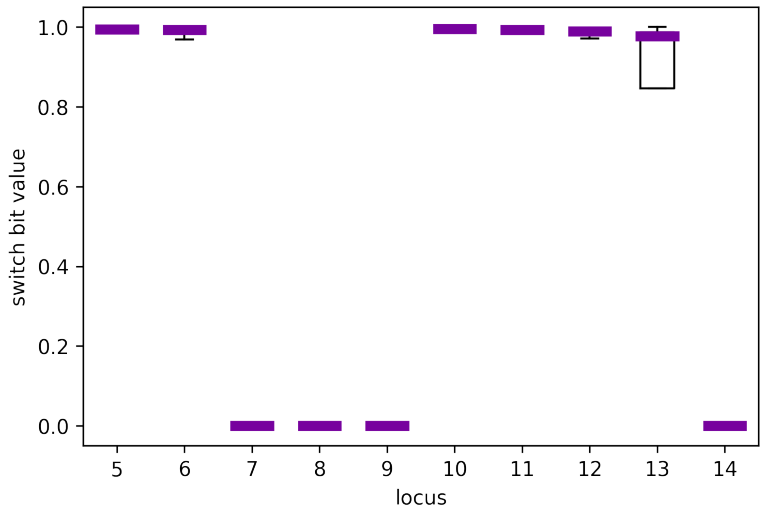


Figure 3.22: *Box plot of switch bit value of various loci in the evolved smaller population (N=300)*

According to the plot, loci 5, 6, 10, 11, 12, and 13 should be under positive selection as their switch bits have high values. Loci 7, 8, 9, and 14, on the other hand, are not positively selected.

The figure 3.23 displays switch bit values of the AP loci in the larger evolved population (N=3000).

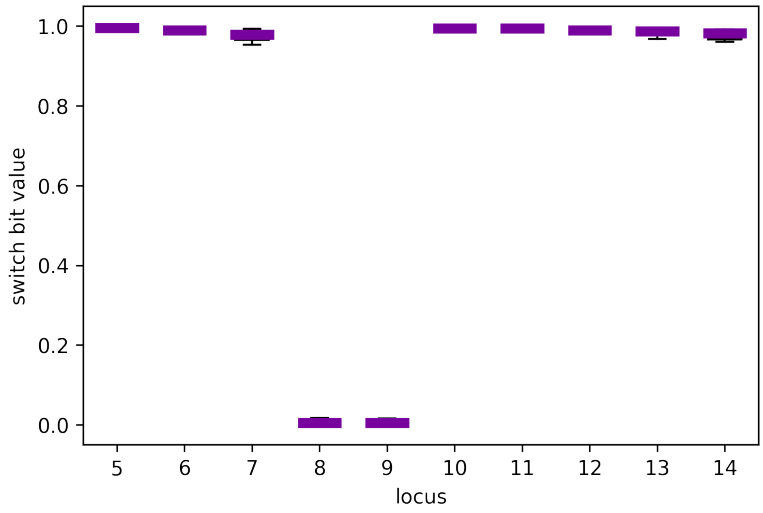


Figure 3.23: *Box plot of switch bit value of various loci in the evolved larger population (N=3000)*

According to the plot, all loci 5–14 but loci 8 and 9 should be under positive selection as their switch bits have high values.

Conclusion

The smaller population has failed to positively select four AP genes which would confer some fitness benefit, while the larger population failed to positively select only two AP genes. It appears that increasing the population size has strengthened positive selection, thus enabling the larger population to accrue more AP genes.

3.2.3 Changing population size can lead to the inversion of evolutionary outcomes

As has been just demonstrated, population size affects evolutionary outcomes of AP genes. However, population size does not have a surgically precise effect on the evolution of AP genes—rather, it alters the selective forces overall modifying the totality of the evolutionary processes, including MA.

We have shown previously (section 3.1) that population size can influence MA factors shaping life history. Here, we will examine how affecting the dynamics of MA reverberates back to modulate the dynamics of AP.

Example 1

In the first example, we will consider the following AP gene $(13_{+0.5}, 5_{-0.2})$. Two simulations will be run with two population sizes (300 and 3000). Binary switch interpreters are used.

In the figure 3.24, we can see the bits of the AP locus.

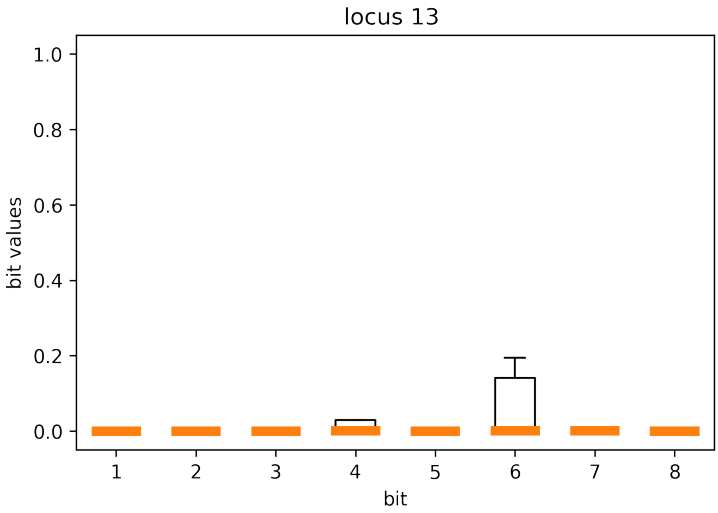


Figure 3.24: *Box plot of bit values of the locus 13 in the smaller evolved population (N=300)*

According to the low-values of the switch bit (bit 8), AP gene seems not to be positively selected.

However, when the population size is increased (figure 3.25), the AP locus becomes positively selected as is evident from the high values of the bit 8 as well as bit 1.

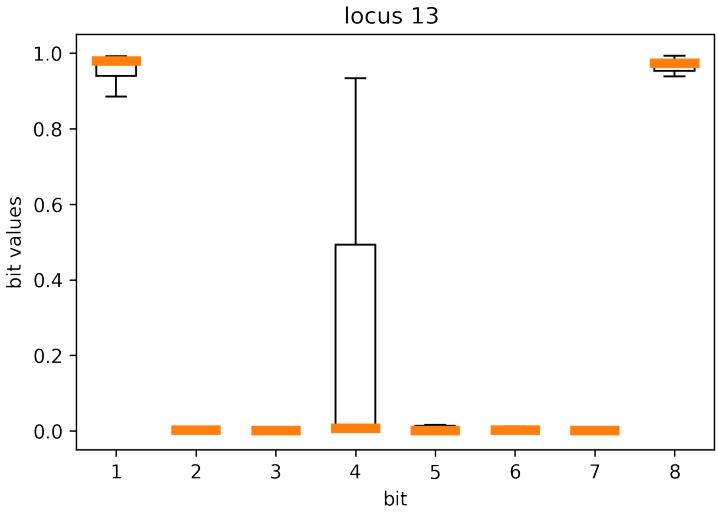


Figure 3.25: *Box plot of bit values of the locus 13 in the larger evolved population (N=3000)*

This might sound unsurprising as we have previously demonstrated that extending population size can lead to stronger selection of AP genes; however, this is not why the AP gene has become selected. To clarify, let us examine another example.

Example 2

In this example, we will consider the following AP gene: $(5_{+0.25}, 13_{-0.5})$. As previously, we will run two simulation with two population sizes; and we are using binary switch interpreters.

In the figure 3.26, we can see the bit values of the AP locus 5.

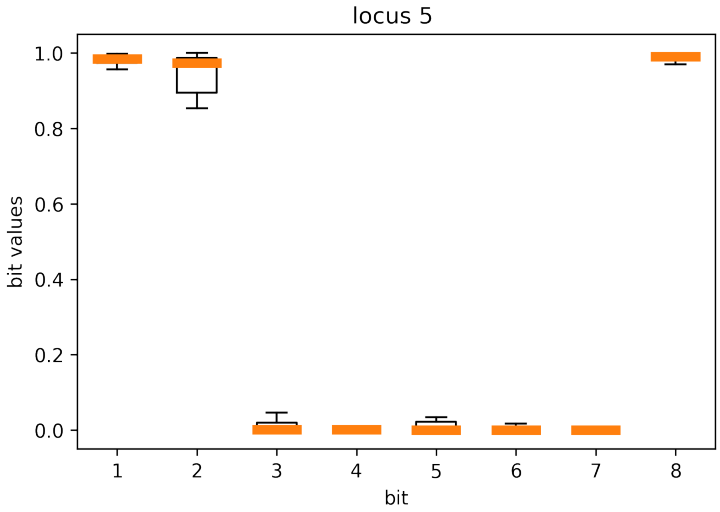


Figure 3.26: *Box plot of bit values of the locus 5 in the smaller evolved population (N=300)*

According to the high values of the switch bit (bit 8), as well as high values of the bits 1 and 2, AP gene seems to be positively, though weakly selected. AP gene seems to not be positively selected.

However, in the larger population (figure 3.27), bit switch bit locus (bit 8) is not positively selected, nor are bits 1 or 2.

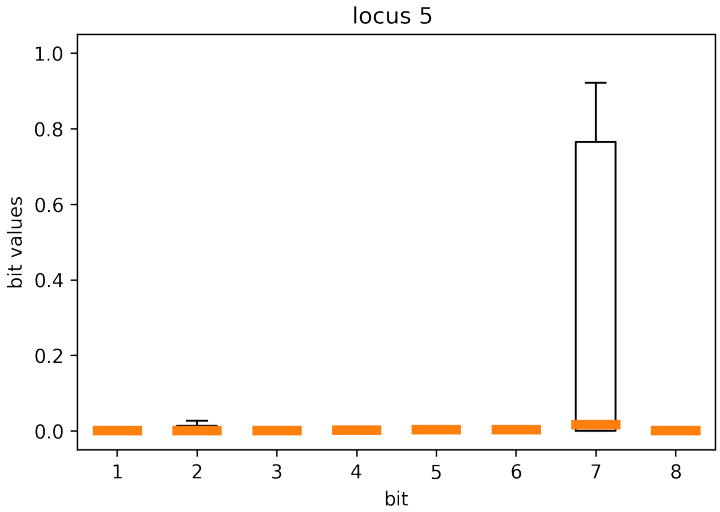


Figure 3.27: *Box plot of bit values of the locus 5 in the larger evolved population (N=3000)*

This means that increasing the population size has led to the loss of positive selection of the AP gene.

This presents a twist!

Previously, we have demonstrated that increasing population size should lead to a stronger selection of AP genes with more robust selection outcomes—but here, the

outcome has inverted. How can it be that the direction of evolution is profoundly altered by population size—a factor that merely affects genetic drift? The key is to explore the Visor profiles of the evolved genomes.

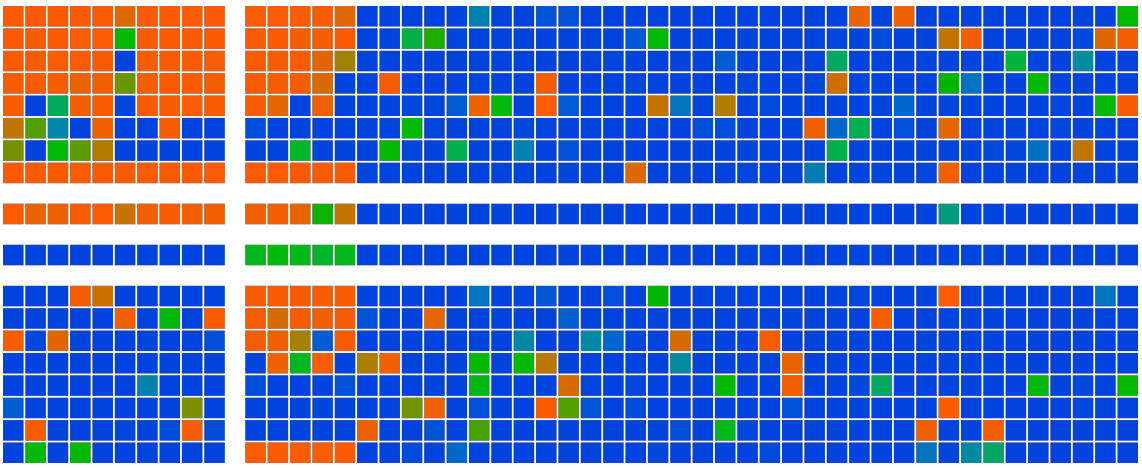


Figure 3.28: *Visor profile of the smaller population (N=300) with an AP gene at locus 5*

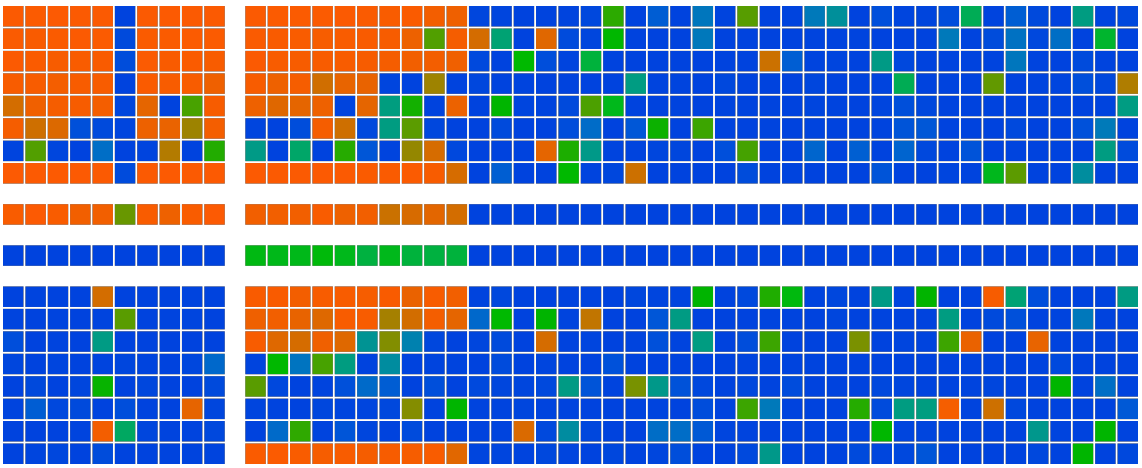


Figure 3.29: *Visor profile of the larger population (N=3000) with an AP gene at locus 5*

The Visor profile of the smaller population (figure 3.28) indicates that selection is strong up to age 15 in the smaller population, while the profile of the larger population (figure 3.29) indicates that selection is strong up to age 20 in the larger population.

Due to MA, smaller populations have evolved shorter lifespans than those of the larger populations and this difference is affecting the balance of the AP tradeoff causing it to evolve in unexpected ways.

Chapter 4

Discussion

Contents

4.1 Mutation accumulation

The conducted experiments suggest that smaller populations (relatively to larger populations) not only fail to select loci that encode for survival and reproduction in later maturity but fail to purify detrimental mutations in the loci they do manage to select. Therefore, smaller populations tend to acquire and maintain fewer beneficial genes which happen to accumulate more detrimental mutations rendering them less effective—ultimately leading to higher mortality amd shorter lifespan.

Furthermore, these observations have been shown to be stable in a variety of scenarios and not to arise as an artifact of the model. In actuality, this is not very surprising once the two theories of population genetics and mutation accumulation are unified.

Namely, as it has been explained in the introduction, population size critically affects genetic drift—smaller populations experience stronger genetic drift, while larger populations experience weaker genetic drift. That is, the randomness in the evolutionary progression is higher for smaller populations than is for larger populations, which means that smaller populations are more likely to experience unexpected evolutionary outcomes (unexpected outcomes being those that are less optimized fitness-wise). If the effect of a gene on fitness is very pronounced, genetic drift will not affect its evolutionary outcome but if its effect on fitness is subtle, it may just make a gene elude selection. This means that mutations that arise with time will not necessarily evolve as predicted by the evolutionary pressure but will drift in frequency, sinking to elimination or rising to fixation.

Now, MA suggests that the fitness implications of survival are inescapably age-decreasing—that survival at younger age is always more important (or at least as important) than survival at older age. In fact, the fitness-importance of survival is the highest all up until maturity, whereafter it rapidly declines. Once the decline has advanced enough, the fitness consequences of survival may be low enough for genetic drift to fully determine its evolutionary outcome. Essentially, MA renders genes and genetic variants that encode for survival at older ages more vulnerable to genetic drift. And since smaller populations experience stronger genetic drift, they are more likely to succumb to its effects and accumulate random mutations. Because mutations are more likely to be detrimental, efficacy of genes encoding for survival at older ages will wane with time, decreasing the genomic probability of survival, and ultimately compressing the genetic lifespan.

4.1.1 Contextualization of results in theory

In this section, we will use the introduced theoretical concepts of genetic drift and MA to explain the evolutionary outcomes of the model under different parameterizations. This should not only serve to demonstrate that these two concepts are sufficient to explain the observations but also how they interact to shape the evolutionary outcomes.

Interpreters: binary, binary switch, switch, uniform

Interpreters do not affect populations structure directly since they do not produce any age-dependent effects (thus they have no direct effects on MA). However, they

have profound effects on genetic drift since they affect the fitness coefficients of bits.

It is generally valid that increasing locus granularity (increasing the value of the parameter `BITS_PER_LOCUS` / BPL) leads to a monotonic decline of half-life. This makes sense as loci with more bits have bits that contribute individually less to the phenotype, thus have smaller fitness coefficients and are ultimately more difficult to be positively selected or purified.

This monotonic decay is especially visible for BPL values around 8 or less, and it is common to all interpreters. However, differences arise for BPL values greater than 8. In the case of binary and binary switch interpreters (see figures 3.5 and 3.6), the half-life is invariant for BPL values of 8 and higher. On the other hand, for uniform interpreters (see figure 3.8), increasing BPL leads to further decay and ultimately extinction. To understand the cause of this difference, it is important to analyze the mechanistics of the interpreters.

Increasing granularity of the uniform interpreter decreases fitness coefficients of bits making them weaker targets for selection. Though increasing granularity of the binary or binary switch interpreters does decrease fitness coefficients as well, it does it to a much lesser extent since each new added bit confers exponentially smaller fitness benefit. Furthermore, values of binary and binary switch interpreter bits cannot be decreased to arbitrarily low levels (as they can be with uniform interpreter); instead, they approximate a convergent power series. This is important in understanding why increasing granularity of binary and binary switch interpreters does not increase the propensity to extinction while it does for the uniform interpreter—the selection coefficients of the uniform interpreter bits can be degraded to arbitrarily small values which are simply exceedingly difficult to maintain via selection.

In the case of the switch parameter, the evolutionary dynamics are a bit different (see figure 3.7). While in previous interpreters every mutation has some effect on the phenotype, in the switch interpreter, most mutations do not lead to any changes to the phenotype, since any locus configuration that contains both zeros and ones (and not exclusively zeros nor exclusively ones) exhibits the same phenotype. This means that once a mutation occurs to the all-1 locus or the all-0 locus, evolution of that locus effectively starts behaving as a random walk. Because of this, the evolutionary process is more random leading to a slight perceived increase toward the higher BPL values, as well as the shuffled ranking of the differently sized populations with respect to half-life.

Maturation age

Half-life increases with maturation age (see figure 3.9). As maturation age increases, half-life must necessarily increase as well; otherwise, the individuals would not reach the reproductive age and the population would go extinct. Therefore, maturation age presents a lower bound on half-life. However, once reproductive age is reached, the already discussed effects of aging predicted by MA set in and present an upper bound on half-life.

However, it is not directly clear why delaying reproduction would lead to higher extinction rates. Since a detrimental mutation experienced at any age before onset of reproduction has the same effect on fitness, purifying detrimental mutation should

in theory not become any more difficult.

First of all, genetic drift is stronger for loci that encode traits exhibited later in life (even if those traits occur before the onset of maturity). As the effects of these loci will not become apparent for a considerable stretch of time, it is possible that external mortality by chance strikes down an individual that has accumulated fewer detrimental mutations in loci that are active just before maturity, while the individual with more detrimental mutations remains alive. Once maturation age becomes too great, the effect of genetic drift may be too strong to be countered by selection. This is also why extinction rates start rising first for the smaller populations.

There is another factor as well that is a consequence of the increase number of loci, thus increased number of 1-bits and ultimately an increased absolute number of detrimental mutations acquired at each reproductive cycle. This effect will be discussed below, in the `mutation probability` section.

Maximum lifespan

As long as maximum lifespan is long enough that almost no individual reaches it, it has no effect on half-life (see figure 3.10). However, once it reaches the evolved half-life, shortening maximum lifespan will trivially reduce the half-life values. If the maximum age is set too low, it may not provide enough time for individuals to replace themselves, thus leading to extinction. This happens for all values set below 15.

The effects of this parameter are strictly artifacts of the model and have little biological significance. It is thus important that the maximum lifespan is held above the evolvable lifespan.

Mutation probability

Half-life decreases with increasing mutation probability (see figure 3.11). As mentioned in the `maturation age` section above, an increase in the absolute number of detrimental mutations acquired at each reproductive cycle can have a detrimental effect on half-life. This is because natural selection acts on the selection gradient (that is, on the relative fitness between individuals), so if nearly all individuals degrade somewhat, the upper bound of that gradient erodes, thus it can hardly pull the population fitness upwards. If fitness of all individuals degrades, it is essentially impossible to avoid extinction unless beneficial mutations occur frequently enough.

This is relevant as increasing mutation probability leads to a higher absolute number of detrimental mutations. Do note that concentrating the phenotypic effect of the loci onto fewer bits (such as in the binary interpreter in comparison to the uniform interpreter) can soften this effect, as many supposedly detrimental mutations actually have a nearly neutral effect on fitness.

Mutation ratio

In AEGIS+ *mutation ratio* is defined as the ratio of the probability of beneficial mutations versus the probability of detrimental mutations.

Half-life increases with increasing *mutation ratio* (see figure 3.12).

Mutation ratio can be brought into connection with mutation probability by construing beneficial mutations as neutralizers of detrimental mutations. As *mutation ratio* increases, the effective rate of detrimental mutation decreases, thus extending half-life by the logic explicated earlier.

Note, however, that reaching the *mutation ratio* of 1:1 does not translate to a maximum half-life, nor does reaching *mutation ratios* somewhat higher than that. That means that even though evolution is biased toward beneficial mutations, detrimental mutations still accumulate in later age, which unequivocally demonstrates that loci in late-life evolve neutrally. It is only once the *mutation ratio* is much higher (approximately 1:10) that the evolved genomes could ensure survival up the the very late ages, due to detrimental mutations being probabilistically extinguished by beneficial mutations.

Cliff survivorship

Cliffing is one of the available overshoot scenarios (discussed in the section just below). When cliffing is enabled, once a population exceeds a given size limit (trigger size), it is slashed by a fraction of that trigger size. Cliff survivorship is that fraction; thus the higher it is, the larger the chunk of the population that survives each cliffing event.

When trying to understand the consequences of cliffing (see figure 3.13), it is important to consider both the effects of MA, as well as genetic drift.

When it comes to genetic drift, cliffing causes population bottlenecks. The smaller the cliff survivorship, the stronger the effects of population bottlenecks—and those are stronger randomization of evolutionary outcomes and more likely fixation of detrimental mutations which ultimately drives the half-life down.

When it comes to MA, it is important to recognize the effect of cliffing on the death structure. When cliff survivorship is 1, this scenario is effectively just a random treadmill. This means that the there is a fairly constant mortality probability imposed on every individual every stage; thus, there is an age-decreasing survival probability as mortality probabilities accumulate multiplicatively. This is the classical scenario described in the theory.

Now, when the cliff survivorship is decreased, each cliffing event produces an ecological gap that is left unfilled for some time, instead of it being reached already at the next stage. This essentially means that the population turnover is decreased (figure 4.1) and that individuals do not incur any mortality. Thus, between cliffing events, the population should evolve under expansionary dynamics. This leads to softer slopes in the death structures and should thus weaken the effect of MA, leading to longer lifespans; however, the results do not support this.

When it comes to extinction, MA cannot explain the rampant extinction rates at very low cliff survivorship values. On the other hand, strong genetic drift can lead to extinction if it is so strong that even the very important loci have a hard time being maintained.

In sum, though cliff survivorship affects both the dynamics of genetic drift and population structure (MA), it seems that they are opposing forces and that genetic drift

plays a more critical role in explaining the decrease of half-life for very low cliff survivorship rates.

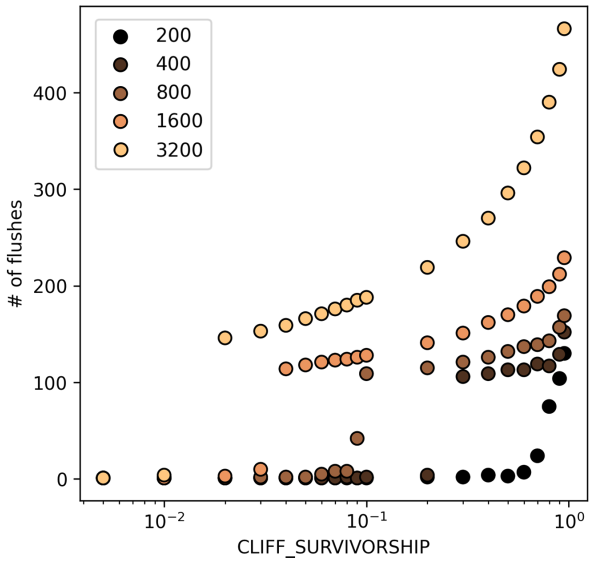


Figure 4.1: *Number of recorded flushes (proxy for population turnover) in a simulation parameterized with a particular cliff survivorship*

Overshoot scenarios

Overshoot scenarios are crucial in shaping the population structure which further influences the forces of MA on the life history evolution.

Comparing three treadmill modes (random, boomer, and zoomer) uncovers significant differences (see figure 3.14). Populations evolving under the `zoomer treadmill` maintain highest half-lives, while the populations evolving under the `boomer treadmill` evolve the lowest half-lives. This is in accordance with the theory, as the `boomer treadmill` imposes ecological mortality on the oldest individuals, thus making the survivorship curve more steep, and weakening the fitness contribution of late-life loci. On the other hand, `zoomer treadmill` imposes ecological mortality solely on the youngest individuals which usually do not even reach maturity, thus the survivorship curve of mature age-groups does not decline, which should lead to weakening if not downright elimination of the factor necessary to produce MA effects.

Lastly, `treadmill random` imposes ecological mortality uniformly across individuals, thus, it sits between the two edge cases and presents a textbook example of MA effects.

When it comes to `starvation`, which is the overshoot event set as default, the results seem to be very much indistinguishable from `random treadmill`. Because both overshoot events distribute mortality load evenly across the population, it makes sense from the MA perspective that they lead to similar, if not identical results. The difference between the two options is the timing of the mortality, which is instantaneous for the `random treadmill`, while it is delayed in the case of `starvation`. Also, `starvation` will usually lead to an overcorrection in the population size by dipping

the population below the threshold in the last stage of elimination (thus producing a slight cliffing effect as observed when `CLIFF_SURVIVORSHIP` parameter is used). These two mechanistic differences can lead to differences in outcome. The mortality delay allows individuals to reach later ages more frequently, thus alleviating a part of the MA effect and extending half-life. On the other hand, an overcorrection introduces a kind of a bottleneck that increases genetic drift, thus constricting half-life. However, the two effects seem to cancel out which makes `random treadmill` a good approximation of the more biologically representative `starvation` option.

When it comes to `cliff`, the evolved half-lives are lower than those of the `random treadmill` and `starvation`. Since cliffing has the bottlenecking effect which increases genetic drift but has no other side effects that would counterbalance that, the half-lives are predictably reduced.

Upper bound on reproduction

Decreasing the upper bound on reproduction leads to a smaller number of offspring born each stage, thus a smaller population turnover due to exceeding the resource limit, and ultimately to a smaller ecological mortality rate. This softens the survivorship curve, allows individuals to live longer, and strengthens selection in late-life loci, leading to an extension of half-life (see figure 3.15).

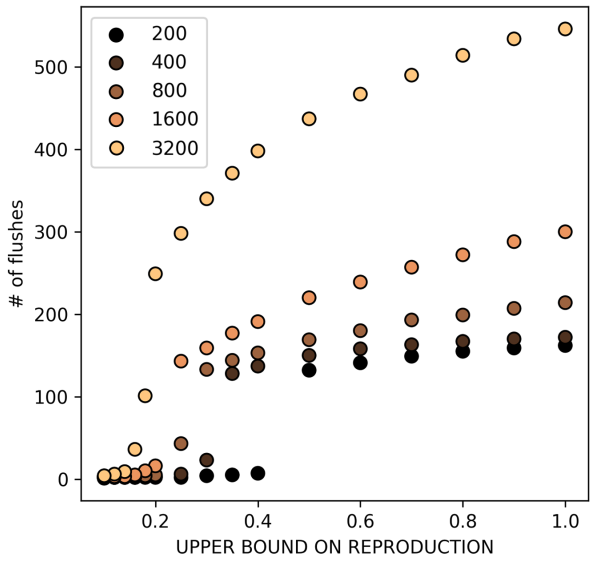


Figure 4.2: *Number of recorded flushes (proxy for population turnover) in a simulation parameterized with a particular upper bound on reproduction*

It is evident from the figure 4.2 that the number of flushes goes down dramatically for low upper bounds on reproduction, which is a proxy for the population turnover.

Naturally, extinction rate goes up as the upper bound on reproduction decreases since individuals need to live longer (that is, maintain more genes for survival) in order to replace themselves. Again, smaller populations are more vulnerable.

Lower bound on reproduction

It may be necessary to address the apparent paradox between the observations made in this and the previous section. Namely, one might wonder how it is that increasing reproductive success should lead to greater half-lives yet in the previous section more reproduction led to shorter half-lives (see figure 3.16).

Now, while it is true that increasing lower bound on reproduction will lead to more reproduction, when we actually examine the increase in population turnover, we observe a feeble increase.

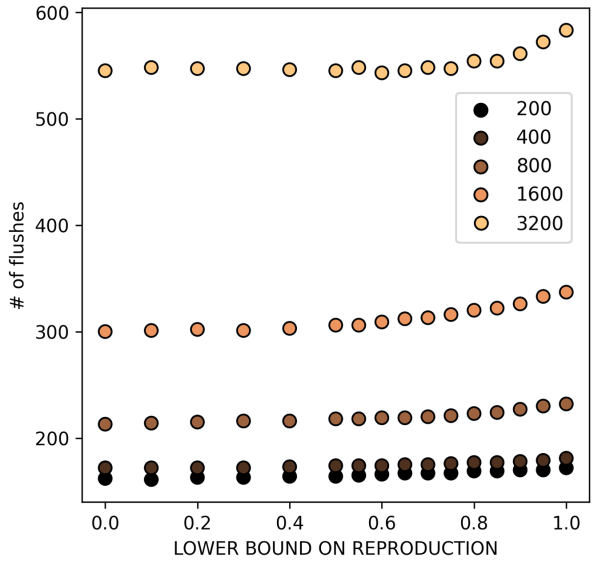


Figure 4.3: *Number of recorded flushes (proxy for population turnover) in a simulation parameterized with a particular lower bound on reproduction*

It is evident from the figure 4.3 that the number of flushes did increase but not by much. (On the other hand, the number of flushes in the previous section increased by much more.) This means that the turnover was not increased by much, and that the ecological mortality has not changed by much either.

For now, we have shown that increasing the lower bound on reproduction should not lead to a strong decrease in half-life; however, what we actually see is an increase in half-life. This has a fairly simple explanation and that is that increasing the reproductive guarantee at all ages leads to survival loci being more valuable as they have a higher probability of leading to reproduction, thus increasing an individual's fitness. Usually, reproduction and survival must coevolve in order to be useful fitness-wise; however, this is not a perfect process, which means that reproduction probabilities cannot be maintained at exceptionally high values. If reproduction probability is lower, the benefit of the corresponding survival locus is lower as well, and the efficiency of selection on it.

In this scenario, there is no extinction as lower bound on reproduction does not inhibit individuals' ability to replace themselves; in fact, it increases their success of doing so.

Reproduction mode and generation overlap

According to the simulations, sexually reproducing populations outperform asexually[1] reproducing populations with respect to the evolved half-life (see figure 3.17). In sexual populations, during reproduction parental genomes undergo recombination and assortment. This should in theory, independently of MA and genetic drift, lead to improved evolutionary outcomes as beneficial and detrimental mutations can be aggregated and separated, and in that form emitted onto the next generation [71]. The performance increase is especially prominent in the non-overlapping generations, while it is modest in the overlapping generations.

When it comes to the existence or lack of generation overlap, the model suggests that non-overlapping generations evolve longer half-lives than overlapping generations under the preset parameterization, which is true for both sexually and asexually reproducing populations. When interpreting this effect, both MA and genetic drift must be considered.

Since non-overlapping generations experience periodic bottlenecks wherein a fraction of the offspring generated survives, the genetic drift should be greatly exacerbated. Overlapping generations do not produce these bottlenecks; instead, all offspring are allowed to survive with some chance up to some ages during which they may express their genomes and suffer the consequences of their genetic makeup. That way, offspring is not blindly eliminated but its elimination depends on its fitness. This should protect the overlapping generation from individuals with grave or even moderately detrimental mutations from surviving and reproducing. In contrast, non-overlapping populations have no such protection barriers. This effect should reduce half-lives of non-overlapping generations, but that is not what we observe.

In order to account for that discrepancy, we must consider MA. First of all, overlapping generations exhibit steep survivorship curves, while non-overlapping generations have virtually flat survivorship curves. This is because overlapping generations must control the population size which they do by continuous elimination of excess individuals, decreasing survivorship in an age-dependent fashion. On the other hand, non-overlapping generations do not need to control the population size as the population cannot grow during a season. Since there is no other source of extrinsic mortality (for example, due to trauma or predation) coded into the model, the non-overlapping generations are effectively not fulfilling the Medawar's condition necessary to produce the MA effect. This leads to significantly longer half-lives in non-overlapping populations.

Note that non-overlapping generations developing longer lifespans than overlapping generations should not be taken as a general truth. It is unwise to extend this observation onto reality without injecting other sources of extrinsic mortality first. However, by extinguishing the Medawar's MA condition, we can clearly see the extent to which half-lives are constricted due to MA.

[1] Note that the reproduction mode has actually been set to `diasexual` in which individuals reproduce asexually but have diploid genomes. Ensuring that both sexual and asexual populations are diploid is important since ploidy affects the selection coefficient of bits.

4.2 Antagonistic pleiotropism

In the introduction we hypothesized that the importance of AP genes in shaping life history is greater in larger populations than in smaller ones. The results support this hypothesis by showing that larger populations acquire AP genes more readily and select them more strongly. In this section, we will explain why that is.

We will start with contextualizing these results in established theory by identifying the theoretical factors that drive the evolution of AP genes, formulating evolutionary predictions based on how these factors were configured in our model, and comparing those predictions with our results.

Performing these steps serves a dual role: it demonstrates that our model behaves in accordance with the theory thus increasing the trustworthiness of the results, but it also shows that the theoretical phenomena emerge naturally from a simple evolutionary setting.

To fully explain the results, we must successfully account for the *selection status* and *selection strength* of the simulated AP genes.

4.2.1 Selection status

An AP gene can either be positively selected, purified, or it can evolve neutrally. Here, I present theoretical considerations that can be used to predict the evolutionary outcome of an AP gene.

Williams' factors

In his seminal paper [15], Williams lists and explains the role of the essential factors that control the outcomes of genes with pleiotropic effects. There are three factors that should be considered for every effect produced by the pleiotropic gene:

1. *Direction of the effect.* It can be positive or negative, i.e. does the effect increase or decrease fitness.
2. *Magnitude of the effect.* That is, by how much the effect increases or decreases fitness.
3. *Proportion of the total reproductive probability influenced by the effect.*

Williams uses the term *m-value* to refer to the signed value of the *magnitude of the effect*. The sign of the *m-value* encodes the *direction of the effect* so that a negative *m-value* denotes a detrimental effect, while a positive *m-value* denotes a beneficial one. Williams also uses the term *p-value* to refer to the value of the *proportion of the total reproductive probability influenced by the effect*. This value comes from the interval $[0, 1]$ where effects with postreproductive onset have a *p-value* of 0, effects with an onset before maturity have a *p-value* of 1, and effects with an onset during maturity have a *p-value* between 0 and 1.

It follows from the abovementioned paper that the expected evolutionary outcome can be predicted by comparing the products of *m-values* and *p-values* of the beneficial ($|m_+p_+|$) and the detrimental effect ($|m_-p_-|$):

- if $|m_+p_+| > |m_-p_-|$, the fitness increment outweighs the fitness decrement of the AP gene, thus the gene is expected to be positively selected,
- if $|m_+p_+| < |m_-p_-|$, the fitness decrement outweighs the fitness increment of the AP gene, thus the gene is expected to be purified, and
- if $|m_+p_+| \approx |m_-p_-|$, the evolution is expected to transpire approximately neutrally.

If *m-values* and *p-values* are known, this case breakdown can be used to predict the evolutionary outcomes of AP genes.

Williams' factors translated in AEGIS+

To use the comparison of *mp* products ($|m_+p_+|$ and $|m_-p_-|$) to predict evolutionary outcomes of AP genes, we need to identify *m-values* and *p-values* in AEGIS+.

Identifying *m-values* is straightforward as they are explicitly encoded in AEGIS+ and can be directly read out from the AP shorthand (a_{m_+}, b_{m_-}). For example, in section 3.2.1, we investigated the AP gene $(1_{+0.2}, 49_{-0.5})$. This gene has *m-values* $+0.2$ and -0.5.

On the other hand, identifying *p-values* is not at all straightforward. To know the *p-value*, we need to know the life history of individuals—more specifically, their survival and reproduction probability distributions, but those are difficult to predict (especially considering that there is a complex interaction in the evolution of AP genes and life histories); that is why AEGIS exists in the first place.

However, if we focus on a subset of cases, we can make solid predictions. We know that the *p-value* is 1 for all effects that are active during prematurity (which we know exactly) and that the *p-value* is 0 for all effects active in the postreproductive period (which we can estimate well). If the effect occurs during the reproductive period, its exact *p-value* in AEGIS+ cannot be reliably predicted; however, if the beneficial and the detrimental effects occur at similar times, their *p-values* are approximately equal which makes predicting tractable again.

When it comes to *p-values*, we can make solid predictions about 9 cases, in which the *p-value* of the beneficial effect is either 0, 1, or some intermediate value and the *p-value* of the detrimental effect is either 0, 1, or equal to the *p-value* of the beneficial effect. The elusive cases are those where both effects occur during the reproductive period but at significantly different time points.

When it comes to *m-values*, there are uncountably many cases to consider as *m-values* can theoretically be any real number. However, since we are interested in the relationship of the *mp* products, all cases can be categorized into three supercases:

- The magnitude of the beneficial effect is greater than the magnitude of the detrimental effect ($|m_+| > |m_-|$).
- The magnitude of the beneficial effect is equal to the magnitude of the detrimental effect ($|m_+| = |m_-|$).
- The magnitude of the beneficial effect is less than the magnitude of the detrimental effect ($|m_+| < |m_-|$).

The exact ratio of the two magnitudes is not relevant since the exact ratio of the *mp* products is not relevant, but their relationship.

In conclusion, adapting Williams' factors into the context of AEGIS+, we identify 3 cases for the *p-value* of the beneficial effect, 3 cases for the *p-value* of the detrimental effect, and 3 cases for the relationship of *m-values* of the two effects. This yields 27 $(3 * 3 * 3)$ AP gene configurations whose evolutionary outcomes we should be able to predict.

Constructing theoretical predictions of selection status of AP genes

In this section, we will make theoretical predictions of selection status for 27 different AP gene configurations. We will distribute these cases into three matrices: in the first matrix, $|m_+| > |m_-|$; in the second matrix, $|m_+| = |m_-|$; and in the third matrix, $|m_+| < |m_-|$. The columns represent different cases for the *p-value* of the detrimental effect, while the rows represent different cases for the *p-value* of the beneficial effect. The cases are indicated as column and row labels in each matrix.

The cells contain values `+`, `N` or `-` to indicate that the prediction under certain configuration is positive selection, neutral evolution or purifying selection, respectively. Two cells contain the value *?* which indicate that the prediction is complex; these cases will be discussed later[2].

The predictions were made using the logic of three cases of *mp* product relationships: $|m_+p_+| > |m_-p_-|$, $|m_+p_+| \approx |m_-p_-|$, and $|m_+p_+| < |m_-p_-|$ with the first one mapping onto `+` (positive selection), the second one onto `N` (neutral evolution) and the third one onto `-` (purifying selection) as has been discussed in the section with Williams' factors.

For example, In the upper-left cell of the matrix T^+, the prediction is positive selection (`+`) because $|m_+p_+| > |m_-p_-|$ since $p_+ = 1$, $p_- = 1$ and $|m_+| > |m_-|$.

$$T^+ = T_{|m_+|>|m_-|} = \begin{array}{c} \begin{array}{ccc} p_- = 1 & 0 < p_- < 1 & p_- = 0 \end{array} \\ \left[\begin{array}{ccc} + & + & + \\ ? & + & + \\ - & - & N \end{array}\right] \end{array} \begin{array}{c} p_+ = 1 \\ p_+ = p_- \\ p_+ = 0 \end{array}$$

$$T^= = T_{|m_+|=|m_-|} = \begin{array}{c} \begin{array}{ccc} p_- = 1 & 0 < p_- < 1 & p_- = 0 \end{array} \\ \left[\begin{array}{ccc} N & + & + \\ - & N & + \\ - & - & N \end{array}\right] \end{array} \begin{array}{c} p_+ = 1 \\ p_+ = p_- \\ p_+ = 0 \end{array}$$

$$T^- = T_{|m_+|<|m_-|} = \begin{array}{c} \begin{array}{ccc} p_- = 1 & 0 < p_- < 1 & p_- = 0 \end{array} \\ \left[\begin{array}{ccc} - & ? & + \\ - & - & + \\ - & - & N \end{array}\right] \end{array} \begin{array}{c} p_+ = 1 \\ p_+ = p_- \\ p_+ = 0 \end{array}$$

[2] In this case, the exact ratio of the magnitudes is required to make a correct prediction.

#	*configuration*	*prediction*
1	$(4_{+0.3}, 5_{-0.1})$	+
2	$(4_{+0.3}, 5_{-0.3})$	N
3	$(4_{+0.3}, 5_{-0.5})$	-
4	$(14_{+0.3}, 15_{-0.1})$	+
5	$(11_{+0.3}, 12_{-0.3})$	N
6	$(14_{+0.3}, 15_{-0.5})$	-
7	$(44_{+0.3}, 45_{-0.1})$	N
8	$(44_{+0.3}, 45_{-0.3})$	N
9	$(44_{+0.3}, 45_{-0.5})$	N
10	$(1_{+0.3}, 40_{-0.1})$	+
11	$(11_{+0.3}, 3_{-0.1})$	?
12	$(4_{+0.3}, 12_{-0.1})$	+
13	$(15_{+0.3}, 41_{-0.1})$	+
14	$(43_{+0.3}, 2_{-0.1})$	-
15	$(45_{+0.3}, 13_{-0.1})$	-
16	$(1_{+0.3}, 40_{-0.3})$	+
17	$(11_{+0.3}, 3_{-0.3})$	-
18	$(4_{+0.3}, 12_{-0.3})$	+
19	$(15_{+0.3}, 41_{-0.3})$	+
20	$(43_{+0.3}, 2_{-0.3})$	-
21	$(45_{+0.3}, 13_{-0.3})$	-
22	$(1_{+0.3}, 40_{-0.5})$	+
23	$(11_{+0.3}, 3_{-0.5})$	-
24	$(4_{+0.3}, 12_{-0.5})$	?
25	$(15_{+0.3}, 41_{-0.5})$	+
26	$(43_{+0.3}, 2_{-0.5})$	-
27	$(45_{+0.3}, 13_{-0.5})$	-

Figure 4.4: Chosen AP configurations and the theoretical predictions of their evolutionary outcomes

Testing theoretical predictions of selection status of AP genes

In the previous section, we have constructed and organized 27 cases of AP gene configurations and their theoretical predictions. In this section, we will run experiments to test these predictions. The table 4.4 contains the information about the experiments that will be run including the AP configurations[3] and theoretical predictions of their evolutionary outcomes.

Here, we will present the plots from the first three experiments; other plots can be found in the appendix. Note that we are using the switch interpreter in these experiments. Switch loci are positively selected when all their bits have high values; they are negatively selected when all their bits have low values; otherwise, they are neutrally evolving.

[3]Note that the onset of prematurity is at age 10.

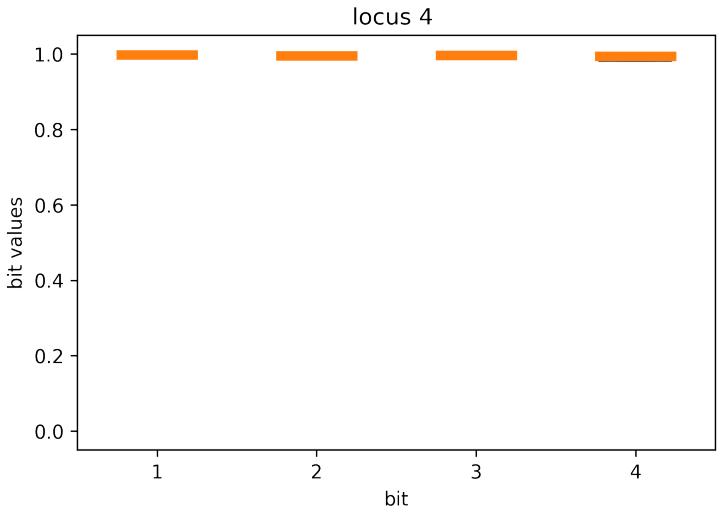


Figure 4.5: *Bit values of the AP gene 1. All bits have high values, thus the AP gene is positively selected*

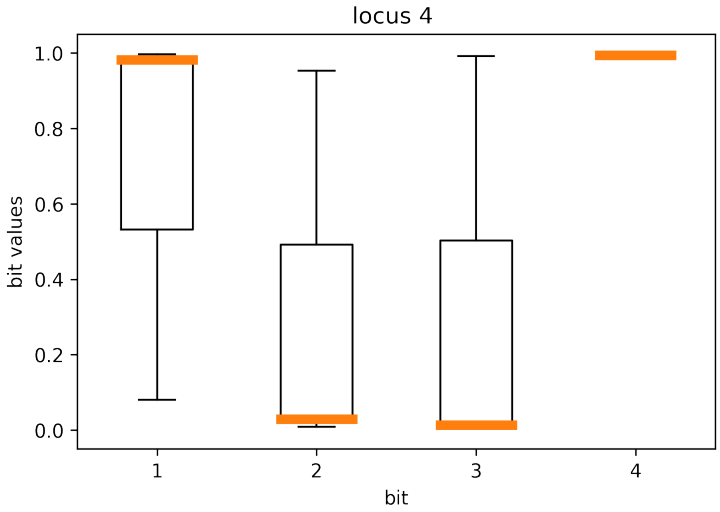


Figure 4.6: *Bit values of the AP gene 2. Values of bits are highly variant and mutually different, thus the AP gene is neutrally evolving*

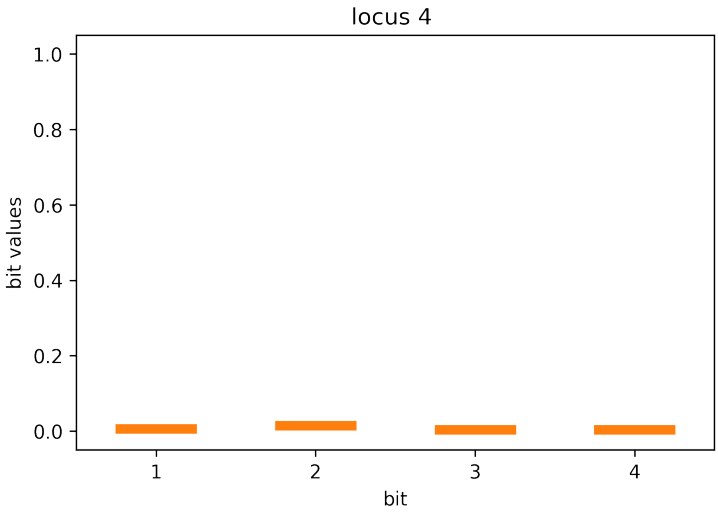


Figure 4.7: *Bit values of the AP gene 3. All bits have low values, thus the AP gene is negatively selected*

According to their bit values, the first three evolved AP genes exhibit positive selection, neutral evolution and negative selection, respectively. This is in concordance with our theoretical predictions laid out in the table 4.4. Other AP genes (presented in the appendix) exhibit evolutionary outcomes in concordance with the theoretical predictions, too.

The cases 11 and 24 are exceptions in this framework as they can theoretically exhibit any of the three selection statuses depending on the exact *m-values* and *p-values* of the pleiotropic effects. This is discussion in depth in the following text.

In conclusion of selection status prediction

In order to contextualize our main results (section 3.2) in the wider theoretical picture, we set out to develop a hypothesis framework which can predict the evolutionary outcomes observed in our results. The first level of the observations is the selection status (positive selection, neutral evolution or purification).

Building upon the condition proposed by Williams, which compares products of *m-values* and *p-values* of the beneficial and the detrimental effects ($|m_+p_+|?|m_-p_-|$), we partitioned the space of AP genes into 27 subspaces for each of which we could predict the evolutionary outcome. We constructed and run experiments which confirmed the theoretical predictions.

4.2.2 Selection strength

The force of positive selection on AP genes can be of various magnitudes. In the previous section, we presented a theoretical framework that successfully predicts evolutionary outcomes of AP genes in AEGIS+. In this section, we will discuss the strength of positive selection and its theoretical underpinnings. Also, we will contextualize our results in the theoretical framework from the previous section by comparing the results with the theoretical expectations and discussing potential deviations from them.

Quantitative differences in evolutionary outcomes

In the section 3.2.1, we have used the AP genes $(1_{+0.2}, 49_{-0.5})$, $(13_{+0.2}, 48_{-0.5})$ as examples of classical AP genes. We observed that both genes were positively selected in both the smaller and the larger population. This is in concordance with our theoretical framework[4].

However, there are still differences between the outcomes of the larger and the smaller population. In contrast to the smaller population, larger population positively selected more bits of the AP loci, which indicates that the selection is more robust in larger population than in the smaller one.

Since the only difference between the simulations with the smaller and the bigger populations is population size, the difference in the evolutionary outcome is caused by the size differential. When population size is small, genetic drift is high, which weakens the force of natural selection. In practice, this means that trajectories of allele frequencies are more erratic which ultimately increases the chance of suboptimal gene variants rising in frequency and achieving fixation. When population size is big, genetic drift is low, so the force of natural selection operates more *smoothly*. In practice, this means that trajectories of allele frequencies exhibit more stable trends and that suboptimal gene variants are less likely to rise in frequency or achieve fixation.

If we understand AP loci with different number of bits as different alleles of the same gene, we can apply the theory of genetic drift to understand why smaller populations evolve AP genes with a smaller number of positively selected bits (in this case 2-3 bits fewer) than the larger populations. In the smaller population, the benefit of an allele with more bits is not significantly stronger than the random changes in allele frequencies that occur due to stochasticity introduced by genetic drift; thus, smaller populations end up with the suboptimal variants.

Qualitative differences in evolutionary outcomes

In the section 3.2.2, we have encoded 10 AP loci: $(5_{+0.2}, 30_{-0.5})$, $(6_{+0.1}, 31_{-0.5})$, $(7_{+0.05}, 32_{-0.5})$, $(8_{+0.02}, 33_{-0.5})$, $(9_{+0.01}, 34_{-0.5})$, $(10_{+0.2}, 40_{-0.5})$, $(11_{+0.2}, 41_{-0.5})$, $(12_{+0.2}, 42_{-0.5})$, $(13_{+0.2}, 43_{-0.5})$, and $(14_{+0.2}, 44_{-0.5})$. According to the theoretical framework, all ten AP genes should exhibit positive selection[5]. However, this seems to not be in total concordance with our results.

Our results suggest that all AP genes are positively selected except that the third, fourth, fifth and tenth AP genes evolve neutrally in the smaller population and the fourth and fifth AP genes evolve neutrally in the larger population.

In the section above, we explained that population size, via genetic drift, modulates

[4]The AP gene $(1_{+0.2}, 49_{-0.5})$ is of type 22, while the AP gene $(13_{+0.2}, 48_{-0.5})$ is of type 25 of our theoretical framework (table 4.4). The theoretical expectations are positive selection in both cases as is observed. Using another approach, positive selection should be expected in both cases since the detrimental effect occurs in postreproductive period, thus causing no real fitness loss, while some fitness gains are achieved by the beneficial effect. More formally, $p_- = 0$ and $p_+ > 0$ for both AP genes which means that $|m_+p_+| > |m_-p_-| = 0$ which should result in positive selection.

[5]First five genes are of the type 22, while the last five genes are of the type 25 of our theoretical framework (table 4.4). This is similar to the section just above.

the strength of natural selection. If genetic drift is very strong, it is possible that it overpowers the force of natural selection thus leading to a neutral evolution of an allele. Since the beneficial effect (m_+) of the first five AP genes decreases with each gene, it is expected that the force of natural selection decreases with it; and this leads to higher susceptibility to genetic drift. In the last five AP genes, the magnitudes of the effects are constant but the proportion of the affected reproduction distribution decreases (p_+) as the beneficial effects occur later and later in life; and this leads to higher susceptibility to genetic drift, too.

In both populations, genetic drift causes some genes to elude positive selection and exhibit neutral evolution, but the number of such genes is much higher in the smaller population.

Also, this explains why our results are in apparent discordance with our theoretical framework. If genetic drift overpowers positive selection, selection will be masked and the relevant genes will appear to evolve neutrally. However, the theoretical expectation is still positive selection. This differs from a scenario in which the theoretical expectation is neutral evolution, which means that there are no conditions in which the AP gene can evolve in any other fashion other than a neutral one.

Inversion of evolutionary outcomes

In the section 3.2.3, we have used the following AP genes: $(13_{+0.5}, 5_{-0.2})$ and $(5_{+0.25}, 13_{-0.5})$, in the first and the second example respectively.

These AP genes belong to cases `11` and `24` in the theoretical framework (table 4.4) for which the theoretical framework offers no prediction.

In the case `11`, we know that $0 < p_+ < 1$, $p_- = 1$, and $|m_+| > |m_-|$; however, this is not sufficient to determine whether $|m_+p_+|$ is greater than, equal to or less than $|m_-p_-|$. Intuitively, this is because the beneficial effect is stronger than the detrimental effect but it occurs later in life; therefore, the final outcome depends on the actual *m-values* and *p-values*.

In the case `24`, the opposite is true. We know that $0 < p_- < 1$, $p_+ = 1$, and $|m_+| < |m_-|$; thus we cannot determine the relation of the mp products and ultimately predict the evolutionary outcome.

When it comes to our results (section 3.2.3), in the example 1 (case `11`), we have observed purifying selection, while in the example 2 (case `24`), we have observed positive selection. Now, to show that a change in the balance of the factors will lead to an inversion of evolutionary outcomes, we have used an indirect approach. Instead of changing the factors themselves, we have changed the population size, while the AP configuration remained the same (i.e. $(13_{+0.5}, 5_{-0.2})$ and $(5_{+0.25}, 13_{-0.5})$). This proved to work as the purified AP locus in the first example became positively selected; and the positively selected AP locus in the second example became purified.

Note that this scenario is fundamentally different than the one from the previous section (3.2.2) wherein two loci that were supposed to be positively selected exhibited drift-like evolutionary pattern instead. Here, the violated expectation is not met with a drift-like evolution but with the counter-expectation. How can that be? At this

point, it is important to take a step back and to connect the observations with some previous conclusions.

When we examine the Visor profiles of the evolved populations (figures 3.28 and 3.29), we can suspect that the underlying driver of this inversion is—extended lifespan. As already reported, population size can, by modulating genetic drift, affect lifespan as it renders loci more or less malleable to the forces of positive selection—a phenomenon known as *selection shadow*. This is the previously described effect of population size on MA (section 4.1); and it is exactly what can be observed as larger populations manage to positively select survival loci up to age 21 (figure 3.29), while the smaller populations manage to do the same only up to age 15 (figure 3.28) in this inversion experiment.

This lifespan extension leads to an increase of *p-values* as increasingly larger portions of the reproduction distribution occur after the onsets of the given effects [6]. This indicates that aging in the evolution of life history emerges from the interplay of genetic drift, mutation accumulation and antagonistic pleiotropism.

Note that this observation has more general consequences than those explored in these two particular examples. In our experiments, it was assumed for the sake of simplicity that AP genes would not cross from one type (table 4.4) into another during evolution[7]. However, this is wholly possible and is very much biologically relevant as life extension can and does uncover previously latent detrimental effects of positively selected AP genes.

In conclusion of selection strength prediction

Experimental results indicated that larger populations select AP genes more robustly (section 3.2.1), which can lead to a more significant acquisition of AP genes (3.2.2) which ultimately contribute more to the shaping of life history. In this discussion, we explained how population size, via genetic drift, modulates the strength of selection leading to deviation from the theoretized outcomes to a drift-like evolutionary patterns. These drift-like patterns decrease the relevance of AP genes in shaping the life history of smaller populations. Furthermore, we demonstrated that genetic drift also has an indirect effect on the evolution of AP genes by modulating the effect of mutation accumulation on life history.

[6] However, the *p-values* still come from the interval $(0, 1)$ since the effects occur after maturity but before the postreproductive period; this means that they remain in their theoretical categories

[7] For example, by onset of the effect changing from the premature period to the mature period

Appendix

Visor elements

Apart from the Visor profile, Visor contains other elements which are described in detail here.

Death structure

On the side, there is a stacked bar plot. This bar plot visualizes the death distribution across age and death causes. Each column visualizes the death counts at a particular age, while colors represent the various causes of death: green the ecological death, orange the genetic death, and blue the simulational death (the death due to reached maximum lifespan; this can only occur in the rightmost column and is only present in exceptional parameterizations).

Note that the bar length is fixed, and all the bars are normalized according to that max value. This is important to keep in mind when comparing two snapshots, as their normalizing factors may be different thus a bar that is shorter in one visualization than in another may actually represent more deaths [8].

Age slider

Since simulations can run for a long time, multiple snapshots are taken at various times. Therefore, there is a slider at the top which can be used to select the snapshot to display. Sliding the knob to the left chooses snapshots taken earlier in the simulation, while sliding it to the right chooses snapshots taken later.

Play button

There is also a play button to the right of the slider that, when activated, cycles through the snapshots so we can have a smooth look at the progression of the evolution.

Termination condition for AEGIS

In previous work on AEGIS [65], a termination condition has been proposed:

$$K > \log\left(\delta \frac{1+\nu}{|p_0(1+\nu)-\nu|} - |1-\mu(1+\nu)|\right)$$

[8] Imagine that in one case, there are 20000 deaths that are uniformly distributed over twenty bars. All bars are of the same max width and represent 1000 deaths. Then, imagine that in another case, also with 20000 deaths, the first stage represents 10000 deaths. The normalization factor here is 10000, and the width of a bar translates into ten times more deaths that in the previous case.

such that the K is the minimum number of generations that should be simulated before we can expect the neutral loci to attain values that do not deviate more than δ from their equilibrium values.

ν represents the *mutation ratio*[9], μ represents the rate of negative mutations (mutation probability), and p_0 represents the average value of a neutral locus at generation zero.

The fundamental concern with using this approach is that it may lead to an underestimation of the required time to stabilization. This is mostly because genomes evolve gradually and evolutionary forces shift as the population genetics changes. This is most apparent on populations that experience an extension of lifespan (such as in figure 4.8) as genes encoding for survival and reproduction at later ages become positively selected only once the genes acting on earlier life are positively selected themselves. The same is true for lifespan restriction in which genes encoding for early life, which are initially positively selected, only start to evolve more neutrally as more and more downstream genes lose positive selection status (such as in figure 4.9). This aspect of evolution is not at all accounted for with this approach using neutral loci which are subject to constant evolutionary forces throughout the simulation.

Another concern is that even if the kinetics of other loci is similar to those of the neutral loci, neutral loci have different equilibrium values than non-neutral loci, thus their evolutionary paths are different. This is especially important if neutral loci have equilibrium values that are lower than their initial values, while non-neutral loci have equilibrium values that are higher than their initial values; and that is because accumulating and selecting positive mutations occurs with significantly different kinetics than accumulating (and not selecting) negative mutations (figure 4.8).

This stepwise evolutionary nature and the discrepancy between achieving equilibrium values for neutral loci vs positively selected loci is visible in the figures 4.8 and 4.9. These figures depict which loci have achieved neutral equilibrium values (with $\delta = 0.09 \approx \nu$) with black dots representing loci that have achieved neutral equilibrium values, and white dots those that have not. The loci are arranged along the x-axis, while the y-axis represents time (ordered from top to bottom). The illustrated loci are all reproduction loci with the x-value denoting the age at which the locus is active.

The first 10 loci (x-values from 0 to 9) evolve trully neutrally as no reproduction is allowed during that period (premature ages) so they have no phenotypic effect.

As we can see from the first figure (4.8), the neutral loci achieve near equilibrium values at y-value 2 (red circle), while at this point the population is still experiencing significant lifespan extension which does not end at least up to y-value of 10 (green circle).

[9]Mutation ratio is the ratio of the 0-to-1 mutation probability to the the 1-to-0 mutation probability.

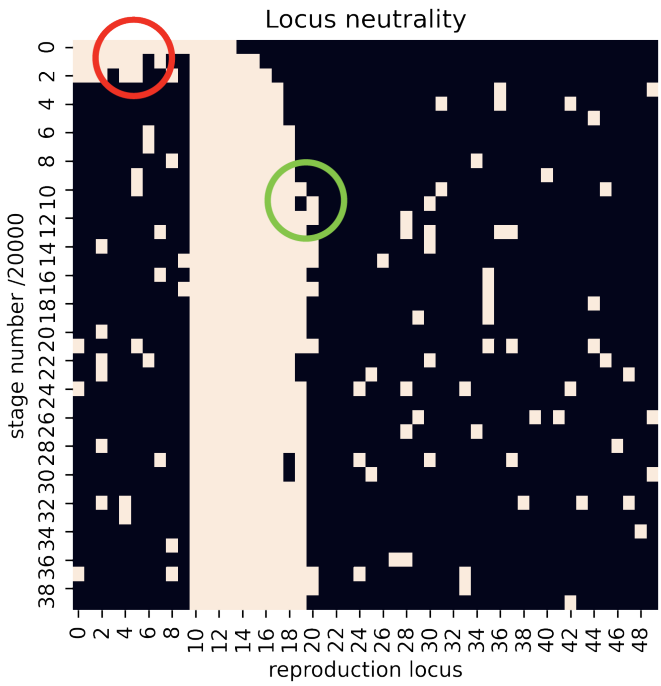


Figure 4.8: *Reproduction loci achieving neutral equilibrium values over time. Black color indicates that neutral equilibrium values are achieved, while white color indicates that they are not achieved. Population size is 20000. Red circle highlights the neutral loci at times at which they are about to achieve they eqilibrium values. Green circle highlights the moment at which the population appears to have achieved its equilibrium.*

In the second figure (4.9), the neutral loci achieve near equilibrium values at approximately y-value 40 (red circle); however, the population is still evolving at least up to y-value 190 (green circle).

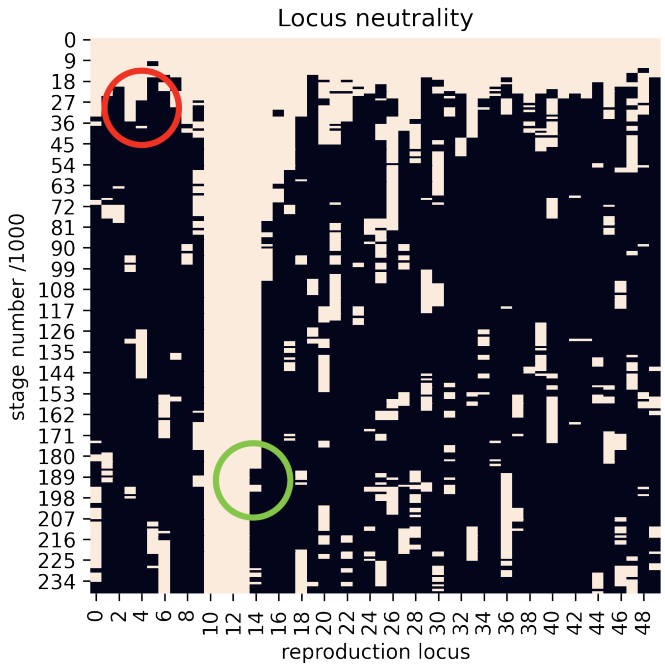


Figure 4.9: *Reproduction loci achieving neutral equilibrium values over time. Black color indicates that neutral equilibrium values are achieved, while white color indicates that they are not achieved. Population size is 1000. Red circle highlights the neutral loci at times at which they are about to achieve they eqilibrium values. Green circle highlights the moment at which the population appears to have achieved its equilibrium.*

There are other minor caveats to this approach. For example, one concern is that the connection between the number of generations and number of stages is fuzzy as reproduction does not occur at regular intervals[10]; instead the time from birth to reproduction can vary widely. Another concern is that this approach only works for uniform interpreters[11].

In conclusion, the proposed termination approach may not be generally usable for all parameter configurations in AEGIS+ but also it may have a tendency to underestimate the simulation time needed to achieve equilibrium values. This may be because evolution of non-neutral loci is more complex than the evolution of neutral loci. Since reliably determining the time of termination is difficult, this thesis uses another technique which instead gives information about the distance of the evolved population from the equilibrium population instead.

Testing theoretical predictions of selection status of AP genes

This section provides the rest of the plots of evolved AP genes from the section 4.2. These AP genes have configurations given in the table 4.4 in all but first three rows.

[10] At least not in the overlapping generations.

[11] When uniform interpreters are used, the phenotypic value of a locus is the average of its bits.

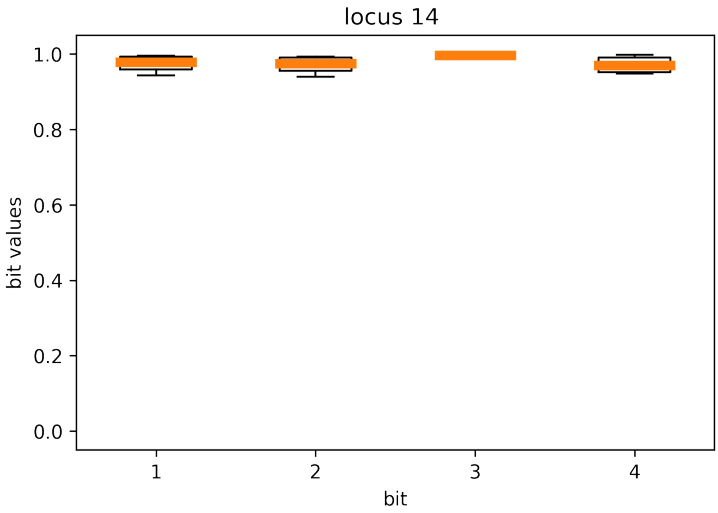


Figure 4.10: *Bit values of the AP gene 4. All bits have high values, thus the AP gene is positively selected*

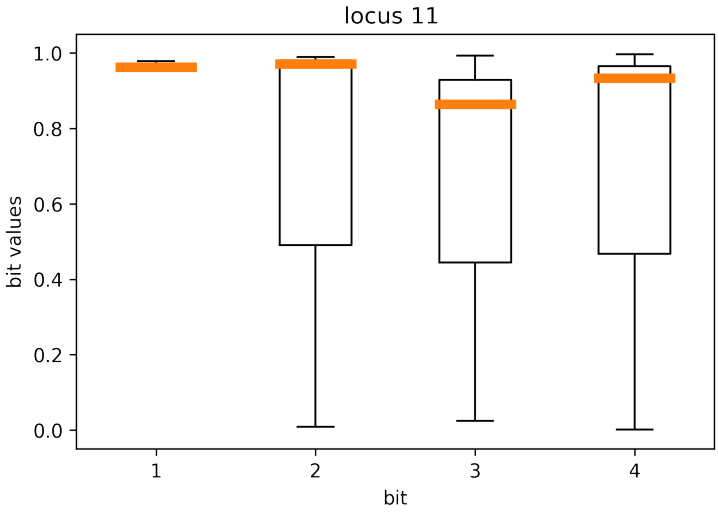


Figure 4.11: *Bit values of the AP gene 5. Values of bits are highly variant, thus the AP gene is neutrally evolving*

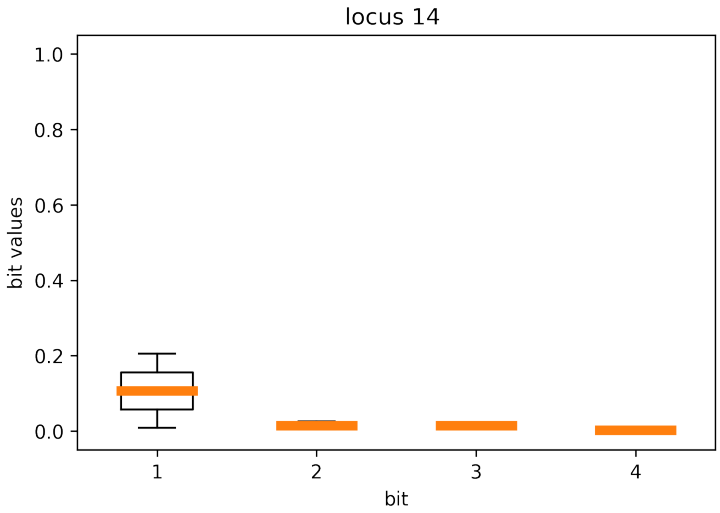


Figure 4.12: *Bit values of the AP gene 6. All bits have low values, thus the AP gene is negatively selected*

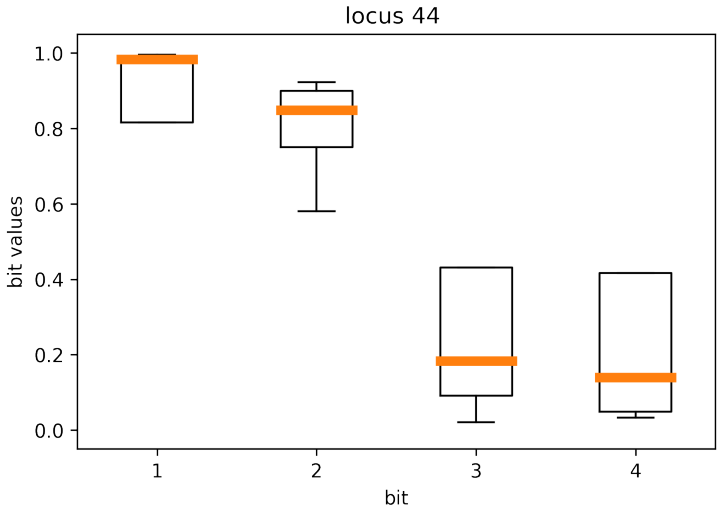


Figure 4.13: *Bit values of the AP gene 7. Values of bits are highly variant and mutually different, thus the AP gene is neutrally evolving*

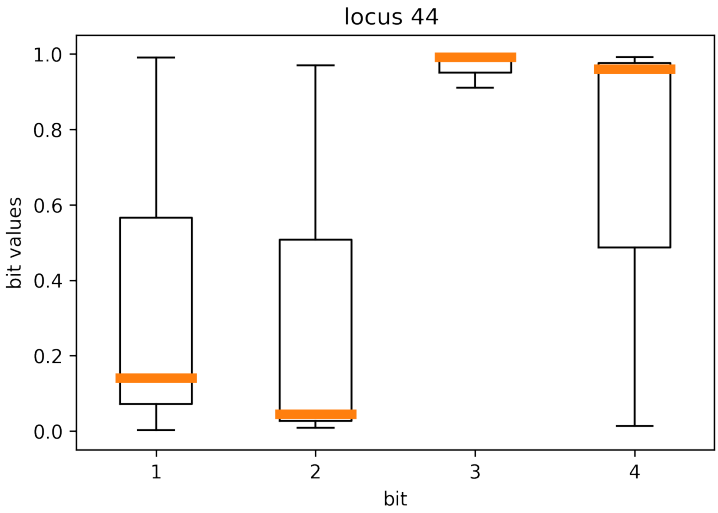


Figure 4.14: *Bit values of the AP gene 8. Values of bits are highly variant and mutually different, thus the AP gene is neutrally evolving*

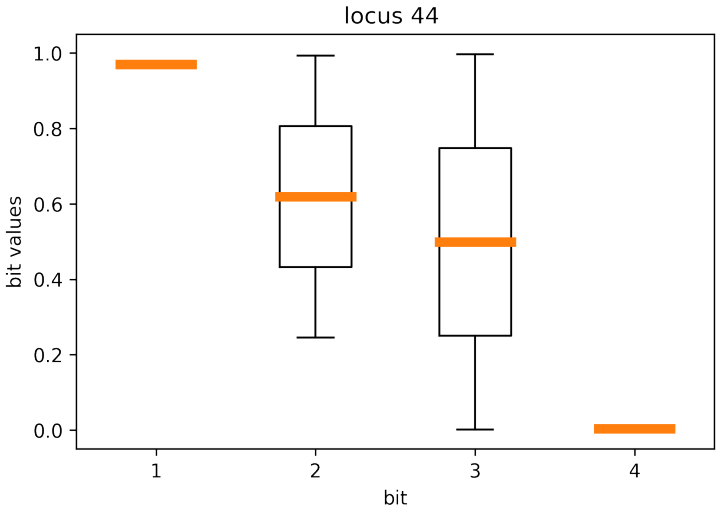


Figure 4.15: *Bit values of the AP gene 9. Values of bits are highly variant and mutually different, thus the AP gene is neutrally evolving*

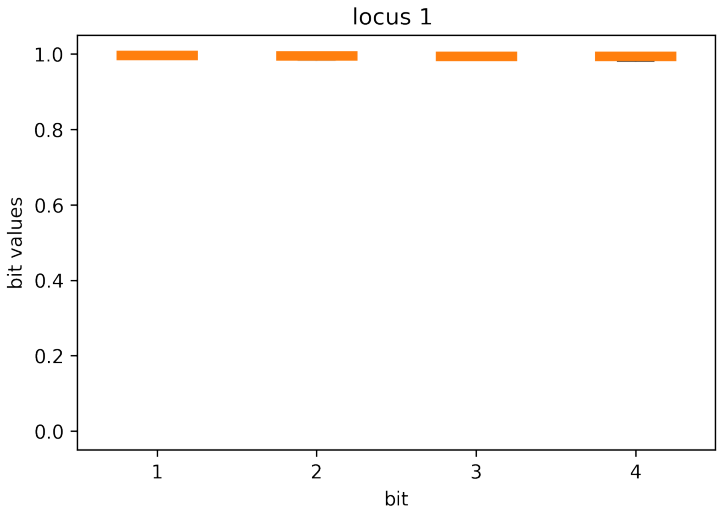


Figure 4.16: *Bit values of the AP gene 10. All bits have high values, thus the AP gene is positively selected*

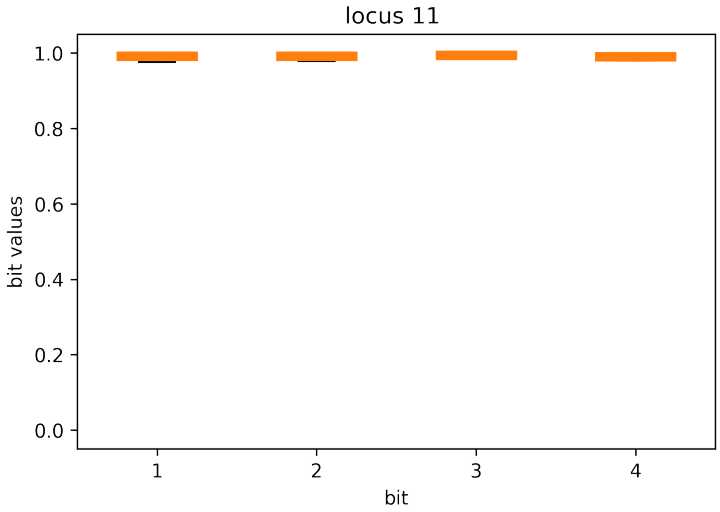


Figure 4.17: *Bit values of the AP gene 11. All bits have high values, thus the AP gene is positively selected*

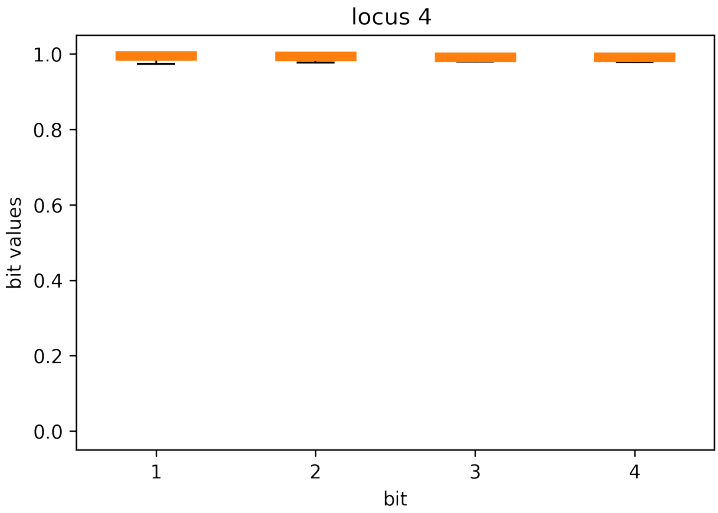


Figure 4.18: *Bit values of the AP gene 12. All bits have high values, thus the AP gene is positively selected*

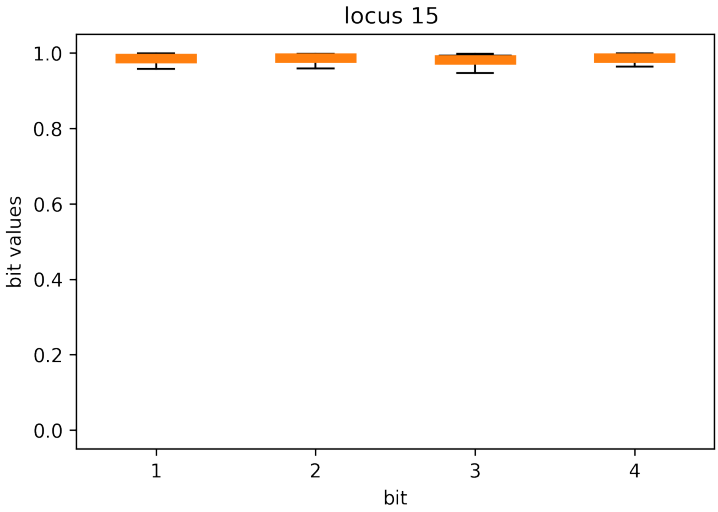


Figure 4.19: *Bit values of the AP gene 13. All bits have high values, thus the AP gene is positively selected*

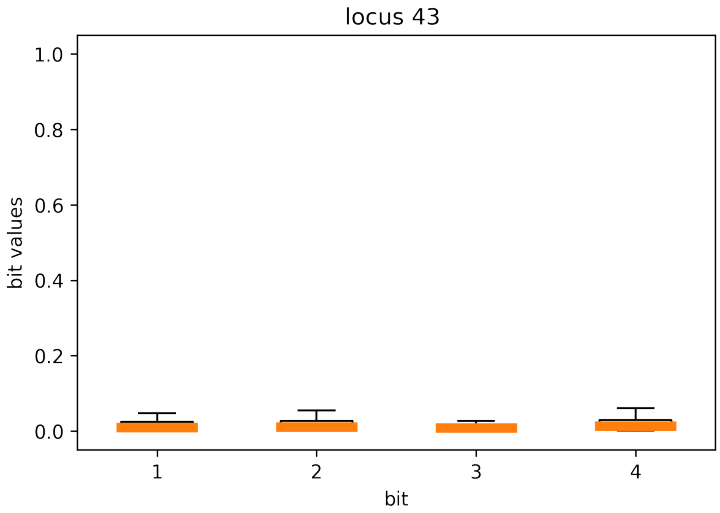


Figure 4.20: *Bit values of the AP gene 14. All bits have low values, thus the AP gene is negatively selected*

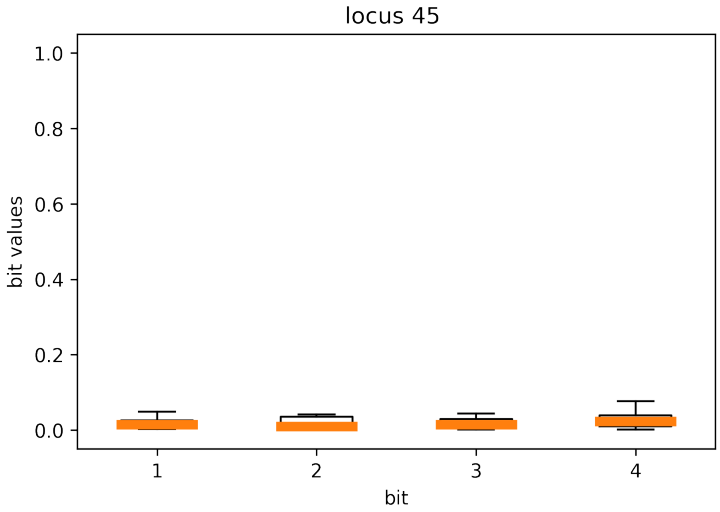


Figure 4.21: *Bit values of the AP gene 15. All bits have low values, thus the AP gene is negatively selected*

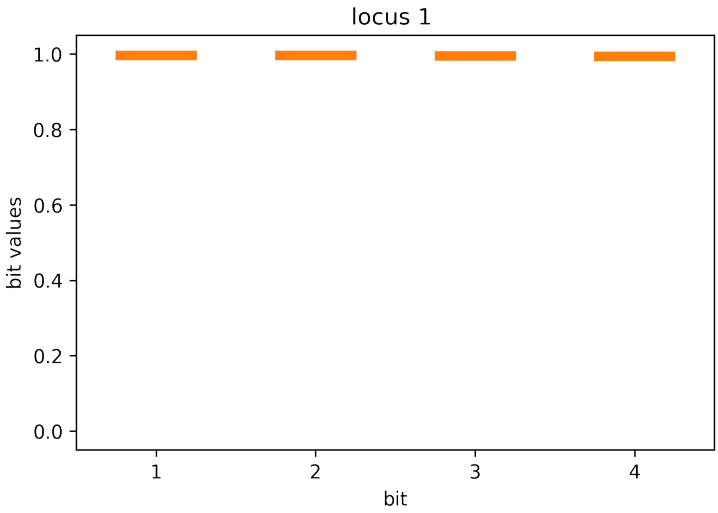


Figure 4.22: *Bit values of the AP gene 16. All bits have high values, thus the AP gene is positively selected*

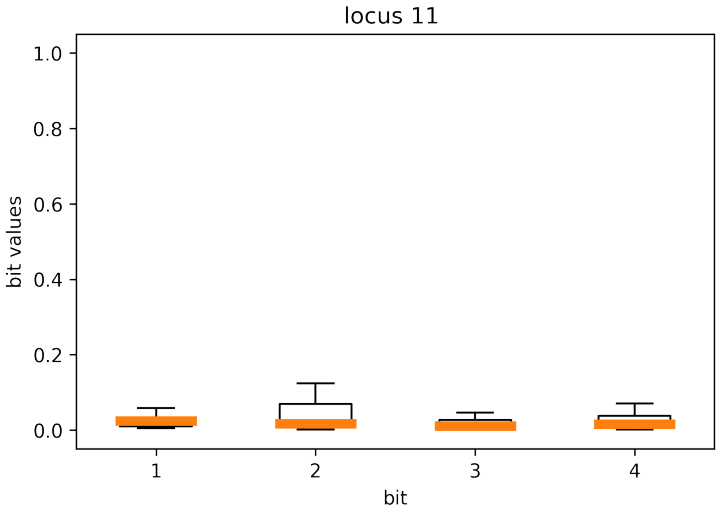


Figure 4.23: *Bit values of the AP gene 17. All bits have low values, thus the AP gene is negatively selected*

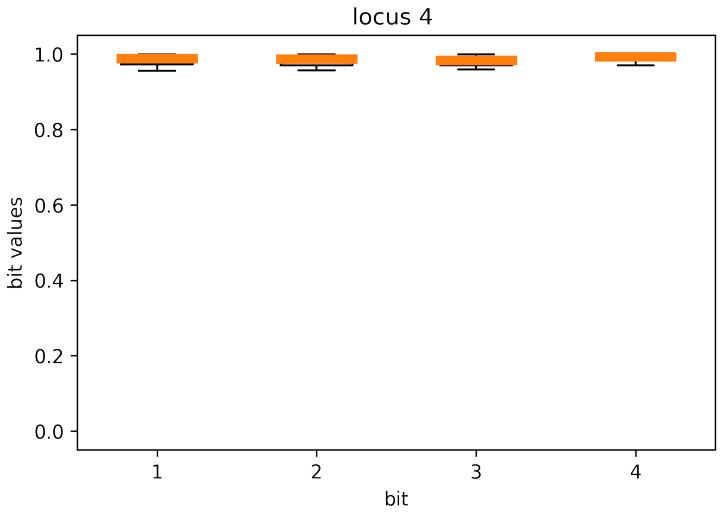


Figure 4.24: *Bit values of the AP gene 18. All bits have high values, thus the AP gene is positively selected*

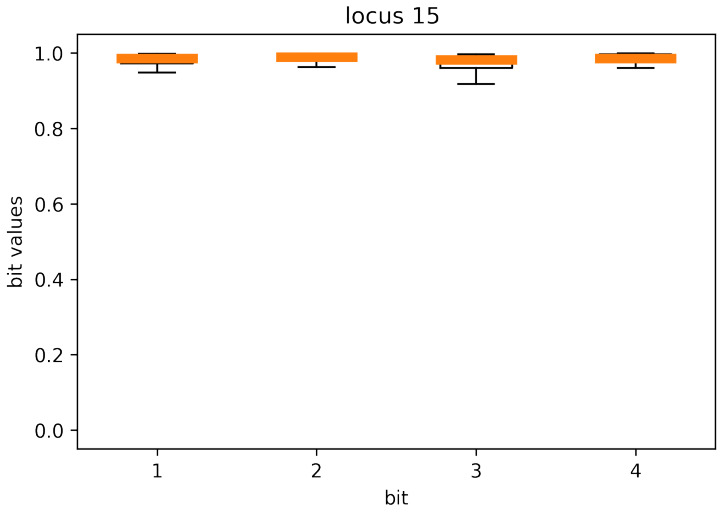


Figure 4.25: *Bit values of the AP gene 19. All bits have high values, thus the AP gene is positively selected*

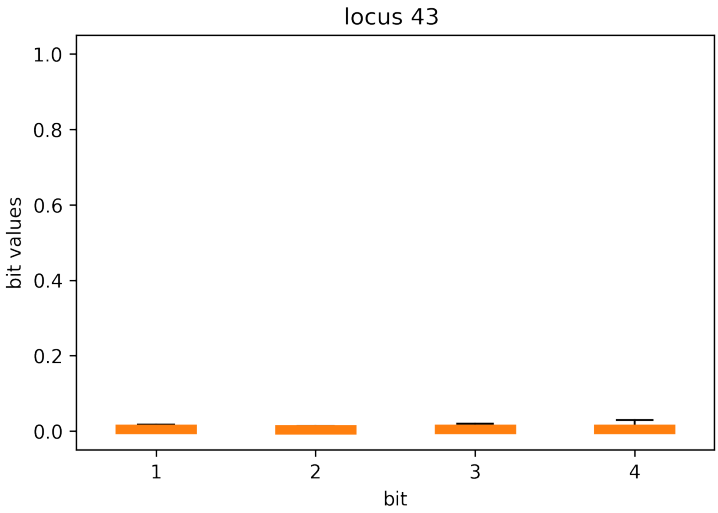


Figure 4.26: *Bit values of the AP gene 20. All bits have low values, thus the AP gene is negatively selected*

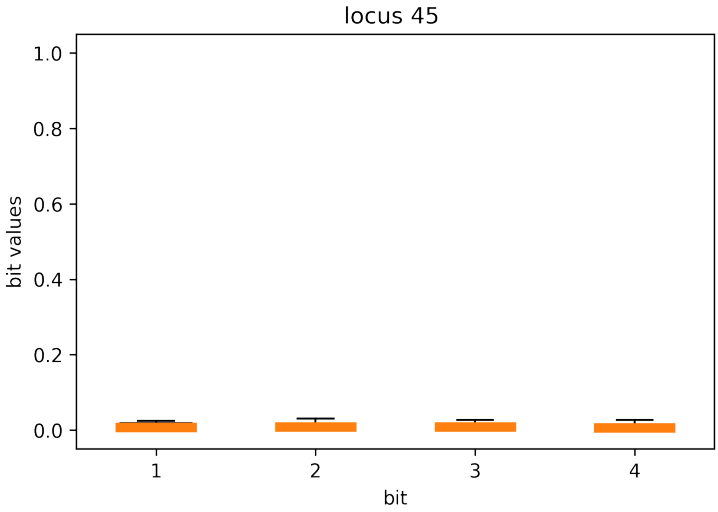


Figure 4.27: *Bit values of the AP gene 21. All bits have low values, thus the AP gene is negatively selected*

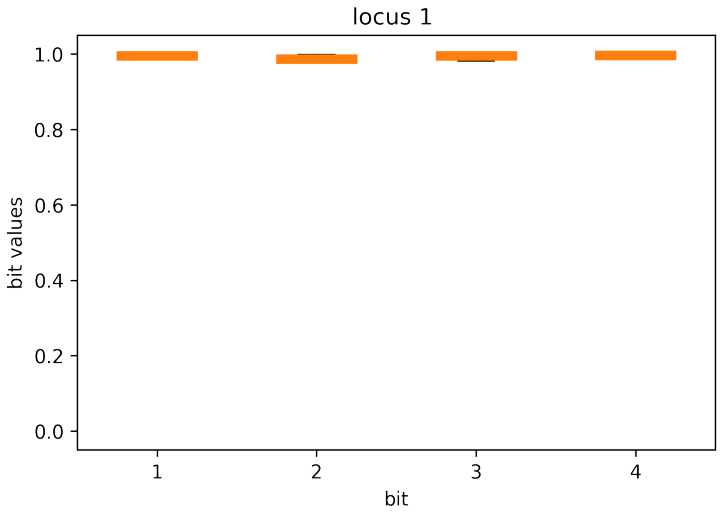


Figure 4.28: *Bit values of the AP gene 22. All bits have high values, thus the AP gene is positively selected*

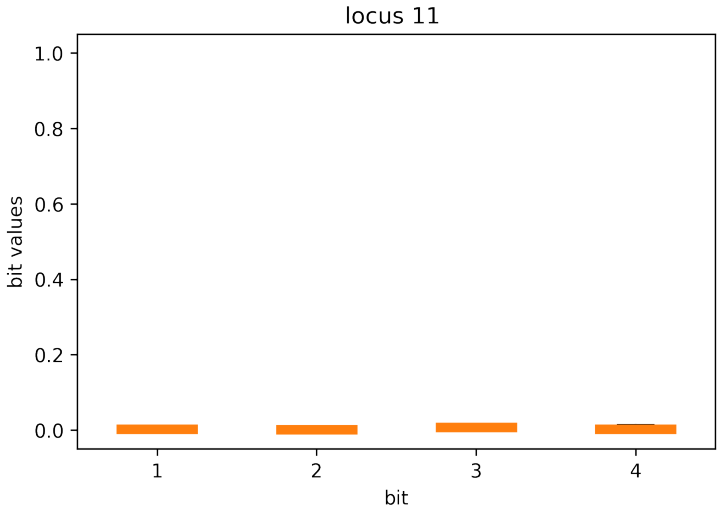


Figure 4.29: *Bit values of the AP gene 23. All bits have low values, thus the AP gene is negatively selected*

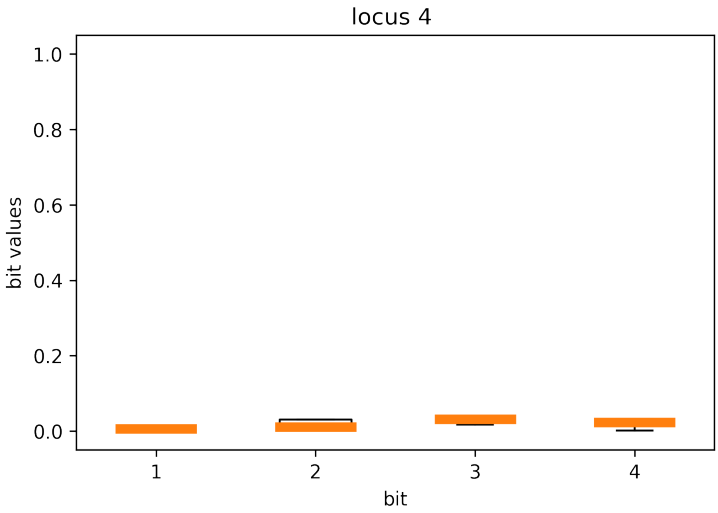


Figure 4.30: *Bit values of the AP gene 24. All bits have low values, thus the AP gene is negatively selected*

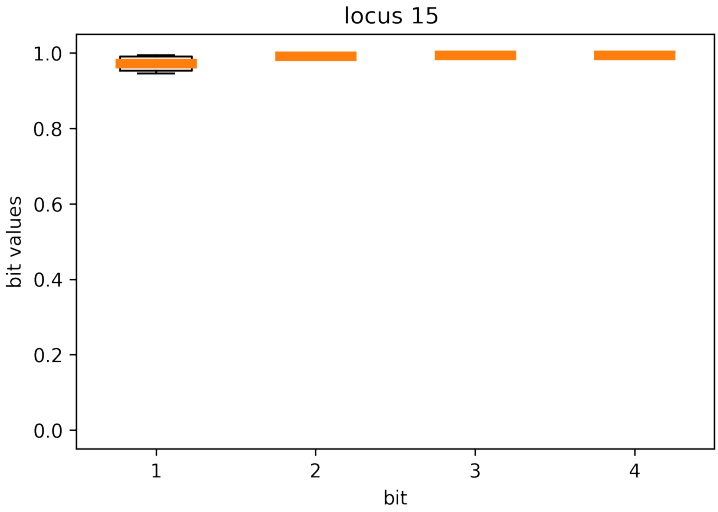


Figure 4.31: *Bit values of the AP gene 25. All bits have high values, thus the AP gene is positively selected*

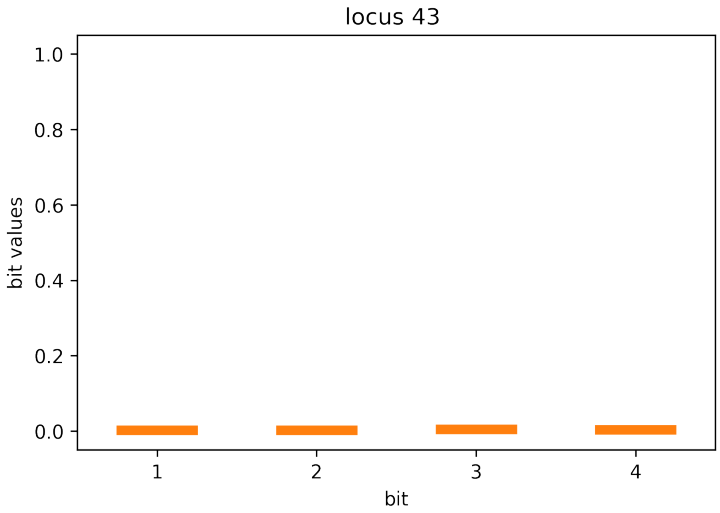


Figure 4.32: *Bit values of the AP gene 26. All bits have low values, thus the AP gene is negatively selected*

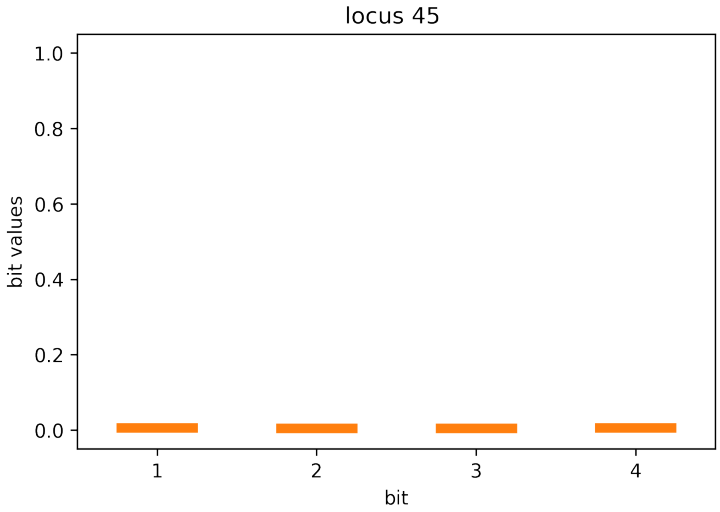


Figure 4.33: *Bit values of the AP gene 27. All bits have low values, thus the AP gene is negatively selected*

Bibliography

[1] Michael Rose et al. "What is aging?" In: *Frontiers in genetics* 3 (2012), p. 134.

[2] Thomas Flatt. "A new definition of aging?" In: *Frontiers in genetics* 3 (2012), p. 148.

[3] Alex Comfort. "The biology of senescence." In: (1956).

[4] Judith Campisi. "Aging, cellular senescence, and cancer". In: *Annual review of physiology* 75 (2013), pp. 685–705.

[5] Don H Anderson et al. "The pivotal role of the complement system in aging and age-related macular degeneration: hypothesis re-visited". In: *Progress in retinal and eye research* 29.2 (2010), pp. 95–112.

[6] Ronald C Petersen. *Mild cognitive impairment: aging to Alzheimer's disease*. Oxford University Press, 2003.

[7] Ramón Cacabelos et al. "Molecular genetics of Alzheimer's disease and aging." In: *Methods and findings in experimental and clinical pharmacology* 27 (2005), pp. 1–573.

[8] Gilberto Levy. "The relationship of Parkinson disease with aging". In: *Archives of neurology* 64.9 (2007), pp. 1242–1246.

[9] Julie C Wang and Martin Bennett. "Aging and atherosclerosis: mechanisms, functional consequences, and potential therapeutics for cellular senescence". In: *Circulation research* 111.2 (2012), pp. 245–259.

[10] Mark Emberton et al. "Benign prostatic hyperplasia: a progressive disease of aging men1". In: *Urology* 61.2 (2003), pp. 267–273.

[11] Mohammed Yousufuddin and Nathan Young. "Aging and ischemic stroke". In: *Aging (Albany NY)* 11.9 (2019), p. 2542.

[12] Rexford S Ahima. "Connecting obesity, aging and diabetes". In: *Nature medicine* 15.9 (2009), pp. 996–997.

[13] Erica PA Rutten et al. "Various mechanistic pathways representing the aging process are altered in COPD". In: *Chest* 149.1 (2016), pp. 53–61.

[14] Kazuhiro Ito and Peter J Barnes. "COPD as a disease of accelerated lung aging". In: *Chest* 135.1 (2009), pp. 173–180.

[15] George C Williams. "Pleiotropy, natural selection, and the evolution of senescence". In: *evolution* (1957), pp. 398–411.

[16] Steven N Austad and Jessica M Hoffman. "Is antagonistic pleiotropy ubiquitous in aging biology?" In: *Evolution, medicine, and public health* 2018.1 (2018), pp. 287–294.

[17] Predrag Ljubuncic and Abraham Z Reznick. "The evolutionary theories of aging revisited–a mini-review". In: *Gerontology* 55.2 (2009), pp. 205–216.

[18] Josh Mitteldorf. "Is Programmed Aging a Cause for Optimism?" In: *Current Aging Science* 8.1 (Mar. 2015), pp. 69–75. ISSN: 1874-6098. URL: https://www.ingentaconnect.com/content/ben/cas/2015/00000008/00000001/art00011.

[19] Theodore C. Goldsmith. "Aging as an evolved characteristic – Weismann's theory reconsidered". In: *Med. Hypotheses* 62.2 (Feb. 2004), pp. 304–308. ISSN: 0306-9877. DOI: 10.1016/S0306-9877(03)00337-2.

[20] J. J. Mitteldorf. "Adaptive aging in the context of evolutionary theory". In: *Biochemistry (Mosc.)* 77.7 (July 2012), pp. 716–725. ISSN: 1608-3040. DOI: 10.1134/S0006297912070036.

[21] Josh Mitteldorf and John Pepper. "Senescence as an adaptation to limit the spread of disease". In: *J. Theor. Biol.* 260.2 (Sept. 2009), pp. 186–195. ISSN: 0022-5193. DOI: 10.1016/j.jtbi.2009.05.013.

[22] Peter V. Lidsky and Raul Andino. "Protection from epidemics is a driving force for evolution of lifespan setpoints". In: *bioRxiv* (Nov. 2017), p. 215202. eprint: 215202. URL: https://doi.org/10.1101/215202.

[23] Vladimir P. Skulachev. "The programmed death phenomena, aging, and the Samurai law of biology". In: *Exp. Gerontol.* 36.7 (July 2001), pp. 995–1024. ISSN: 0531-5565. DOI: 10.1016/S0531-5565(01)00109-7.

[24] Andrew F.G. Bourke. "Kin Selection and the Evolutionary Theory of Aging". In: *Annual Review of Ecology, Evolution, and Systematics* 38.1 (Dec. 2007), pp. 103–128. DOI: 10.1146/annurev.ecolsys.38.091206.095528. URL: https://doi.org/10.1146/annurev.ecolsys.38.091206.095528.

[25] Ronald D. Lee. "Rethinking the evolutionary theory of aging: Transfers, not births, shape senescence in social species". In: *Proc. Natl. Acad. Sci. U.S.A.* 100.16 (Aug. 2003), pp. 9637–9642. ISSN: 0027-8424. DOI: 10.1073/pnas.1530303100.

[26] Joshua Mitteldorf. "Evolutionary Orthodoxy: How and Why the Evolutionary Theory of Aging Went Astray". In: *Current Aging Science* 7.1 (Feb. 2014), pp. 38–47. ISSN: 1874-6098. URL: https://www.ingentaconnect.com/content/ben/cas/2014/00000007/00000001/art00007.

[27] Piotr Paweł Chmielewski. "Human ageing, longevity and evolution: can ageing be programmed?" In: *Anthropological Review* 82.4 (Dec. 2019), pp. 417–433. DOI: 10.2478/anre-2019-0032.

[28] Peter Brian Medawar. "An unsolved problem of biology". In: (1952).

[29] Thomas BL Kirkwood. "Evolution of ageing". In: *Nature* 270.5635 (1977), pp. 301–304.

[30] Peter Brian Medawar. *The uniqueness of the individual*. Routledge, 2019.

[31] Peter Brian Medawar. "Old age and natural death". In: *Mod. Quart.* 2 (1946), pp. 30–49.

[32] Thomas BL Kirkwood and Robin Holliday. "The evolution of ageing and longevity". In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 205.1161 (1979), pp. 531–546.

[33] Michael R Rose. *Evolutionary biology of aging*. Oxford University Press on Demand, 1994.

[34] Thomas BL Kirkwood. "3 Age Differences in Evolutionary". In: *Understanding Human Development: Dialogues with Lifespan Psychology* (2003), p. 45.

[35] Ronald D Lee. "Rethinking the evolutionary theory of aging: transfers, not births, shape senescence in social species". In: *Proceedings of the National Academy of Sciences* 100.16 (2003), pp. 9637–9642.

[36] Eric Le Bourg. "Evolutionary theories of aging can explain why we age". In: *Aging* 39 (2014), pp. 8–23.

[37] Alexei A Maklakov. "Aging: why do organisms live too long?" In: *Current Biology* 23.22 (2013), R1003–R1005.

[38] Daniel Fabian and Thomas Flatt. "The evolution of aging". In: *Nature Education Knowledge* 3.3 (2011), pp. 1–10.

[39] Vadim N Gladyshev. "Aging: progressive decline in fitness due to the rising deleteriome adjusted by genetic, environmental, and stochastic processes". In: *Aging cell* 15.4 (2016), pp. 594–602.

[40] Merril Silverstein et al. *Handbook of theories of aging*. Springer Publishing Company, 2008.

[41] Juan Antonio Rodriguez et al. "Antagonistic pleiotropy and mutation accumulation influence human senescence and disease". In: *Nature Ecology & Evolution* 1.3 (2017), pp. 1–5.

[42] Erping Long and Jianzhi Zhang. "Retesting the influences of mutation accumulation and antagonistic pleiotropy on human senescence and disease". In: *Nature ecology & evolution* 3.7 (2019), pp. 992–993.

[43] Juan Antonio Rodriguez et al. "Reply to: Retesting the influences of mutation accumulation and antagonistic pleiotropy on human senescence and disease". In: *Nature ecology & evolution* 3.7 (2019), pp. 994–995.

[44] Alexei A Maklakov, Locke Rowe, and Urban Friberg. "Why organisms age: Evolution of senescence under positive pleiotropy?" In: *Bioessays* 37.7 (2015), pp. 802–807.

[45] Leonid A Gavrilov and Natalia S Gavrilova. "Evolutionary theories of aging and longevity". In: *TheScientificWorldJOURNAL* 2 (2002), pp. 339–356.

[46] Leonid A Gavrilov and Natalia S Gavrilova. "Reliability theory of aging and longevity". In: *Handbook of the Biology of Aging*. Elsevier, 2005, pp. 3–42.

[47] Leonid A Gavrilov and Natalia S Gavrilova. "The quest for a general theory of aging and longevity". In: *Science of Aging Knowledge Environment* 2003.28 (2003), p. 5.

[48] C Franceschi et al. "The network and the remodeling theories of aging: historical background and new perspectives". In: *Experimental gerontology* 35.6-7 (2000), pp. 879–896.

[49] TBL Kirkwood and A Kowald. "Network theory of aging". In: *Experimental gerontology* 32.4-5 (1997), pp. 395–399.

[50] Reinhard Buerger. "Mathematical properties of mutation-selection models". In: *Genetica* 102.0 (Mar. 1998), pp. 279–298. ISSN: 1573-6857. DOI: 10.1023/A:1017043111100.

[51] William D Hamilton. "The moulding of senescence by natural selection". In: *Journal of theoretical biology* 12.1 (1966), pp. 12–45.

[52] Ronald Aylmer Fisher. *The genetical theory of natural selection*. 1958.

[53] Brian Charlesworth et al. *Evolution in age-structured populations*. Vol. 2. Cambridge University Press Cambridge, 1994.

[54] J. A. Moorad and D. W. Hall. "Age-dependent mutational effects curtail the evolution of senescence by antagonistic pleiotropy". In: *Journal of Evolutionary Biology* 22.12 (2009), pp. 2409–2419.

[55] Fotios Drenos and Thomas BL Kirkwood. "Modelling the disposable soma theory of ageing". In: *Mechanisms of ageing and development* 126.1 (2005), pp. 99–103.

[56] Adam Prügel-Bennett. "Modelling Evolving Populations". In: *J. Theor. Biol.* 185.1 (Mar. 1997), pp. 81–95. ISSN: 0022-5193. DOI: 10.1006/jtbi.1996.0295.

[57] Jacob A. Moorad and Daniel E. L. Promislow. "A Theory of Age-Dependent Mutation and Senescence". In: *Genetics* 179.4 (Aug. 2008), pp. 2061–2073. ISSN: 0016-6731. DOI: 10.1534/genetics.108.088526.

[58] T. J. P. Penna. "A bit-string model for biological aging". In: *J. Stat. Phys.* 78.5 (Mar. 1995), pp. 1629–1633. ISSN: 1572-9613. DOI: 10.1007/BF02180147.

[59] T. J. P. Penna and D. Stauffer. "EFFICIENT MONTE CARLO SIMULATION OF BIOLOGICAL AGING". In: *Int. J. Mod. Phys. C* 06.02 (Apr. 1995), pp. 233–239. ISSN: 0129-1831. DOI: 10.1142/S0129183195000186.

[60] D. Stauffer. "The Penna Model of Biological Aging". In: *Bioinf. Biol. Insights* 1 (Jan. 2007), p. 117793220700100005. ISSN: 1177-9322. DOI: 10.1177/117793220700100005.

[61] Ronald Lee. "Sociality, selection, and survival: Simulated evolution of mortality with intergenerational transfers and food sharing". In: *Proc. Natl. Acad. Sci. U.S.A.* 105.20 (May 2008), pp. 7124–7128. ISSN: 0027-8424. DOI: 10.1073/pnas.0710234105.

[62] Martin Ackermann et al. "On the evolutionary origin of aging". In: *Aging Cell* 6.2 (Apr. 2007), pp. 235–244. ISSN: 1474-9718. DOI: 10.1111/j.1474-9726.2007.00281.x.

[63] Svetlana Krivenko and Mikhail Burtsev. "Simulation of the Evolution of Aging: Effects of Aggression and Kin-Recognition". In: *Advances in Artificial Life*. Berlin, Germany: Springer, Sept. 2007, pp. 84–92. ISBN: 978-3-540-74912-7. DOI: 10.1007/978-3-540-74913-4_9.

[64] Witold Dzwinel and David Yuen. "Aging in hostile environment modeled by cellular automata with genetic dynamics". In: *Int. J. Mod. Phys. C* 25.16 (Mar. 2005), pp. 357–376. DOI: 10.1142/S0129183105007182.

[65] William J. Bradshaw, Arian Šajina, and Dario Riccardo Valenzano. "AEGIS: An In Silico Tool to model Genome Evolution in Age-Structured Populations". In: *bioRxiv* (May 2019), p. 646877. eprint: 646877. URL: https://doi.org/10.1101/646877.

[66] D. Charlesworth, M. T. Morgan, and B. Charlesworth. "Mutation Accumulation in Finite Populations". In: *J. Hered.* 84.5 (Sept. 1993), pp. 321–325. ISSN: 0022-1503. DOI: 10.1093/oxfordjournals.jhered.a111351.

[67] Michael Lynch, John Conery, and Reinhard Burger. "Mutation accumulation and the extinction of small populations". In: *The American Naturalist* 146.4 (1995), pp. 489–518.

[68] Safe Martin et al. "On Stopping Criteria for Genetic Algorithms". In: *Advances in Artificial Intelligence – SBIA 2004*. Berlin, Germany: Springer, Sept. 2004, pp. 405–413. ISBN: 978-3-540-23237-7. DOI: 10.1007/978-3-540-28645-5_41.

[69] David Greenhalgh and Stephen Marshall. "Convergence Criteria for Genetic Algorithms". In: *SIAM J. Comput.* (July 2006). URL: https://epubs.siam.org/doi/abs/10.1137/S009753979732565X.

[70] Noraini Mohd Razali, John Geraghty, et al. "Genetic algorithm performance with different selection strategies in solving TSP". In: *Proceedings of the world congress on engineering*. Vol. 2. 1. International Association of Engineers Hong Kong. 2011, pp. 1–6.

[71] John A. Birdsell and Christopher Wills. "The Evolutionary Origin and Maintenance of Sexual Recombination: A Review of Contemporary Models". In: *Evolutionary Biology*. Boston, MA, USA: Springer, Boston, MA, 2003, pp. 27–138. ISBN: 978-1-4419-3385-0. DOI: 10.1007/978-1-4757-5190-1_2.